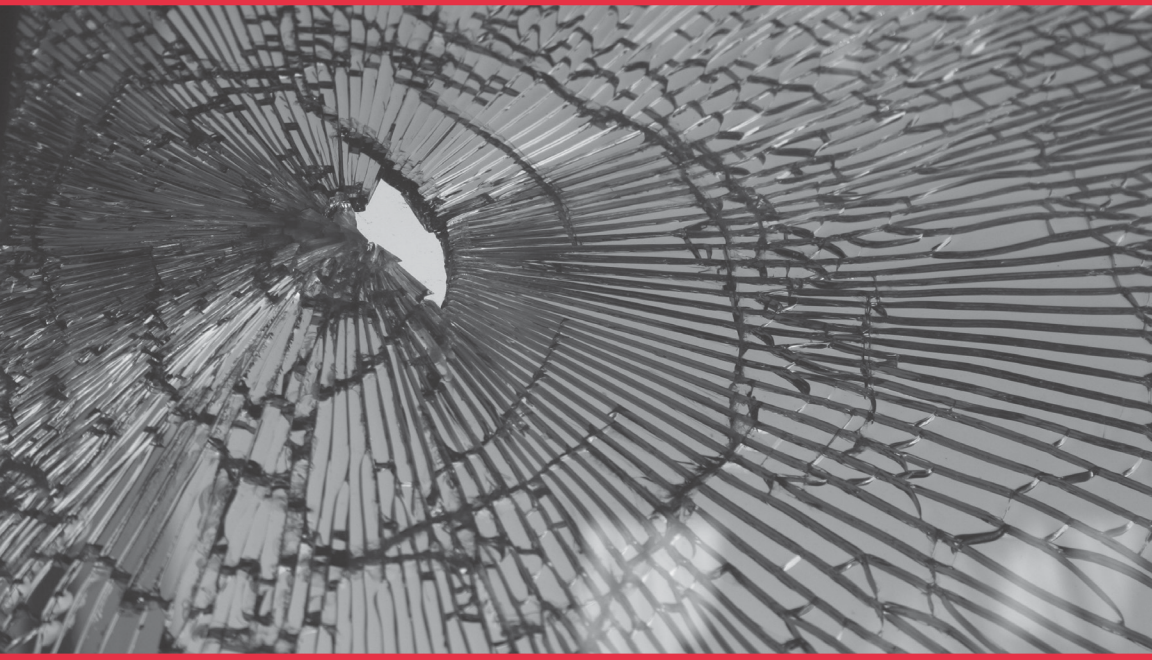
O'REILLY®

Compliments of ANACONDA Powered by Continuum Analytics

Breaking Data Science Open

How Open Data Science Is Eating the World



Michele Chambers, Christine Doig & Ian Stokes-Rees

Breaking Data Science Open

How Open Data Science Is Eating the World

Michele Chambers, Christine Doig, and Ian Stokes-Rees

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

Breaking Data Science Open
by Michele Chambers, Christine Doig, and Ian Stokes-Rees

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com/safari*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Editor: Tim McGovern
Production Editor: Nicholas Adams
Proofreader: Rachel Monaghan
Interior Designer: David Futato
Cover Designer: Randy Comer

February 2017: First Edition

Revision History for the First Edition
2017-02-15: First Release

978-1-491-97299-1

[LSI]

Table of Contents

Preface

Data science has captured the public's attention over the past few years as perhaps the hottest and most lucrative technology field. No longer just a buzzword for advanced analytical software, data science is poised to change everything about an organization: its potential customers, its expansion plans, its engineering and manufacturing process, how it chooses and interacts with suppliers, and more. The leading edge of this tsunami is a combination of innovative business and technology trends that promise a more intelligent future based on the pairing of open source software and cross-organizational collaboration called *Open Data Science*. Open Data Science is a movement that makes the open source tools of data science—data, analytics, and computation—work together as a connected ecosystem.

Open Data Science, as we'll explore in this report, is the combination—greater than the sum of its parts—of developments in software, hardware, and organizational culture. The ongoing consumerization of technology has brought open source to the forefront, creating a marketplace of ideas where innovation quickly emerges and is vetted by millions of demanding users worldwide. These users industrialize products faster than any commercial technology company could possibly accomplish. On top of this, the Agile trend fosters rapid experimentation and prototyping, which prompts modern data science teams to constantly generate and test new hypotheses, discarding many ideas and quickly arriving at the top 1 percent that can generate value and are worth pursuing. Agile has also led to the fusing of development and operations into DevOps, where the top ideas are quickly pushed into production deployment to reap value. All this lies against a background of ever-

growing data sources and data speeds ("Big Data"). This continuous cycle of innovation requires that modern data science teams utilize an evolving set of open source innovations to add higher levels of value without recreating the wheel.

This report discusses the evolution of data science and the technologies behind Open Data Science, including data science collaboration, self-service data science, and data science deployment. Because Open Data Science is composed of these many moving pieces, we'll discuss strategies and tools for making the technologies and people work together to realize their full potential. Continuum Analytics, the driving force behind Anaconda, the leading Open Data Science platform powered by Python, is the sponsor of this report.

CHAPTER 1

How Data Science Entered Everyday Business

Business intelligence (BI) has been evolving for decades as data has become cheaper, easier to access, and easier to share. BI analysts take historical data, perform queries, and summarize findings in static reports that often include charts. The outputs of business intelligence are "known knowns" that are manifested in stand-alone reports examined by a single business analyst or shared among a few managers.

Predictive analytics has been unfolding on a parallel track to business intelligence. With predictive analytics, numerous tools allow analysts to gain insight into "known unknowns," such as where their future competitors will come from. These tools track trends and make predictions, but are often limited to specialized programs designed for statisticians and mathematicians.

Data science is a multidisciplinary field that combines the latest innovations in advanced analytics, including machine learning and artificial intelligence, with high-performance computing and visualizations. The tools of data science originated in the scientific community, where researchers used them to test and verify hypotheses that include "unknown unknowns," and they have entered business, government, and other organizations gradually over the past decade as computing costs have shrunk and software has grown in sophistication. The finance industry was an early adopter of data science.

Now it is a mainstay of retail, city planning, political campaigns, and many other domains.

Data science is a significant breakthrough from traditional business intelligence and predictive analytics. It brings in data that is orders of magnitude larger than what previous generations of data warehouses could store, and it even works on streaming data sources. The analytical tools used in data science are also increasingly powerful, using artificial intelligence techniques to identify hidden patterns in data and pull new insights out of it. The visualization tools used in data science leverage modern web technologies to deliver interactive browser-based applications. Not only are these applications visually stunning, they also provide rich context and relevance to their consumers. Some of the changes driving the wider use of data science include:

The lure of Open Data Science
: Open source communities want to break free from the shackles of proprietary tools and embrace a more open and collaborative work style that reflects the way they work with their teams all over the world. These communities are not just creating new tools; they're calling on enterprises to use the right tools for the problem at hand. Increasingly, that's a wide array of programming languages, analytic techniques, analytic libraries, visualizations, and computing infrastructure. Popular tools for Open Data Science include the R programming language, which provides a wide range of statistical functionality, and Python, which is a quick-to-learn, fast prototyping language that can easily be integrated with existing systems and deployed into production. Both of these languages have thousands of analytics libraries that deliver everything from basic statistics to linear algebra, machine learning, deep learning, image and natural language processing, simulation, and genetic algorithms used to address complexity and uncertainty. Additionally, powerful visualization libraries range from basic plotting to fully interactive browser-based visualizations that scale to billions of points.

The gains in productivity from data science collaboration
: The very-sought-after unicorn data scientist who understands everything about algorithms, data collection, programming, and your business might exist, but more often it's the modern, collaborating data science teams that get the job done for enterprises. Modern data science teams are a composite of the skills

represented by the unicorn data scientist and work in multiple areas of a business. Their backgrounds cover a wide range of databases, statistics, programming, ETL (extract, transform, load), high-performance computing, Hadoop, machine learning, open source, subject matter expertise, business intelligence, and visualization. Data science collaboration tools facilitate workflows and interactions, typically based on an Agile methodology, so that work seamlessly flows between various team members. This highly interactive workflow helps teams progressively build and validate early-stage proof of concepts and prototypes, while moving toward production deployments.

The efficiencies of self-service data science
: While predictive analytics was relegated to the back office and developed by mathematicians, data science has empowered entire data science teams, including frontliners—often referred to as *citizen data scientists*—with intelligent applications and ubiquitous tools that are familiar to businesspeople and use spreadsheet- and browser-based interfaces. With these powerful applications and tools, citizen data scientists can now perform their own predictive analyses to make evidence-based predictions and decisions.

The increasing ease of data science deployment
: In the past, technology and cost barriers prevented predictive analytics from moving into production in many cases. Today, with Open Data Science, both of these barriers are significantly reduced, which has led to a rise in both producing new intelligent applications and intelligence embedded into devices and legacy applications.

What do the new data science capabilities mean for business users? Businesses are continually seeking competitive advantage, where there are a multitude of ways to use data and intelligence to underpin strategic, operational, and execution practices. Business users today, especially with millennials (comfortable with the open-ended capacities of Siri, Google Assistant, and Alexa) entering the workforce, expect an intelligent and personalized experience that can help them create value for their organization.

In short, data science drives innovation by arming everyone in an organization—from frontline employees to the board—with intelligence that connects the dots in data, bringing the power of new ana-

lytics to existing business applications and unleashing new intelligent applications. Data science can:

- Uncover totally unanticipated relationships and changes in markets or other patterns
- Help you change direction instantaneously
- Constantly adapt to changing data
- Handle streams of data—in fact, some embedded intelligent services make decisions and carry out those decisions automatically in microseconds

Data science enriches the value of data, going beyond what the data *says* to what it *means* for your organization—in other words, it turns raw data into intelligence that empowers everyone in your organization to discover new innovations, increase sales, and become more cost-efficient. Data science is not just about the algorithm, but about deriving value.

CHAPTER 2
Modern Data Science Teams

At its core, data science rests on mathematics, computer science, and subject matter expertise. A strong statistical background has traditionally been assumed necessary for one to work in data science. However, data science goes far beyond that, transforming expertise in statistics, data, and software development into a practical real-world discipline that solves a wide range of problems. Some of the additional skills required in a data science team include:

- Defining business needs and understanding what is of urgent interest to the business
- Determining what data is relevant to the organization and balancing the value of the data against the cost and risk of collecting and storing it
- Learning the tools to collect all kinds of data, ranging from social media to sensors, and doing the necessary initial cleaning, such as removing errors and duplicates
- Exploring data to develop an understanding of it and to discover patterns and identify anomalies
- Identifying the analytic techniques and models that will connect data sources to business needs
- Performing feature engineering to prepare the data for analysis, including data normalization, feature reduction, and feature generation
- Building, testing, and validating data science models

- Creating powerful visualizations to support the data science model narrative and make the analysis easy for end users to consume
- Using the data science model and visualization to build an intelligent application or embed the model into an existing application or device

Good statisticians are a hot commodity, and people who can do all the things just listed are even rarer. It is no surprise, then, that an urgent shortage of data scientists plagues multiple industries across the globe. Given the complexity and technical sophistication of the requirements, we can't expect individuals to enter the field quickly enough to meet the growing need—which includes any company that doesn't want to fall behind and see its business taken by a more data-savvy competitor.

We must form teams of people that embody all the necessary skills. For example, a 2013 article in *CIO* magazine points out that asking a technologist to think like a businessman, or vice versa, is rarely successful and that the two sides must know how to talk to each other. Being able to communicate with each other and combine their different skill sets makes the team more effective. To that end, modern data science teams typically include (Figure 2-1):

Business analysts
: Subject matter experts in the organization. Good at manipulating spreadsheets and drawing conclusions; used to exploring data through visualizations and understanding the business processes and objectives.

Data scientists
: Good at statistics, math, computer science and machine learning, perhaps natural language text processing, geospatial analytics, deep learning, and various other techniques.

Developers
: Knowledgeable in computer science and software engineering; responsible for incorporating the data scientists' models into applications, libraries, or programs that process the data and generate the final output as a report, intelligent application, or intelligent service.

Data engineers
: Responsible for building and maintaining the data pipelines and storage mechanisms used for organizational data.

DevOps engineers
: Shepherd models and programs from test environments to production environments, ensuring that production systems are running successfully and meet requirements.

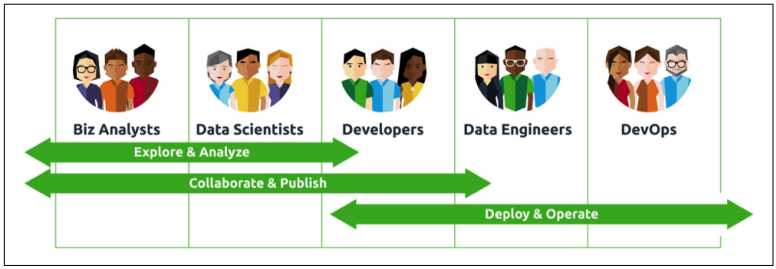


Figure 2-1. Participants in an organization's data science team

We'll explore how to get these team members engaged in activities that uncover key information your organization needs and facilitate their working together.

The pace of business today demands responsive data science collaboration from empowered teams with a deep understanding of the business that can quickly deliver value. As with Agile approaches, modern data science teams are being tasked to continuously deliver incremental business value. They know that they have to respond quickly to trends in the market, and they want tools that let them produce answers instantly. The new expectations match the experiences they're used to online, where they can find and order a meal or log their activities and share videos with friends within seconds. Increasingly, thanks to the incorporation of artificial intelligence into consumer apps, people also expect interfaces to adapt to their interests and show them what they want *in the moment*. With decreasing costs of computing, on-demand computation in the cloud, and new machine learning algorithms that take advantage of that computing power, data science can automate decisions that depend on complex factors and present other decisions in a manner that is easier for people to visualize.

CHAPTER 3
Data Science for All

Thanks to the promise of new insights for innovation and competitiveness from Big Data, data science has gone mainstream. Executives are spending billions of dollars collecting and storing data, and they are demanding return on their investment. Simply getting the data faster is of limited value, so they are seeking to use the data to enrich their day-to-day operations and get better visibility into the future.

Data science is the path to monetizing the mounds of data now available. But old-school tools are laden with technical hurdles and huge costs that don't align well with the the needs of Big Data analysis and aren't agile enough to keep up with the almost continuously evolving demands driven by changes in the Big Data stack and in the marketplace.

Enter Open Data Science. Open Data Science is a big tent that welcomes and connects many data science tools together into a coherent foundation that enables the modern data science team to solve today's most challenging problems. Open Data Science makes it easy for modern data science teams to use all data—big, small, or anything in between. Open Data Science also maximizes the plethora of computing technologies available, including multicore CPUs, GPUs, in-memory architectures, and clusters. Open Data Science takes advantage of a vast array of robust and reliable algorithms, plus the latest and most innovative algorithms available. This is why Open Data Science is being used to propel science, business, and society forward.

Take, for example, the recent discovery of gravitational waves by the Ligo project team, which utilizes Python, NumPy, SciPy, Matplotlib, and Jupyter Notebooks. And consider the DARPA Memex project, which crawls the web to uncover human trafficking rings using Anaconda, Nutch, Bokeh, Tika, and Elastic Search. Then there's the startup biotech firm Recursion Pharmaceuticals, which uses Anaconda and Bokeh in its mission to eradicate rare genetic diseases by discovering immune therapies from existing pharmaceutical shelved inventories.

What tools and practices enable Open Data Science and expand the number of new opportunities to apply data science to real-world problems? Open source software and a new emerging Open Data Science stack. In the next section, we'll dig into each of these further.

Open Source Software and Benefits of Open Data Science

At the heart of Open Data Science lies open source software with huge and vibrant communities. Open source provides a foundation where new contributors can build upon the work of the pioneers who came before them. As an open source community matures and grows, its momentum increases, since each new contributor can reuse the underlying work to build high-level software that makes it easier for more people to use and contribute to the effort. When open source software is made available, the software is tested by far greater numbers of users in a wider range of situations—not just the typical cases the original designers envisioned, but also the edge cases, which serve to quickly industrialize the software.

Open Data Science tools are created by a global community of analysts, engineers, statisticians, and computer scientists. They are written today in languages such as Java, C, Python, R, Scala, Java, and C, to name just a few. Higher-level languages, such as Python and R, can be used to wrap lower-level languages, such as C and Java. This global community includes millions of users and developers who rapidly iterate the design and implementation of the most exciting algorithms, visualization strategies, and data processing routines available today. These pieces can be scaled and deployed efficiently and economically to a wide range of systems. Traditional tools, typically from commercial software providers, evolve slowly. While there are advantages to this stability and predictability, these tools

are often architected around 1980s-style client-server models that don't scale to internet-oriented deployments with web-accessible interfaces. The Open Data Science ecosystem, on the other hand, is founded on concepts of standards, openness, web accessibility, and web-scale-oriented distributed computing.

Open source is an ideal partner to the fast-paced technology shifts occurring today. The success of any open source project is based on market demand and adoption. Let's take a look at the reasons that open source has become the underpinning of this new work model:

Availability
: There are thousands of open source projects, offering any number of tools and approaches that can be used to solve problems. Each open source project initially is known to only a small community that shares a common interest. The projects grow if they meet a market demand, linger on in obscurity, or simply wither away. But as adoption increases, the successful projects mature and the software is used to solve more problems. This gives everyone access to the accumulated experience of thousands of data scientists, developers, and end users. Open source software is therefore democratizing: it brings advanced software capabilities to residents of developing countries, students, and others who might not be able to afford the expensive and proprietary tools.

Robustness
: Because every alpha and beta release goes out to hundreds or thousands of knowledgeable users who try it out on real-world data sets and applications, errors tend to get ironed out before the first official release. Even when the tools are out in the field, someone usually steps up to fix a reported bug quickly; if the error is holding up your organization, your development team can fix it themselves. Open source software also guarantees continuity: you are not at the mercy of a vendor that may go out of business or discontinue support for a feature you depend on.

Innovation
: In the past, when a new algorithm was invented, its creators (typically academics) would present a paper about it at a conference. If the presentation was well received, they'd return the following year with an implementation of the algorithm and some findings to present to peers. By the time a vendor discovered the

new algorithm or there was enough customer demand for it, three to five years had passed since the development of the algorithm. Contrast that to today. The best and brightest minds in universities invent a new algorithm or approach; they collaborate, immediately open-source it, and start to build a community that provides feedback to help evolve the technology. The new algorithm finds its way into many more applications than initially intended, and in doing so, evolves faster. Users can choose from a plethora of algorithms and use them as is or adjust them to the particular requirements of the current problem. This is exactly what you see unfolding in many open source communities: Python, R, Java, Hadoop, Scala, Julia, and others. Because there are so many tools and they are so easy to exchange, new ideas fueled by the power of collective intelligence can be put into practice quickly. This experimentation and prototyping with near-instantaneous feedback spreads ideas and encourages other contributors to also deliver cutting-edge innovations.

Transparency

In proprietary tools, algorithms are opaque and change requests are subject to the pace of the vendor. Open source provides the information you need to determine whether algorithms are appropriate for your data and population. Thanks to decades of academic research, there is an abundance of Open Data Science tools that disclose algorithms and processing techniques to the public via open source, so that data scientists can ensure the technique is appropriate to solving the problem at hand. Additionally, data scientists can leverage open source algorithms and improve them to suit their problems and environments. This flexibility makes it easier and faster for the data science team to deliver higher-value solutions. With open source, data scientists no longer have to blindly trust a black-box algorithm. They can read the code of the algorithms they will be executing in production to make sure they are correctly implemented.

Responsiveness to user needs

Open source software was usually developed to scratch someone's own itch (to use a metaphor popularized by Eric Raymond in his book *The Cathedral & the Bazaar*) and is extended over time by its users. Some proprietary vendors, certainly, stay very attuned to their customers and can produce new features at fre-

quent intervals, but open source communities are uniquely able to shape software to meet all their users' requirements. Proprietary vendors bundle a plethora of features in one solution, while Open Data Science allows enterprises to pick and choose the features they need and build a custom solution.

Interoperability
: Open source communities pay attention to each other's work and converge quickly on simple data formats, so tying tools from different projects together is easy. In contrast, proprietary vendors deliberately seek incompatibility with competing products (most readers will remember one vendor's promise to "embrace and extend" some years ago, although that vendor is now working very well with open source communities) and their own formats tend to become complex over time as they strive for backward compatibility. Open source communities are also very practical and create bridges to legacy technology that allow organizations to redeploy these systems into modern architectures and applications, where necessary.

Efficient investment
: Open source projects do demand an initial investment of team time to evaluate the maturity of the software and community. Additionally, installing open source software can be challenging, especially when it comes to repeatability. But over time, it is much more cost-effective to run and maintain open source software than to pay the licensing fees for proprietary software.

Knowledgeable users
: Many programmers and other technical teams learn popular open source tools in college because they can easily learn via the endless online resources and freely download the software. This means they come to their first jobs already trained and can be productive with those tools the moment they first take their seats. It is harder to find expertise in proprietary products in students straight out of college. Moreover, many open source users are adept at navigating the communities and dealing with internals of the code.

In short, modern data science teams have many reasons to turn to open source software. It makes it easy for them to choose the right tool for the right job, to switch tools and libraries when it is useful to do so, and to staff their teams.

The Future of the Open Data Science Stack

Data is everywhere. There's more of it and it's messier. But it's just data. Everything around the data is changing—compute, hardware, software, analytics—while the structure and characteristics of the data itself are also changing.

For the last 30 years, programmers have basically lived in a monoculture of both hardware and software. A single CPU family, made by Intel and running the x86 instruction set, has been coupled with a single succession of operating systems: DOS, then Windows, and more recently, Linux. The design of software and business data systems started with this foundation. You placed your data in some kind of relational database, then hired a crew of software developers to write Java or .NET and maybe a roomful of business analysts who used Excel.

But those software developers didn't generally have to think about potentially scaling their business applications to multiple operating system instances. They certainly almost never concerned themselves with thinking about low-level hardware tradeoffs, like network latency and cache coherence. And for business applications in particular, almost no one was really tinkering with exotic options, like distributed computing and, yes, cache coherence.

This siloed monoculture has been disrupted. At the most fundamental level, computer processors and memory—two tectonic plates under all the software we rely upon for business data processing and analytics—are being fractured, deconstructed, and revolutionized, and all kinds of new technologies are emerging.

It is well known that the age of Moore's law, the steady increase in serial CPU performance, has come to an end. The semiconductor industry has reached the limits set by atomic size and quantum mechanics for how small they can make and how fast they can run a single transistor. So now, distributed and parallel computing are mainstream concepts that we have to get good at if we want to scale our computing performance. This is much, much more complex than simply swapping out a CPU with one that's twice as fast.

NVIDIA's latest generation of GPUs delivers five teraflops on a single chip. Depending on workload, that's roughly 100x faster than a vanilla Intel CPU. And people are buying racks of them—high-frequency traders, hedge funds, artificial intelligence startups, and

every large company with enough resources to put together an R&D team.

On the other end of the spectrum, Amazon and the other cloud vendors want us to stop thinking about individual computers and move to a new paradigm, where all computational resources are elastic and can be dynamically provisioned to suit the workload need. You want a thousand computers for the weekend? Click a button. This is a new way of thinking. Anyone who has had to deal with traditional IT departments can testify as to how long it takes to get a new data center set up with 1,000 computers and about 25 racks of 42 1U servers. How many years? Now you can do it almost instantly. We no longer think in terms of a physical "PC" or a "server." Instead, this has dissolved into a mere slider on a web page to indicate how many you want and for how long.

While the cloud vendors are abstracting away the computer, a raft of technologies are emerging to abstract away the operating system. The technology space around containers, virtualization, and orchestration is churning with activity right now, as people want to deconstruct and dissolve the concept of an "operating system" tied to a single computer. Instead, you orchestrate and manage an entire data center topology to suit your computational workload. So Windows? Linux? Who cares? It just needs an HTTP port.

And that's all just at the hardware and operating system level. If we go anywhere up the stack to applications, data storage, and so on, we find similar major paradigm shifts. You're probably intimately familiar with the technology hype and adoption cycle around Hadoop. That same phenomenon is playing out in many other areas: IoT, business application architecture, you name it.

What may prove to be the largest disruption yet is about to hit next year: a new kind of storage/memory device called 3D Xpoint. A persistent class of storage like disk or SSD, it's 100x faster than SSDs and almost as fast as RAM. It's 10x more dense than RAM. So instead of a 1 TB memory server, you'll have 10 TB of persistent memory.

To make this concrete: the new storage fundamentally changes how software is written, and even the purpose of an operating system has to be redefined. You never have to "quit" an application. All applications are running, all the time. There's no "save" button because everything is always saved.

The rate of fundamental technology innovation—and not just churn—is accelerating. This will hit data systems first because every aspect of how we ingest, store, manage, and compute business data will be disrupted.

This disruption will trigger the emergence of an entire new data science stack, one that eliminates components in the stack and blurs the lines of the old stack. Not only will the data science technology stack change, but costs will be driven down and the old-world proprietary vendors that didn't adapt to this new world order will finally tumble as well.

CHAPTER 4

Open Data Science Applications: Case Studies

Open Data Science has brought the ingredients of data science—data, analytics, and computation—within everyone's reach. This is fueling a new generation of intelligent applications that solve previously intractable problems and facilitate innovative discoveries. Here are a few case studies of Continuum Analytics' clients that showcase the power of Open Data Science.

Recursion Pharmaceuticals

This biotech startup found that the enormous size and complex interactions inherent in genomic material made it hard for biologists to find relationships that might predict diseases or optimize treatment. Through a sophisticated combination of analytics and visualization, Recursion's data scientists produced heat maps that compared diseased samples to healthy genetic material, highlighting differences. The biologists not only can identify disease markers more accurately and quickly, but can also run intelligent simulations that apply up to thousands of potential drug remedies to diseased cells to identify treatments.

This has greatly accelerated the treatment discovery process. Fueled by Open Data Science, Recursion Pharmaceuticals has been able to find treatments for rare genetic diseases—specifically, unanticipated uses for drugs already developed by their client pharmaceutical companies. The benefits to patients are incalculable, because treat-

ments for rare diseases don't provide the revenue potential to justify costly drug development. Furthermore, small samples of patients mean that conventional randomized drug trials can't produce statistically significant results and therefore the drugs might otherwise not be approved for sale.

TaxBrain

The Open Source Policy Center (OSPC) was formed to "open-source the government" by creating transparency around the models used to formulate policies. Until now, those models have been locked up in proprietary software. The OSPC created an open source community seeded by academics and economists. Using Open Data Science, this community translated the private economic models that sit behind policy decisions and made them publicly available as open source software. Citizen data scientists and journalists can access these today through the OSPC TaxBrain web interface, allowing anyone to predict the economic impact of tax policy changes.

Having represented the tax code in a calculable form, this team can now ask questions such as: what will be the result of increasing or decreasing a rate? How about a deduction? By putting their work on the web, the team allows anyone with sufficient knowledge to ask such questions and get instant results. People concerned with taxes (and who isn't?) can immediately show the effects of a change, instead of depending on the assurances of the Treasury Department or a handful of think-tank experts. This is not only an Open Data Science project, but an open data project (drawing from published laws) and an open source software project (the code was released on GitHub).

TaxBrain is a powerful departure from the typical data science project, where a team of data scientists creates models that are surfaced to end users via reports. Instead, TaxBrain was developed by subject matter experts who easily picked up Python and created powerful economic models that simulate the complexities of US tax code to predict future policy outcomes in an interactive visual interface.

Lawrence Berkeley National Laboratory/ University of Hamburg

In academia, scientists often collaborate on their research, and this is true of the physicists at the University of Hamburg. As with many scientists today, they fill a role as data scientists. Their research is quantified with data, and the reproducibility of their results is important for effective dissemination.

Vying for time on one of the world's most advanced plasma accelerators is highly competitive. The University of Hamburg group's research must be innovative and prove that their time on the accelerator will produce novel results that push the frontiers of scientific knowledge.

To this end, particle physicists from Lawrence Berkeley National Laboratory (LBNL) and the University of Hamburg worked together to create a new algorithm and approach, using cylindrical geometry, which they embedded in a simulator to identify the best experiments to run on the plasma accelerator. Even though the scientists are on separate continents, they were able to easily collaborate using Open Data Science tools, boosting their development productivity and allowing them to scale out complex simulations across a 128 GPU cluster, which resulted in a 50 percent speedup in performance. This cutting-edge simulation optimized their time on the plasma accelerator, allowing them to zero in on the most innovative research quickly.

As more businesses and researchers try to rapidly unlock the value of their data in modern architectures, Open Data Science becomes essential to their strategy.

CHAPTER 5
Data Science Executive Sponsorship

In the previous chapter, we've learned why Open Data Science matters. It has the potential to dramatically transform the way organizations pursue critical strategic initiatives, as well as the way they track and measure success.

These enabling characteristics of Open Data Science, coupled with its widespread grassroots adoption, have a downside. Individuals within an organization can now make technology decisions that do not have up-front costs and can therefore bypass the established review processes—covering technical suitability, strategic opportunity, and total cost—that have traditionally been required of proprietary technology. Even worse is that different individuals may make conflicting technology decisions that come to light only once projects have made significant progress. Thus, to ensure that the short- and long-term business outcomes of adoptixng Open Data Science are aligned with the company's strategic direction, executive sponsorship is essential.

This might sound like the typical IT-projects-need-executive-sponsorship soapbox. But keep in mind that we're talking about making room in the enterprise IT landscape for a new world where Open Data Science connects with new and existing data to inform everything from day-to-day micro decisions to occasional strategic macro decisions. Open Data Science introduces new risks that are mitigated by appropriate executive sponsorship:

De facto technology decisions
: Expedient decisions made on the basis of a technically capable individual being excited about some new technology can quickly become the de facto deployed technology for a project, group, or organization.

Open Data Science anarchy
: The risk that purely grassroots-driven Open Data Science heads off in too many different directions and escapes the organization's ability to manage or leverage Open Data Science at scale.

The attitude that Open Data Science has zero cost
: Although it's true that Open Data Science can reduce costs dramatically, supporting and maintaining the Open Data Science organization does have a cost to it, and this cost must be budgeted for.

The dynamic and agile approach to planning that Open Data Science implies also brings leadership challenges: executive review of an Open Data Science initiative would, in a traditional enterprise, typically happen during the budgeting stage. But it must take place apart from the budget approval process, with more consideration given to adoption costs down the road, rather than acquisition costs up front. When adopting Open Data Science, executives need to keep an eye on managing its strategic value, while considering how it aligns architecturally with existing and future systems and processes. Some of the key adoption costs to consider are integration, support, training, and customization.

By bringing Open Data Science into the enterprise, lines of business will need to work closely with the IT organization to guarantee that security, governance, and provenance requirements are still satisfied. For this to succeed, the executive sponsor needs to be involved in a different way. Executives will need to help shape and advocate for the right team structure and processes that are much more innovative and responsive. It is also important for the executive sponsor to set a tone appropriate to an Open Data Science environment, encouraging the use of a diverse mix of tools and technologies rather than "picking winners."

Dynamic, Not Static, Investments

With traditional analytics software, when you decided to purchase a platform or system from a vendor, you were effectively wedded to that decision for a considerable time. All the strategic decisions—and spending allocations—were made up front. And then you got what you got. Because of the size of the investment, you'd have to commit to this decision for the long haul. This static investment is quite different than the dynamic investments that are made with Open Data Science.

In the Open Data Science world, you'll have the advantage of moving more swiftly, and getting things up and running more quickly on the front end, as the open source software is freely available for people to download and start using right away. They don't have to wait for corporate purchasing cycles. Neither do they have to wait for the long upgrade cycles of commercial software products, as the the brightest minds around the world contribute to open source software innovation and development, and their efforts are made instantly available. That's a definite plus. Less up-front big planning and big budgeting is needed. But that's when things begin to differ from the traditional world. You have to continually make new choices and new investments, as your needs—and the technology—evolve. Thus, executives will need to stay engaged in order to manage this dynamic investment for the long run.

Executive sponsorship of Open Data Science initiatives serves two requirements—requirements that are sometimes at cross purposes. The executive's job is to balance the need to give the data science teams flexibility to pursue their endeavors with the need to impose IT controls and stability for the good of the business.

Let's now look at the different ways executives will need to exercise this balance for different types of Open Data Science functions.

Data Lab Environment

This essential Open Data Science element consists of tools for exploratory and ad hoc data science, data visualization, and collaborative model development. This small group of people is charged with actively seeking out open source technologies and determining the fit of these technologies for the enterprise. This group typically prototypes projects with the new technology to prove or disprove

the fit to both the business and technology leaders in the enterprise. If the technology proves to be valuable to the organization, then this team shepherds its adoption into the mainstream business processes. This often includes finding vendors that support the open source technology and can help the enterprise manage the use of open source.

Because this team needs the authority to experiment without being constrained by strict production environment requirements, the backing of the executive sponsor is essential to allow the team the freedom to experiment with a wide variety of open source technologies. This framework allows the data science team to work in a less constrained sandbox environment and be agile, while ensuring that what they do aligns with business operational requirements. The data lab team needs to be accountable to the executive sponsor and have negotiated clear objectives tied either to project outcomes or a calendar timeline.

Team Management, Processes, and Protocols

Executives should also oversee the people aspect of Open Data Science initiatives. We'll discuss the makeup of a successful Open Data Science team shortly, but on the managerial level it's vital to bear in mind the goal: successful data science happens in empowered, collaborative, cross-functional teams. The challenge for executive sponsorship is forming and supporting groups of diverse people to come together, collaborate, cross-pollinate, share ideas and skills, and work together productively. Executives must be able to establish an organizational system to support that. The team must have the collective skill set not just to do the exploratory work into the previously unknown, but to take those exploratory pieces and translate them into operational analytics environments. Some of these initiatives will end up where some parts are fully automated, generating results that any user—even one without analytics or statistical skills—can look at and understand. Then there's the large cadre of Excel and web app users to whom you need to provide operational direction. How do you equip and empower them, as opposed to disenfranchising and isolating them? Some strategies to consider are training opportunities, internal coaches or advisors, and multitiered support avenues.

Executive sponsors must impose accountability on the data science team. They need to maintain control over what the data science

team is doing, to ensure that every data science initiative is translatable to operational systems, resulting in more stable, established analysis systems. Open Data Science initiatives must also be aligned with business goals and key performance metrics to measure the impact they're having. Otherwise, the data science team could generate systems that might sound and look exciting and represent great innovation, but which have no relationship to web apps that would make them operationally useful. Thus, the only way to translate innovation into operationally realistic large-scale systems is with proper executive sponsorship and oversight.

Data Services and Data Lifecycle Management

In the Open Data Science world, executives need to consider the priorities and strategies for the management of both in-motion streaming data (network-based resources) and at-rest data sources (filesystem and database resources). Different approaches are required for each so that they are exposed and made accessible to the right people. Executives have a role to play in overseeing data services and data lifecycle management as it becomes a strategic capability of the organization. They must take ownership of the process, and make sure that it aligns with business needs. This is where a Chief Data Officer comes in: his or her primary role is to balance the business priorities to derive value from data with the IT priorities to reduce the risks and costs of managing that data. Appropriate executive sponsorship can ensure that IT provides open access to data services that will allow analysts to gain insight into the business. An orientation toward Open Data Science and appropriate access to corporate data sources will often reveal troves of existing data from which business value can be extracted.

Infrastructure and Infrastructure Operations

Think of all the storage, compute, and networking systems that make up the infrastructure, as well as the management of all these elements. In the Open Data Science world, this infrastructure is a living, breathing creature. It must be flexible and able to scale. A huge change in the new world of Open Data Science is having to move from traditional database systems to distributed filesystems like Hadoop/HDFS to be able to handle the exponential growth in data collection. Executives need to understand how Open Data Science initiatives require a different kind of infrastructure that is typi-

cally part of a larger IT ecosystem. This means identifying the systems you have in place, how they operate, and how you manage and maintain them in the face of constantly evolving tools and platforms in the Open Data Science world. Executives must also understand and take ownership of automated deployments—whether for real-time automated results, or jobs that are batched nightly.

The goal here is to create an environment where people, systems, and processes can support an Open Data Science environment, which will deliver flexibility and innovation in a way that would be much harder—or impossible—to achieve through traditional software.

Executive Sponsorship Responsibilities

Executives sponsoring a modern Enterprise Data Science environment need to oversee three important areas: governance, provenance, and reproducibility. These responsibilities span multiple parts of the organization, but in the case of data science teams, it is imperative that executives understand their duties when it comes to Open Data Science initiatives.

Governance

The executive sponsors first and foremost need to establish principles related to data sources: privacy, security, model "ownership," and any regulatory compliance that may be necessary for the organization's domain of operations. Governance oversight involves security (are the technical mechanisms in place to protect our sensitive data?) and managing errors and mistakes (what happened and how can we ensure it doesn't happen again?).

For example, what happens if you discover that your data analysis pipeline has been compromised? In an Open Data Science world, you have many components that work flexibly together. Do you know your exposure to that total risk? This is a key area for corporate governance. How do you create smart policies about how your data analysis is done? Assertion of the governance model is a purely human process. How do you create the data science infrastructure itself to ensure governance? Executives need policies to make sure the system is monitored and maintained securely.

A managed approach to Open Data Science adoption will mean there are clear mechanisms, either procedural or automated, to control and track the utilization of Open Data Science tools and environments within the organization.

Provenance

Provenance is another essential piece of Open Data Science that requires executive sponsorship. Provenance is the traceability of chronological or historical events to ensure that we know where data comes from, how it's been transformed, who has had access to it, and what kind of access they've had. Executives need assurance that every element of reported analyses—whether metrics, computed data sets, or visualizations—can be tracked back to original data sources and specific computational models. Provenance also covers "chain of custody" for analytics artifacts, identifying who has been involved in processing data or creating analytical models. Aggregating quantitative information, creating analytical routines, developing models of systems, and then coming up with quantitative results is a complex process, generating its own set of data. This internal metadata must be kept organized and accessible to the executives in charge of your data science initiatives.

This is especially critical for organizations that are under external regulatory constraints to track generated data sets and quantitative information: How did you end up with these conclusions or observations? The history of the data, plus who was involved, and all the transformation steps along the way, must always be clear.

Tracking provenance isn't limited to being able to track each "upstream" stage in an analytical process. It's not hard to envision a scenario where the ability to track the decisions that arise "downstream" from a data set or model is important: you might discover, for example, an analytical model that is generating systematic errors. For some time, its output data has been corrupted, so all downstream results that utilize that generated data set are suspect. You need to know where it was used, and by whom. Provenance requires you to identify the scope of the impact, so you can address the fallout.

Reproducibility

Executive sponsors should also prescribe the degree to which data science artifacts can be recreated. This encompasses issues of archiving source data, recording any data transformations, identifying the key software components, and documenting any analytical models or report-generating routines.

Reproducibility requires more than just provenance information, as just knowing where data came from and who did something to it doesn't necessarily mean you can reproduce it. It's not just a matter of recording what happened, but being able to go back to an exact "state." For example, without a timestamped record in a database, you can't go back and get the exact data that was used on a model yesterday at 7 pm.

CHAPTER 6

The Journey to Open Data Science

Organizations around the world, both small and large, are embarking on the journey to realize the benefits of Open Data Science. To succeed, they need to establish the right team, use the right technology to achieve their goals, and reduce migration risks. For most organizations, this journey removes barriers between departments as teams start to actively engage across the company and shift from incremental change to bold market moves.

The journey to Open Data Science is being forged with new practices that accelerate the time to value for organizations. In the past, much of the analysis has resulted in reports that delivered insights but required a human in the loop to review and take action on those insights. Today organizations are looking to directly empower frontliners and embed intelligence into the devices and operational processes so that the action happens automatically and instantaneously rather than as an afterthought. Adopting an Open Data Science approach is different from merely adopting a technology, however.

Moving to any new technology has an impact on your team, IT infrastructure, development process, and workload. Because of this, proper planning is essential. The drivers for change are different in every organization, so the speed and approach to the transition will also vary.

Team

Shifting to an Open Data Science paradigm requires changes. Successful projects begin with people, and Open Data Science is no different. New organizational structures—centers of excellence, lab teams, or emerging technology teams—are a way to dedicate personnel to jump-start the changes. These groups are typically charged with actively seeking out new Open Data Science technologies and determining the fit and value to the organization. This facilitates adoption of Open Data Science and bridges the gap between traditional IT and lines of business. Additionally, roles may shift—from statistician to data scientist and from database administrator to data engineer—and new roles, such as computational scientist, will emerge.

With these changes, the team will need additional training to become proficient in Open Data Science tools. While instructor-led training is still the norm, there are also many online learning opportunities where the team can self-teach using Open Data Science tools. With Open Data Science, recruiting knowledgeable resources is much easier across disciplines—scientists, mathematicians, engineers, business, and more—as open source is the de facto approach used in most universities worldwide. This results in a new generation of talent that can be brought onboard for data science projects.

Whether trained at university or on the job, the data science team needs the ability to integrate multiple tools into their workflow quickly and easily, in order to be effective and highly productive. Most of the skills-ready university graduates are very familiar with collaborating with colleagues across geographies in their university experience. Many are also familiar with Notebooks, an Open Data Science tool that facilitates the sharing of code, data, narratives, and visualizations. This familiarity is critical because data science collaboration is crucial to its success.

Research shows that the highest indicator of success for data scientists is curiosity. Open Data Science satisfies their curiosity and make them happy, as they are constantly learning new and innovative ways to deliver data science solutions. Moving to Open Data Science increases morale, as data scientists get to build on the shoulders of giants who created the foundation for modern analytics. They feel empowered by being able to use their choice of tools, algorithms, and compute environments to get the job done in a pro-

ductive and impactful way that satisfies their natural curiosity and desire to make meaningful changes with their work.

Technology

Selecting technology with Open Data Science is significantly easier than with proprietary software, because the software is freely available for download. This allows the data science team to self-serve their own proof of concept, trying out the Open Data Science technology required to meet the specific needs of their organization. For data science, there is no shortage of choices. Open source languages such as Python, R, Scala, and Julia are frontrunners in Open Data Science, and each of these languages in turn offers many different open source libraries for data analysis, mathematics, and data presentation, such as NumPy, SciPy, Pandas, and Matplotlib, available at no cost and with open source licensing. No matter what your data science goals are, there will be an open source project that meets your needs.

Some open source software only works effectively on local client machines, while other open source software supports scale-out architectures, such as Hadoop. Typically, a commercial vendor fills the gap on supporting a wider variety of modern architectures.

Migration

A migration strategy to Open Data Science should align with the business objectives and risk tolerance of the organization. It is not necessary to commit to fully recoding old analytic methods into some Open Data Science framework from the start. There is a range of strategies from completely risk-averse (do nothing) to higher risk (recode), each with its own pros and cons.

A coexistence strategy is fairly risk-averse and allows the team to learn the new technology, typically on greenfield projects, while keeping legacy technology in place. This minimizes disruption while the data science team becomes familiar and comfortable with Open Data Science tools. The "open" in Open Data Science often means there are existing strategies to integrate aspects of the legacy system either at a services or data layer. Existing projects can then migrate to Open Data Science when they reach the limits of the proprietary technology. The team then phases out the proprietary technologies

over time—for example, using a continuous integration and delivery methodology—so the Open Data Science capabilities slowly subsume the legacy system's capabilities. The introduction of Big Data projects has become an ideal scenario for many companies using a coexistence strategy; they leave legacy environments as is and use Open Data Science on Hadoop for their Big Data projects.

A migration strategy is slightly riskier and moves existing solutions into Open Data Science by reproducing the solution as is with any and all limitations. This is often accomplished by outsourcing the migration to a knowledgeable third party who is proficient in the proprietary technology as well as Open Data Science. A migration strategy can take place over time by targeting low-risk projects with limited scope, until all existing code has been migrated to the organization's Open Data Science environment. Migration strategies can also migrate all the legacy code via a "big bang" cutover. The data science solutions are improved to remove the legacy limitations over time.

A recoding strategy is higher risk and takes advantage of the entire modern analytics stack to reduce cost, streamline code efficiency, decrease maintenance, and create higher-impact business value more frequently, through better performance or from adding new data to drive better results and value. The objective of recoding is to remove limitations and constraints of legacy code by utilizing the advanced capabilities offered by Open Data Science on modern compute infrastructure. With this strategy, the organization often completes a full risk assessment—which includes estimates for cost reduction and improved results—to determine the prioritization of projects for recoding.

CHAPTER 7
The Open Data Science Landscape

Now that you have a sense of what Open Data Science is, and how to prepare culturally and organizationally for it, it's time to talk about the technology tools available in the Open Data Science world.

The Open Data Science community has grown to match and outperform traditional commercial tools for statistics and predictive analytics. In fact, Open Data Science tools are rapidly becoming the tools of choice for data scientists, with Python and R as the primary languages. They are part of an incredibly rich ecosystem with innumerable additional resources in the open source world that go well beyond the capabilities offered by commercial software. In the Big Data space, it is clear that Open Data Science technologies such as Hadoop, Spark, MongoDB, and Elastic Search are frequently preferred over commercial alternatives—and not simply due to the price differential, but because they offer the most powerful and capable enterprise-ready technology today for the problems they address.

Because of the scale and self-managed/anarchical structure of open source communities, the Open Data Science community can also seem chaotic from the outside. That's why many organizations have adopted open source distributions backed by companies that provide compatibility and enterprise guarantees. You need to bring order into that chaos, so you can leverage the diverse array of languages, packages, and tools within your company, and have them work in your environment.

While a continually changing ecosystem may appear unwieldy, the Open Data Science community has developed different approaches to software packaging and delivery to address the challenges of rapid uncoordinated innovation. The Open Data Science landscape, just like the Big Data landscape, offers a myriad of choices at every level in the stack. These tools can be used individually or interoperably, as the Open Data Science ecosystem follows design patterns that encourage easy interfacing with data and services. This is in contrast to legacy systems, which were built explicitly to be walled gardens with proprietary interfaces and data formats. By recognizing that data science is a complex world that uses many different approaches, the Open Data Science ecosystem has evolved a rich set of tools that address specific needs. Rather than pushing a static, slow-changing monolithic proprietary application, the Open Data Science community is continuously evolving the technology landscape in order to adopt the latest innovations and meet the changing needs of data science teams worldwide.

What the Open Data Science Community Can Do for You

With Open Data Science, data scientists can:

Access the very latest innovations in the industry
: A lot of the latest innovations, like Hadoop, came from large companies like Yahoo, Facebook, and Google. These companies developed tools for their own use, then gave back to the community by open-sourcing their technology. By adopting Open Data Science, you can get innovation almost directly from key players in major markets. For example, TensorFlow is an open source package developed by Google that has become a major deep learning artificial intelligence (AI) technology used by data scientists. If you use only commercial tools, you would quickly fall behind the competition.

Participate in a vibrant community
: You can report bugs, solve issues, and contribute to solutions, all while getting direct access to the actual developers of the tools you use. This gives you the type of access—and opportunity to provide input that can help your company—that is unheard of in commercial analytics tool circles.

Teach yourself
: No need to pay unaffordable licenses to learn a new technology—you can learn for free. With most commercial analytical software, you can learn the necessary skills only through premium licensing, support, and training contracts. Educators are realizing the benefits of building their courses around Open Data Science, as it increases the accessibility of their material and portability of the skills their students gain. Members of the Open Data Science community are motivated by mass adoption and therefore provide documentation online, run community support forums, and provide video walkthroughs, demos, and training. This lowers the barriers to entry for everyone.

Use the right tool for the job
: Data analysis, visualization, sharing, storage—these all require different tools (a complexity that multiplies when you consider the range of data types that an enterprise deals with). In the Open Data Science world you don't have to make a one-size-fits-all decision. You can choose to use R *and* Python *and* Scala. You can use Tensorflow *and* Theano *and* Scikit-learn. You use the best tool for the problem you currently have. This is a marked contrast to proprietary technologies, which often tend to be insular and promote tools from that community. Despite the many articles about "R vs. Python," and the like, the reality is that data scientists need choices and flexibility.

Open Data Science brings all the following domains together to solve the world's data challenges with the most innovative software in each field (see Figure 7-1):

- Statistics, machine learning, optimization, artificial intelligence
- Big Data and high-performance computing
- Data storage, including data warehousing, RDBMS, and NoSQL storage technologies
- Business analytics and intelligence
- Notebooks, integrated development environments, analytic pipeline workflows
- Plotting and visualization
- Web technologies
- Extract, transform, and load (ETL) pipelines

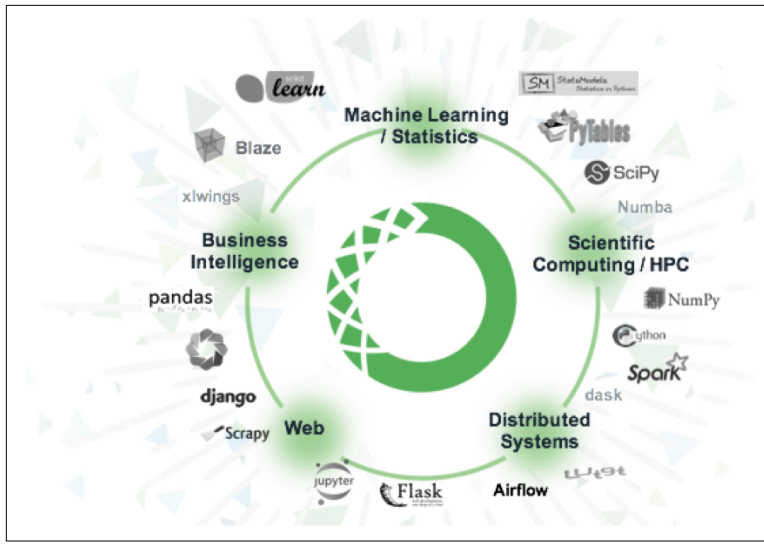


Figure 7-1. Open Data Science combines innovative software from many domains to address the world's data challenges

The Power of Open Data Science Languages

R, Python, Julia, and Scala are the most popular of the Open Data Science languages. Here's an overview of what they do and why they are best of breed in the field.

Why R?

R is a programming language and software environment for statistical computing and graphics, supported by the R Foundation for Statistical Computing. R offers a lot of statistical models, and many statisticians have written their apps in R. It has been the historical leader in open source statistical analysis, and there is a clear concentration of statistical models that have been written using R. The public R package archive, CRAN, contains over 8,000 community contributed packages. Microsoft, RStudio, and other companies provide commercial backing for R-based computing.

Why Python?

Python is a popular high-level, general-purpose, dynamic programming language that is commonly referred to as the easiest language to read and to learn. It is emerging as the leading language for Open

Data Science because it combines rapid development with the ability to easily interface with high-performance algorithms written in C or Fortran (see Figure 7-2). As a result, there is a rich suite of vectorized numerical libraries that can be used as a foundation for data science. Python's syntax allows programmers to express concepts clearly and concisely compared with languages such as C++ or Java. The fact that many web applications and most data science applications are now written in Python lowers the adoption barriers. Previously different parts of an organization, such as statistics, data, and engineering teams, used different languages and had to either rewrite code to share it among themselves, or deal with decreased performance because of impedance mismatch when wrapping functionality from one language to the next. They can now have a shared language to communicate with each other. Integration becomes much easier and lowers barriers to understanding between different teams, even those with different skill sets and roles.

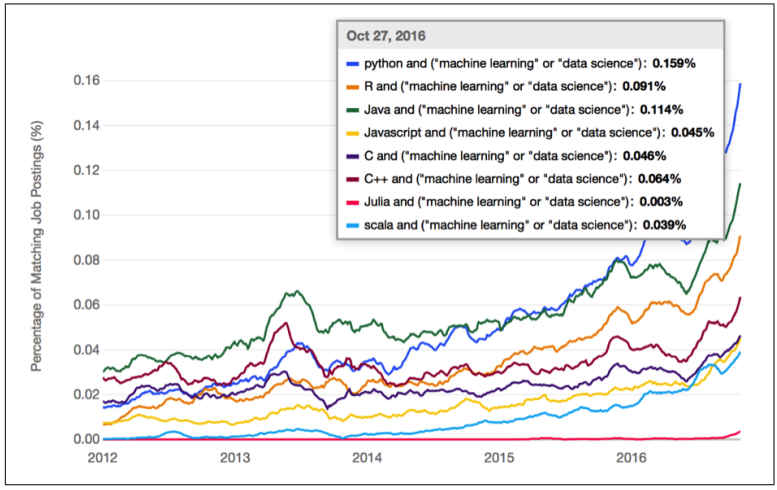


Figure 7-2. Python has emerged as the most popular programming language for data science (Source: KDnuggets)

Why Scala?

Scala, a syllabic abbreviation for "scalable language," is a general-purpose programming language. Scala has full support for functional programming and a strong static type system. Like Java, Scala is object-oriented and runs on top of the Java JVM. Because of this Java connection, it has been adopted by the Big Data Hadoop eco-

system. It has also been popularized by Spark, which is implemented in Scala. Currently it lacks the broad spectrum of functionality and supporting data science libraries that are available in Python and R.

Why Julia?

Julia is a high-level dynamic programming language designed to address the requirements of high-performance numerical and scientific computing while also being effective for general-purpose programming. Julia is a newer language, and hasn't been around as long as Python. It is suitable for innovators and early adopters who are looking for the highest performance parallel computing language focused on numerical algorithms.

Established Open Data Science Technologies

Notebooks and Narratives

One key paradigm that has evolved in the Open Data Science space is that of "notebook"-centric computing and analytics. The most popular notebook technologies today are Jupyter (supporting over 40 languages, though most heavily used by the Python data science community), RStudio (exclusively for R-based data science), and Apache Zeppelin (for the Java and Scala Hadoop data science community). Notebooks allow data scientists to create and share documents that contain runnable code, data, equations, visualizations, and explanatory text. This aspect of Open Data Science is quite popular: the Jupyter project alone (see Figure 7-3) estimates its user base at over 3 million people. The style of these notebooks and their ability to "tell a data science story" make them powerful collaboration and publishing tools. They are a 21st-century version of an analytics script, augmented with hypertext and inlining of the output—whether that is a data table, a graphic, or an interactive data visualization. Some notebook interfaces also offer presentation modes that allow users to directly utilize their notebook-based analysis in a slide-show mode, viewable through a web browser. These notebooks are self-contained and therefore portable: they can be emailed to colleagues or published onto websites where the content can be viewed directly. It's all about explainability, reproducibility, and interactivity.

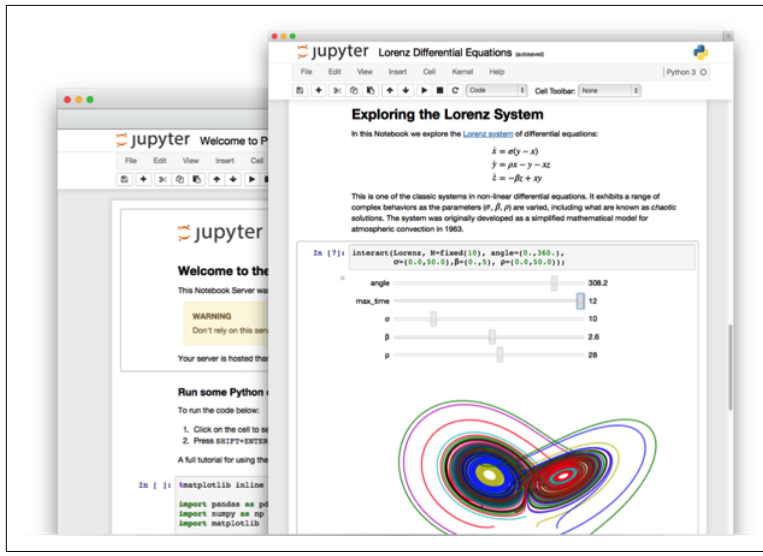


Figure 7-3. The Jupyter project

Hadoop/Spark

Hadoop is an open source, Java-based programming framework that supports the processing and storage of extremely large data sets in a distributed computing environment. Initially developed at Yahoo, based on Google research, Hadoop is now sponsored by the Apache Software Foundation. Apache Spark is an open source *parallel processing* framework for running large-scale applications across *clustered* computers by providing distributed in-memory storage and a table-like data structure, developed and supported by Databricks. Both Hadoop and Spark are Big Data technologies that have allowed data scientists to analyze ever-larger amounts of data, making new and bigger data sources accessible to data scientists—and both are excellent examples of the interplay between open source and commercial sponsorship.

Anaconda

Anaconda is the leading Open Data Science platform and empowers data science teams to deliver compelling data science applications. It is Python-centric but also has rich support for R. Anaconda makes it easy for data scientists and IT to use and govern open source packages that include the latest innovations in machine learning, deep

learning, and AI. The entire data science team can can easily collaborate, publish, and deploy data science applications. Business analysts can self-service powerful analytics from the familiar Microsoft Excel application to meet their on-demand and ad hoc analysis needs. There have been 11 million downloads of the single-user Anaconda distribution up to the end of 2016. Its popularity is due to it providing a turn-key environment for all platforms that delivers over 720 of the most popular Open Data Science packages in an open format that is customizable and extensible.

H2O

H2O provides a Java-based distributed AI platform with high-performance, low-level libraries including Python and R bindings. Higher-level analyst-oriented tools, such as the Flow web GUI, allow users to construct AI workflows in a notebook-like environment, while DevOps teams can utilize Steam to deploy AI models and workflows to a cluster. H2O also integrates with Spark for data access.

Pandas

Pandas is viewed as a Swiss Army knife for data processing and analysis. It provides a rich tabular data structure with an intuitive Python interface and can connect—for input or output—to a wide range of data sources, such as relational databases, JSON files, Excel, CSV, or binary data formats. It is built on top of the established NumPy library, which implements efficient vectorized operations entirely in C.

Scikit-Learn

Scikit-Learn is the premier Python machine learning library. It provides dozens of machine learning algorithms and supporting functionality, all utilizing a common interface for model training, prediction, and scoring. There is a large community of contributors who are both improving the existing methods and incorporating/implementing new machine learning strategies into the library.

Caret

Caret is the R analog to Scikit-Learn and has over 200 R-accessible classification and modeling algorithms bundled into a common interface. It is flexible and extensible.

Shiny

Shiny is an R package that is widely used to create simple web-based interactive data apps. It has a large following due to its ease of use and the availability of both a cloud-hosting facility for those apps and a Shiny server where they can be deployed.

Dask

Dask is a data science framework used to easily parallelize algorithms and data science workloads. It takes advantage of available memory and computer power to maximize use of memory, minimize execution time, and optimize performance of complex algorithms. Dask creates a task graph based on the data and then intelligently schedules the execution of the tasks to optimize throughput. Traditional data science Python libraries, mostly targeted for a single core, can manage a smaller amount of data than what Big Data technologies, like Spark, can. Dask allows you to extend those libraries without having to leave the Python ecosystem, so data scientists can manage larger amounts of data using the language and libraries they're most familiar with.

Emerging Open Data Science Technologies: Encapsulation with Docker and Conda

Containerization technology has revolutionized how developers build, ship, and deploy applications. Docker, at the forefront of the revolution, has enabled developers to run applications on any suitable physical machine without any worries about dependencies. Its wide adoption and growing ecosystem of tools around managing containers and scaling them has widely changed how deployment is understood. Docker has provided a specialized case of virtualization that is much lighter weight, focusing on rapid provisioning and simplicity in specification.

While Docker-style containerization has primarily been adopted by DevOps teams and software engineers, the concepts of encapsulation have also proved valuable for data acientists. The fast release cycles of many Open Data Science libraries and their interdependencies can make it hard to control the analytics and development environment, and containerization is one solution to that challenge. This challenge is exacerbated by the production deployment of models, workflows, and projects into large-scale automated environments.

Most data scientists develop in one environment (OS X, Windows, or Linux), but they might be working on multiple projects that require different versions of libraries. For example, they might be able to use the latest Pandas version in their new project, but have a year-old version of Pandas for a legacy project that needs a quick fix. While containers can be one way to support both of these configurations, that burdens a data scientist with, effectively, maintaining two distinct computers on their laptop or workstation and having to become their own system administrator, handling virtual network drives, filesystem mounting, and data management between these containers. An alternative to this strategy is using Conda environments, which allow data scientists to easily isolate each of those projects in their developing system of choice, and to switch from one to another, while avoiding the "distinct machines" presented by containers. Conda leverages capabilities already built in to modern operating systems to sandbox software and processes, allowing multiple versions of the same libraries or software to be installed at the same time (Figure 7-4).

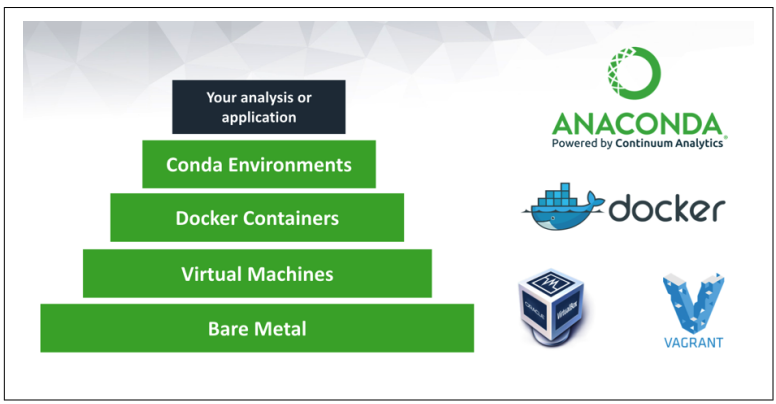


Figure 7-4. Encapsulation layers for modern data science applications

Open Source on the Rise

Open Data Science tools are becoming data scientists' tools of choice. The rich ecosystems that are growing up around these tools put them well beyond the capabilities offered by commercial software when it comes to creating models for analyzing data. Open Data Science languages such as R, Python, Julia, and Scala, along with the rich supporting analytics and tools, are making data science a force for good within today's enterprises.

CHAPTER 8
Data Science in the Enterprise

While we've discussed the challenges and rewards of Open Data Science in general, we'll now turn our attention to the specific needs of enterprise users. Enterprises are turning to new rich data sources to augment their data science efforts, and are using advanced techniques (including machine learning and artificial intelligence) to increase the accuracy and precision of their work. The combination of enterprise-scale data sources and the technology to make the most of them means their results can be more robust and drive new business value. One such avenue is personalization, which demands not just batch-processed aggregated market analysis but real-time individualized results. To truly understand the customer, product, market, employees, network, and systems, enterprises are using the most granular level of data. From customer clicks and transactions, to individual DNA data and facial data, to geolocation and sensors, the list of new data sources appears unending. This data is being used with the latest innovative and advanced analytics techniques available—typically open-sourced immediately by academic researchers worldwide—by teams that collaborate across groups, departments, and geographies to create better models that yield significantly better results. And perhaps most exciting is that the tools your teams can leverage today are simultaneously being improved upon and released by researchers, academics, and members of the Open Data Science community such that your organization can have even better results tomorrow.

What we are observing is a shift away from legacy analytics strategies, with their one-off and ad hoc analysis, toward modern enter-

prises, which demand reproducible production workloads that can be embedded into the applications used by frontline workers to do their jobs.

Despite the multitude of benefits, enterprises are still having difficulty taming the anarchy of Open Data Science since their internal processes were not established to deal with continuous change. So how do enterprises harness Open Data Science so they can realize these benefits?

How to Bring Open Data Science to the Enterprise

The bar for Open Data Science in an enterprise context is different than in scientific or academic arenas. In the enterprise there are regulations and established "best practices" to reduce risk and help streamline operations. Additionally, the enterprise context imposes complex infrastructure operations that need to coexist with any new technology. Making Open Data Science work in the enterprise requires additional capabilities that are not typical in the Open Data Science community. This is an area where Open Data Science vendors add value by creating the capabilities that allow enterprises to tame the open source anarchy and support the enterprise requirements for:

- Governance
- Collaboration
- Operations
- Big Data
- Machine learning and artificial intelligence
- Interactive dashboards and apps
- Self-service analytics

Governance

Enterprises require control over the creation, sharing, and deployment of data science assets. The authentication or permissions established for data science assets must integrate with a wide variety of enterprise authentication systems, such as LDAP, Active Directory, and Kerberos to govern and track all the Open Data Science activities. This includes access to specific versions of open source

libraries and packages, plus specific versions of the data science assets created by your team.

Additionally, a full history or the provenance of the data science assets (e.g., data, models, and apps) must be tracked in order to create the transparency often demanded by regulators or compliance review boards.

Collaboration

Data science teams use different methodologies to collaborate based on their education, experience, and training. In the Open Data Science era an Agile approach is typically adopted, allowing the enterprise's globally distributed teams to harness their collective intelligence to deliver the most comprehensive analytics assets in the shortest time. With notebooks, individuals can explore and plot their data and create data science models with interactive visualizations that easily be shared. However, enterprises require that these assets be governed by the same enterprise authentication that controls access to other enterprise assets including those for data science. Plus, enterprise teams need to collaborate in real time, which requires *collaborative locking*, a feature that allows multiple people to work on a Notebook without overwriting each other's work.

For the benefits of governance and effective enterprise-oriented collaboration, these data science assets are centrally housed so that ideas, best practices, and new innovations can be easily discovered and shared among the data science team. Publishing data science assets into a common repository increases the collective intelligence of the entire enterprise data science team thanks to reuse that builds upon previous work.

Operations

Gone are the days of analytics being relegated to a dark corner in the office. Open Data Science has brought data science out into the light, and enterprises expect that their models will be deployed into a production environment and used by frontline workers day in and day out to make evidence-based decisions that drive value. Whereas in the past analytics teams managed their own offline analysis and provided static reports and results, today's modern data-centric enterprise requires analytical and predictive models to be rapidly deployed, refined, and continuously monitored. Thus, the figurative

Berlin Wall that has existed between analytics teams and the operational IT systems must be destroyed in order for the enterprise to reap the dividends that modern analytics can generate. Open Data Science is the catalyst for tearing down that wall, as it removes both the technical and cost barriers that blocked data science from moving into production. However, enterprises also require reproducibility to ensure consistent and maintainable data science applications. Reproducibility, or the ability to easily replicate data science assets, includes being able to:

- save data science work—packages, notebooks, and environments—under different versions for traceability and lineage;
- create packages for custom algorithms, including all dependencies;
- encapsulate data science projects and environments; and
- migrate data science assets and environments from development to staging to production.

Once the data science application is ready to move into production, enterprises demand the flexibility to execute or run it on premises or in a private or public cloud. While the application may initially be launched in a public cloud, enterprises need the flexibility to bring the application on premises if regulations or business requirements change. Conversely, they also need to be able to take on-premises applications and easily move them to the cloud.

With today's modern infrastructure, deployment often means running the applications scaled out to tens or hundreds of servers in order to accommodate large volumes of data or intense computational demands. Usually this is on a distributed cluster running Spark or a high-performance operating system such as CentOS under a multitude of resource schedulers, including Yarn, Mesos, or Slurm. While many of these tools are open source, enterprises typically use a combination of proprietary and open source tools to support their clusters, requiring additional layers of interoperability beyond what the base open source libraries provide. This is why enterprises are turning to Open Data Science: it is an ecosystem that enables both open source and proprietary tools to interoperate seamlessly so enterprises can focus their efforts on building high-value applications. These tools capture and automate expertise that previously required human-in-the-loop processes or offline analysis. Such high-performance real-time automated analytics creates value

for the enterprise without the constraints of proprietary or open source software.

While Spark and other distributed systems are adept at managing simple workloads, epitomized by the MapReduce paradigm, they are not designed for numeric computing and advanced analytics. Their original design was optimized for the bulk data ingest and querying common in data engineering and business analytics.

Enterprises familiar with high-performance computing are very aware of the huge amount of effort typically needed to divide and conquer these more complex workloads, which often drive significant value for their enterprise. Until recently the ability to perform these kinds of large-scale complex analyses was possible only with proprietary data management and analytics tools. Enterprises have been actively seeking Open Data Science solutions to deliver easier-to-implement parallel computing frameworks for data science workloads. Dask is an emerging data science parallel computing framework that is quickly being adopted by leading hedge funds and financial institutions. Its relative simplicity and interoperability are exactly what enterprises have been seeking to scale their data science workloads to Big Data.

Open Data Science has enabled the shift of data science workloads from ad hoc usage into day-to-day production workloads. With that shift comes a need for enterprises to manage the dozens and hundreds of data science models that can now easily be deployed into production environments. In addition, enterprises require full data science lifecycle management to continuously reevaluate the efficacy of the data science models as changing business conditions impact both the available and required data assets.

Big Data

While Big Data is a revolution that was led by open source, open source falls short of the enterprise's needs for governance and operations support for their Big Data environments. While much of the Big Data revolution was fueled by businesses' need to reduce storage costs, it has matured significantly, and now enterprises are seeking to derive business value from their Big Data by leveraging data science to drive a new generation of applications. The diversity of workloads is astounding. Enterprises are looking to:

- query and explore their Big Data with interactive querying and visualization tools;
- cleanse and transform their Big Data to develop a better understanding of their business and to prepare for further analysis; and
- process a wider variety of data from both internal and external data sources ranging from data at rest to data in motion.

In order to support this variety of Big Data processing, Open Data Science has connectors not only to Big Data storage (such as HDFS and JSON object stores) but also to traditional data warehouses, streaming data sources, geospatial data, image data, and many others. As the aperture for data increases so, too, do the computational processing requirements in order to handle more complex and intensive workloads. Open Data Science delivers multiple capabilities to easily process the vast variety of workloads and allow enterprises to harness their hardware to scale up and out to meet their SLAs (service-level agreements).

Machine Learning and Artificial Intelligence

Enterprises have been using statistics and basic predictive techniques for decades, but now many of them are pushing forward into more innovative areas, including machine learning and artificial intelligence. Many of the most cutting-edge innovations in these areas are instantly available via open source, and enterprises are taking advantage of them through Centers of Excellence (COE) and early prototyping. However, since these are rapidly evolving areas, it has been difficult to move this work into production systems to realize their full benefits. With Open Data Science, enterprises not only are able to manage and govern innovations, but also can take advantage of the quickly emerging additions and updates that are being contributed to open source regularly.

Modern artificial intelligence techniques, in particular, are computationally intensive—that is, they need a lot of processing horsepower and memory to crunch through the data. While in the open source community, the application of machine learning and artificial intelligence is often limited to smaller data to fit into the available hardware, enterprises require infrastructure that will process the data to derive the insights and trigger the actions that meet their business goals. This means that enterprises require their advanced analytics

models and applications to be deployed on massive distributed clusters, and they also take advantage of the latest hardware advances, including GPUs. With these unrelenting enterprise demands to scale analytics, the next decade will see a series of new hardware advances that will require flexibility from organizations wishing to rapidly leverage them.

Interactive Dashboards and Apps

In the internet age, we have come to expect that everything we need to know is as simple as a quick Google search. The chasm between the typical enterprise application and the world of Google continues to widen, and enterprises are demanding new workflows that allow their entire team to have a "Google-like" experience in their day-to-day work. While dashboards displaying simplistic charts with drill-down for detailed views were hugely helpful to enterprises whose workforce grew up with mainframes and desktop computers, today's emerging workforce grew up in the world of Google and are digital natives. To empower this workforce, enterprises are seeking interactive dashboards and web-based apps that provide rich context and the agility to interact with the organization's data in order to gain operational insights. While interactive visualizations are not yet a fully immersive experience, they are a stepping stone to that future. Interactive visualizations bring the data to life in a way that simple bar and pie charts have never done. These interactive dashboards and apps deliver a narrative about the data that enriches our ability not only to synthesize the data but also to understand the trends, forecasts, and tradeoffs to be made in the business. By enhancing the visual experience, enterprises can more readily understand the intelligence embedded in this next generation of dashboards and apps, and use that knowledge to garner the best benefits for their organization.

Self-Service Analytics

Open source communities develop tools for themselves and share these tools with the world. These communities attract people with deep technical expertise, and the tools they develop are aimed at peers with comparable knowledge and skills. So while these tools are often intuitive for individual technical experts, they are not easy for the professional who has deep business expertise but lacks a comparable level of technical expertise. Yet enterprises have far more of the

latter group of workers and are seeking to empower them to make evidence-based decisions as well. Ideally, both constituents—technical experts and business experts—should use the same underlying technology so their work can be collaborative rather than siloed. With Open Data Science, enterprises can blend these worlds by infusing data science work into common business applications, such as Microsoft Excel, so that the business experts can easily "self-service" their insights. This frees data scientists to work on more complex issues that require deeper technical expertise and can unleash new value for the enterprise. By arming both parties with the appropriate tools, data science permeates the enterprise and propels competitive advantage.

CHAPTER 9
Data Science Collaboration

We have previously discussed the data scientist unicorn—that mythical creature that has deep computer science, advanced analytics, and business or domain expertise. If this person exists at all, they are not found in the typical enterprise setting. More commonly, the various perspectives and skills required to deploy high-value data science applications are supplied through the aggregated skill and knowledge of many people in an organization. In the past, collaboration meant that teams operated as though they were in a relay race, handing off the baton once their lap of the track was complete. The adoption of Agile methodologies into the workforce means collaboration looks more like a soccer team, working together to get the ball down the field and score the goal. While soccer players have the luxury of seeing, hearing, and interacting with each other on the field while trying to score the goal, however, most data science teams span departments and locations, making it much more difficult to develop an analysis workflow, perform an ad hoc analysis, or create a data science application. Data science teams need to share expertise and data across these organizational and geographic boundaries to build, test, and deploy data science models that drive value for their organization. When a data science team is empowered to work together as a tight ensemble, helping each other and boosting each other's efforts, they can deploy better-performing data science applications faster for their organization to reap the greatest business value. Collaboration, then, is an essential aspect of Open Data Science.

While each data science team is unique, there are core roles that are typically found on a high-performance data science team. We briefly described these roles in Chapter 2, but to recap, they include:

Business analyst
: An expert in the business or domain that understands the problems, challenges, and opportunities in the business

Data scientist
: Typically an expert in the application of machine learning, deep learning, and artificial intelligence to real-world problems. Data scientists typically collaborate with the business analysts to define the goals and then start exploring the data to begin developing an approach for the data science model that will achieve the goals.

Data engineer
: An expert in data storage, cleansing, and processing. Data engineers typically collaborate with business analysts and data scientists to marshal the data needed for the analysis and perform a first pass at cleansing or normalizing the data so further analysis can be performed. Once an approach is identified by the data scientist, the data engineer will often perform feature engineering to derive insightful attributes from combinations of data or eliminate features that are not crucial for the analysis.

Developer
: An expert in building algorithms and software applications, including the embedding of the data science model into the application. Developers typically collaborate with data scientists and business analysts to ensure that the algorithm and data science application will meet the stated goals.

DevOps engineer
: An expert in operationalizing software on various hardware and operating systems. DevOps engineers typically collaborate with developers and data engineers to deploy the data science application into production.

Different roles on the data science team will "speak" different languages, but it's important to understand that their skills will not conflict with, but rather complement, each other (see Figure 9-1). This diverse group of people, each bringing unique skills and knowledge, can come together and work very effectively if they have the right

platform to facilitate their collaboration, instead of siloed tools designed for a limited role. Open Data Science tools help individual team members accomplish their work and yet also easily interoperate to facilitate the collaboration.

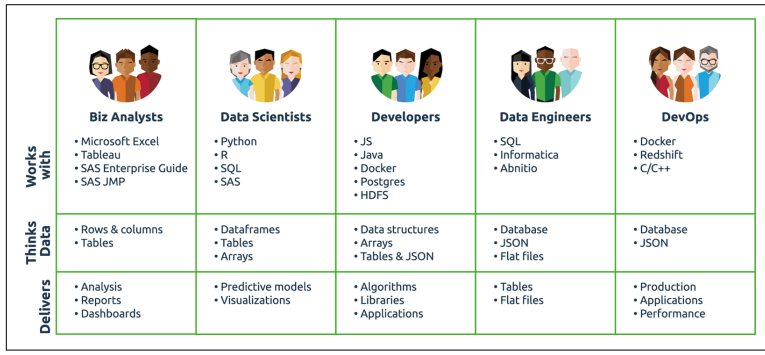


Figure 9-1. Data science team roles complement, rather than conflict with, each other

How Collaborative, Cross-Functional Teams Get Their Work Done

Data science teams typically use a combination of local desktop machines and a centralized server or cloud server to facilitate their collaboration on data science assets. The team will need access to the same data and the same platform to streamline their collaboration.

When data science teams build models, they typically use notebooks to encapsulate and share their work product. Notebooks are essentially the data science equivalent of spreadsheets because they are lightweight frameworks that encapsulate the narrative, data, code, and visualizations for a data science model and, just like a spreadsheet, they can invoke formulas to produce new results, augmenting the "base" data. Updating and running a notebook produces fresh results and interactive visualizations that generate new insights and predictions. A notebook is a point of collaboration for the entire data science lifecycle, from the exploratory stage to production deployment. Different team members can share what they are doing: a visualization expert can come up with great visualizations; data engineers can share where the data came from and how it was transformed for the analysis; developers can come up with new algorithms; data scientists can create predictive models and explain why

they chose the approach they did; and business analysts can write the story of what it means, and why it's meaningful. Algorithms, data, visualizations, and prose all come together to create a "data story."

Just like a spreadsheet, notebooks can easily be shared among members of the team via file sharing or email. However, this method does not improve productivity for the data science team. High-performance data science teams want to create shareable, reusable artifacts that are easy to discover across the team. This allows the data science team to build up foundational data science assets, which they can then compose into data science applications that address increasingly complex issues with ease. Not only does this tremendously increase the team's productivity, but it also allows the team to continuously deliver higher-value data science applications. As the notebook is refined to create such a reusable data science asset, the final version of the notebook can be run as a lightweight production deployment application so that others can use it in execution mode. Examples of notebooks with interactive visualization can be viewed at *http://nbviewer.jupyter.org*.

Data Science Is a Team Sport

In the Open Data Science world, collaboration happens in real time. It's not a "waterfall" model, where individuals work on pieces individually and then pass on the results. Instead, collaborative teams with different skills sets are brought together and are constantly interacting and collaborating. As suggested earlier you should think of them as a soccer team on the field all at once, rather than runners in a relay race.

This approach requires a platform that allows great flexibility to support different types of projects and ad hoc analysis work, and allows the data science models and applications to be deployed into production environments without delay. The key to successful collaboration is the ease with which the platform allows the team to coordinate and communicate and coordinate—often across geographic boundaries and time zones—to efficiently deliver high-impact results. Best practices used by high-performance data science teams include:

Shared Open Data Science platform
: Collaborating teams need to have a common foundation for their data science work. Earlier sections have discussed at length how a common Open Data Science platform enables a high-impact data science team.

Standups/check-ins
: By using regular "standups"—whether daily or a couple of times a week—teams can check in on progress. In this way the team stays aligned, catches misunderstandings early, and clarifies goals as needed to ensure that everyone is on the same page. In an Agile method that has become popular over the past decade in the software engineering world, each team member has one minute to speak and shares a three-part update: yesterday, today, and obstacles to progress. Any discussion is deferred until after the standup, keeping these meetings to less than 10 minutes in most cases.

Project management
: There are many different ways and tools to manage projects. One methodology that works well with high-performance Open Data Science teams is the kanban approach, popularized in the 1970s and 1980s by the Toyota Production System (TPS). This system of small task estimation and tracking allows teams and team leads to match the amount of work in progress (WIP) to the team's capacity to deliver. The popular Trello project management tool uses a visual board to scope the project and cards to identify WIP. These cards are assigned to individuals and passed onto various stages as the work progresses. Individuals also create new cards, easily specifying their needs from other members of the team. These boards and cards make it easy to see at a glance where a project stands, what is ready to review, where there are bottlenecks, and what is still backlogged to be accomplished.

Instant messaging (IM)
: By now widely adopted in many organizations, IM software can accelerate and improve processes that were previously handled by email, phone, or meeting. The use of instant messaging has the advantage of real-time communication of digital assets such as screen shots, data samples, code snippets, URLs, or error messages. There are many high-quality IM platforms available today that include integrations that provide automated alerts for

things such as build systems, bug reports, document updates, or software commits.

Demo events
: Experimentation is a hallmark of any leading, innovative data science team, and having opportunities to share progress, experimentation, and new ideas with others is a great way to connect people within teams and teams across an organization. These events can be organized weekly, monthly, or quarterly and provide an informal space to demonstrate work in progress with short lightning-talk-style project demos.

User groups or meetups
: The Open Data Science ethos is embodied by individuals who feel connected to the larger community, so encouraging team members to participate in community-oriented events (such as evening user group gatherings or meetups and longer workshops or conferences) helps to build that connection and spirit. Companies reap numerous benefits from engaging with the larger Open Data Science community by sponsoring events, providing a high-quality meeting space, or sending engineers to speak. It creates visibility for the company's innovation, establishes relationships that can facilitate staffing, exposes the team and company to community-based innovation, and provides professional development opportunities for staff.

Collaborating Across Multiple Projects

A significant challenge for the data science team is that they'll often be in a situation where they have a current primary project, while at the same time they are required to be updating, maintaining, or reviewing work from a previous project. They may also be simultaneously scoping out the initial phases of some new opportunity on the horizon. On top of that, there are the one-off analytics questions that come up in a meeting or "as a favor" through an email request.

Addressing all of these can be difficult in an environment in which data tools, data sources, data streams, and data transformation strategies are constantly evolving.

In an Open Data Science environment, it is unrealistic to expect everyone to be in lockstep at all times on all projects with exactly the same set of tools. While it's necessary for everyone to be on the same

page for particular projects, it's also necessary for people to have the flexibility to use the latest version or the experimental version of a tool or clustering algorithm, and to switch them out quickly and easily so the project can grow and innovate, even as it's being constructed.

This brings up another reason why reproducibility is key to data science teams. Data science teams need the ability to easily reproduce their entire environment—data, code, specific versions of analytic libraries, and more—so they can share environment and support multiple projects. As discussed earlier in our coverage of Open Data Science trends, virtualization and containerization provide one strategy to address this challenge.

For example, every project can have a managed set of virtual machine (VM) images—perhaps "development," "test," and "production"—with team members working cooperatively with the systems engineering group to identify the necessary requirements at each stage of a project. The VM images are then cut and distributed, allowing everyone to operate from the same starting point. The downsides to this are that it distances the team members from their "natural" operating environment, reduces the flexibility to experiment to find the right Open Data Science tools for the job, and will still leave the systems engineering team with obstacles when it comes to transitioning the project to a production environment.

Using containers is very similar to using VMs, with the advantage that they are lighter weight, faster to start, and faster to create. However, you are constrained on the base operating system images that can be used, and that can be a hurdle for members of the data science team. They will still suffer from many of the same challenges as when using VMs.

Another solution is to standardize on a single portable data science language. Keep in mind, however, that this is the old way of doing things. Exclusive use of a portable, cross-platform language such as Java, Python, Matlab, R, or SAS once held great promise as "the one way" in which analysts, scientists, statisticians, and engineers could all work together. Although this still appeals to some people, experience has shown that trying it will result in a more constrained analytics environment with limited flexibility; no good solution to the issue of variable software versions for different projects; and the use of languages in domains they are not well suited for—for example,

general-purpose programming with Matlab, or creation of web interfaces with Python.

The optimal solution is to leverage a portable, cross-platform software configuration system. Tools that can snapshot and recreate a software environment on any system provide a mechanism to support reproducibility—so that the same analysis environment can be recreated in the future, or deployed on a new system—and that collaboration can happen without having to take a full system image as is required with VMs or containers. The challenge is to verify that all necessary aspects of the analytics environment can be captured suitably for reliable reproducibility.

Collaboration Is Essential for a Winning Data Science Team

It takes a village to do data science right. You need the data scientists, business analysts, data engineer, developers, and DevOps engineers to create reusable data science assets that can continuously drive more value for your organization. Open Data Science work requires teamwork, and collaboration is mission-critical for data science teams to succeed.

CHAPTER 10

Self-Service Data Science

As we discussed in the past chapter, a skills gap exists in the market for data scientists—those mythical unicorns with the ideal combination of computer science, math, and domain expertise. But even those organizations that have managed to snare data scientists can have trouble getting the assets they create to the frontline employees who need them.

Keep in mind that for every data scientist in an enterprise, there are approximately 50 to 100 frontline workers—including business analysts, salespeople, marketing managers, production managers, and customer-service representatives—who could use data science to make better business decisions.

This chapter will explain how a self-service approach can be used to empower business analysts. Equipping the maximum number of people within an organization to perform basic data science activities will allow them to work directly with data, informing their day-to-day work with contextualized intelligence that delivers game-changing business value.

Self-Service Data Science

To make data science easier to get and easier for frontline workers to use, data scientists need to shift perspectives and walk in their shoes. In other words, data scientists need to understand and have empathy for the perspective and experiences of their colleagues who are

involved in other lines of business but who can also benefit from quantitative analytical information and tools.

This means contextualizing data science to the business problem or issue and making "data driven" simply the way that decision makers at every level typically do their work. Just as Siri makes it easy for consumers to perform intelligent searches, data scientists need to embed contextualized intelligence into the way we work in enterprises today.

This is the major issue—call it the democratization of business. The technology world has been trying to democratize data for 20 years now with limited to no success. Tableau made it easy for business analysts to create attractive data charts from historical data sources. But now people want to use data to predict the future or get a recommendation while they still have an opportunity to change the outcome. The next wave of the data science revolution is to empower business leaders and analysts with the right technology to make it easy for them to embed intelligence into all their analysis.

Analysts and other business users have the business context, and the intelligence, to put the numbers in context. There are several approaches to ensuring success when empowering business leaders and analysts:

- Meet me where I am
- Make it dead simple
- Make it intuitive

Meet Me Where I Am

Many frontline workers live in the world of Microsoft Excel since it is easy to combine data, analysis, and visualization to come up with actionable plans. They can easily modify data to perform basic "what if" analyses. But with data science, they would be able to do so much more.

By merging Excel with Open Data Science business users can have a seat at the data science table. This empowers business users to move away from their comfort zones and make their own personal journey from simple consumers to builders of data science applications. With Excel it's easy for them to program even if they don't realize they're programing. Formulas, pivot tables, and even Visual Basic

scripting are familiar to millions of business-oriented Excel users. And it is easy to share work with others in the organization since it is 100% encapsulated in a single file. This can be done via email, instant message, or collaboration tools such as SharePoint.

Imagine breaking Open Data Science open for these millions of business analysts if a similar model can be achieved—for example, by embedding data science assets into Excel. This would empower and connect business users to leverage Open Data Science tools and assets to drive value for the business with greater insights.

This also makes it easy to interact with and provide feedback to the data science teams, so that the data scientists' work can improve as well. Using a strategy of templates, or examples, business analysts are empowered to do more advanced analytics. Although they might not be able to create data science models from scratch, they can hack the assets by copying and/or making modifications so that they start on their journey to becoming data scientists themselves.

Make It Dead Simple

There are multiple techniques for making Open Data Science much easier for frontline workers.

Data catalog

A data catalog can document available data sources, synthesized data sets, or preprocessed data sets available within the organization. This acts as a repository for raw and transformed data and as a template for transformations that can be reused by other teams. An important side effect is that the data catalog also contains full provenance, or the history of the origin and transformations applied to the data.

Say that you have a database where you collect all the transformations that you, your team, and your company are making on multiple data sets. When new data comes in, you clean it for consistency. For example, you want to identify customers as male or female, but some data uses "M" and "F" and others spell it out, and still others only use lowercase letters. Your very first step is to "normalize" your data so that it's all consistent. You would also do things like populate missing values with default values, and identify invalid entries to be purged.

Then, your second stage would be to transform the data. You perform an analysis on male versus female customers using analytic techniques. The data catalog can then record that sequence of data manipulations and register either the generated (transformed) data, or provide a synthetic data source that can be generated "on demand." In this way the data catalog also can be a data service or data source.

Another use of an enterprise data catalog is discoverability: it may be that a required data set has already been captured, cleaned, and published—ready for your team to use with little or no additional effort. You don't have to consider all the downstream ramifications of a transformation because you have the full provenance. This is what a data catalog offers you.

Templates

Templates are like paint-by-number kits: everything you need is in the box. There's a guide (the numbered diagram) and the paints (the preselected color palette). As a novice, you can simply follow the directions and produce a reasonable painting. If you're more experienced, you can start to experiment, altering the mix of colors and loosely interpreting the lines.

A data science template, such as a rotational churn model or a demand forecast model, is similar. It makes it easy for you to connect your data and apply the predefined model and associated visualizations to produce an application or report. Once the frontline workers become familiar with the tools, they can experiment, alter, and customize the model and visualizations. But the template provides a step-by-step guide to begin the journey to make it easier.

Interactive visualization apps

Many organizations use dashboards that display graphs, charts, and data tables in an attempt to simplify data analysis, business insights, and follow-on actions. While these provide an improved and simplified presentation of the organization's data sources, they lack context and often fail to deliver any clarity around the question of "so what?" or even more importantly "what next?" The new world order involves storytelling that commingles a narrative and the supporting data. For the past few years this has been done with infographics, but these are one-off artifacts that require data artisans to produce. Although the impact of these interactive visualization apps can be

very powerful, the lag in producing them has made it difficult for people to deploy them systematically. Now, it's possible to create these power narratives easily with notebooks.

Notebooks

As previously discussed the notebook-based analytics approach has gained wide popularity, lowering the bar for the adoption of advanced analytics tools and strategies. Notebooks combine a narrative that creates relevance and context for any interactive visualizations. Authors, collaborators, and consumers can all interact with the data and immediately reach informed conclusions. These visualization apps allow frontline workers to see and interact with powerful data science models. Although most notebooks today are initially created by data scientists, this can be done in a way that allows frontline workers to experiment with or modify model parameters in order to understand the impact of multiple scenarios.

Make It Intuitively Obvious

Data should be made intuitive to the frontline employees who will actually be using it. This can be done in two ways: make apps that are either intuitively relevant to the employee's industry, like retail, pharmaceuticals, or manufacturing, or create problem-domain apps that focus on a particular problem that spans different industries. Let's take a deeper dive into these two solutions.

Problem-domain apps

Splunk is probably the best-known application for a specific problem domain—in this case, IT logs. The Splunk application uses the rich IT log data and provides interactive reports and visualization apps that allow IT professionals to understand and predict the impact of changes. Problem-domain apps embed data science models into specific business problem applications so that frontline workers can use the predictions and recommendations in the context of their daily workflow. This makes it incredibly easy for frontline workers to incorporate data science into their daily jobs. They utilize Splunk's end-user web-based interface to search IT logs the same way we use Google to search the internet. These applications are easily trusted, as typically hundreds of years of expertise are built into the application, matching much of frontline workers' experience and intuition.

Vertical apps

Vertical applications, such as Guavus—a Big Data application for planning and operations used in the telecommunications industry—are tailored for industry-specific nomenclature, regulations, and norms. These applications are prebuilt to address specific business problems and encapsulate deep industry expertise into the data science models. Similar to the problem-domain apps, vertical apps are prebuilt with the industry-specific data sources and are typically used by frontline workers daily to get their jobs done.

Self-Service Is the Answer—But the Right Self-Service Is Needed

Employees need access to quantitative tools that can provide them with answers to their questions. Self-service Open Data Science approaches can empower individuals and teams through easy-access analytics apps that embed contextual intelligence.

With self-service data science, the employees who are immersed in a particular business function can leverage data to inform their actions without having to wait for resource-constrained data science teams to provide some analysis. This lowers the barrier to adoption, thus expanding the scope of data analytics impacting business results.

Right now, data scientists are unique and inhabit a world of their own. To unleash the power of data, businesses need to empower frontline workers to easily create their own analysis. This infuses intelligence throughout the organization and frees up the data scientists to innovate and work on the biggest breakthroughs for the enterprise. As data and data science become more approachable, every worker will be a data scientist.

The benefits of self-service data science are twofold. First, you get empowered business teams who can leverage their contextual intelligence with the data science to get exciting business results. Secondly, data science becomes embedded in the way business employees work. You know you've reached your goal when you hear an employee say of the data science, "It's just how I do my job."

CHAPTER 11
Data Science Deployment

Enterprises have struggled to move beyond sandbox exploratory data science in their organization into actionable data science that is embedded in their production applications. Those challenges can be organizational or technological.

Organizational challenges usually result from data science teams that are unable to communicate with other parts of the organization. This may manifest as a lack of cooperation and communication between engineering and data teams; there may be no processes in place to integrate data science insights into production application. Engineers might be brought into the discussion once models are already written, while data scientists may not be trusted with access to production systems or the creation of production-oriented applications.

Data science teams may have problems integrating insights into production if the team lacks the appropriate experience. Having only data scientists with modeling skills, but without data engineering, DevOps engineering, or development skills, is a recipe for conflict. Data science teams need to be able to understand production system requirements, constraints, and architecture, and factor those into the packaging, provisioning, and operation of their "production deployed" analytics workflows, models, or applications.

Technological challenges, too, make it difficult to bring data science models to production. Organizations where the engineering and data science teams use a disjoint combination of multiple programming languages (including, for example, Java, C, R, and SAS) should

expect to experience extra challenges when it comes to integration. Some organizations establish costly rewriting processes to move exploratory data science to production. This tends to be error-prone and laborious, taking months or even years to fully reap the benefits promised by the data science team.

We have observed organizations that, out of necessity, make data scientists responsible for their own deployments. Such decisions can generate a lot of tension: DevOps frustrated by the choices made by the data scientist team, and the data science team forced to invest time into an area (IT infrastructure management) that is not their core strength.

Some data science teams tackle the problem of deployment with technical tools, both commercial solutions and in-house software deployment systems. Other organizations diversify the data team to include DevOps, data engineers, and developers that can focus on deployment, but are part of the project from the start. This enhances the team's ability to make informed decisions that will later smooth the deployment of data science models into production environments.

This chapter covers the pros and cons of different data science deployment strategies for a variety of assets used in Open Data Science. The ultimate goal is to facilitate the efficient promotion of data science artifacts to production systems so they are accessible by the widest possible audience. We will see how Open Data Science platforms that automate operations and aid the progression from "experimental" to "data lab" to "production" can be implemented in a way that allows data scientists to create better models, visualizations, and analyses, freeing them from dealing with packaging and provisioning, and supporting operationalized assets.

What Data Scientists and Developers Bring to the Deployment Process

Data scientists approach software deployment from a fundamentally different position compared to typical enterprise systems engineers. Data scientists focus on creating the best quantitative models and associated analyses. An organization committed to Open Data Science will support data science teams in retaining their appropriate

focus, making it easy for data scientists to provision the infrastructure they need without impacting their productivity.

One of the most effective strategies used today is to leverage a platform that can expose data science assets through a universal API, allowing it to be incorporated into larger production systems without having to burden an engineering team with recoding algorithms or a DevOps team with supporting incompatible service interfaces. RESTful APIs, which present HTTP-based network services that transact data typically through JSON, can be this universal interface.

The Traditional Way to Deploy

Until recently, the enterprise deployment of data science assets required rewrites of SAS, Python, or R code into "production" languages like C or Java.

Such efforts are prone to errors. Developers frequently don't understand the models, leading to translation mistakes, and it is rare to have suitably comprehensive test suites to be confident that the reimplementation is complete and correct. This problem is exacerbated by the fact that, once deployment is complete, data science teams are unable to help resolve problems with the deployed models since they are not familiar with the reimplementation.

This process is very costly, it duplicates effort, and businesses don't derive much benefit from it. Instead they get expensive errors and duplicated effort reimplementing work that has already been completed. The two teams typically operate independently and are unable to coordinate their efforts in a productive way, often leading to tension and misunderstanding.

The downsides of this traditional deployment strategy can be summarized as follows:

Cost
: Time is money and the delay introduced by porting data science models to production systems delays the organization's ability to derive value from those analytics assets.

Technology
: Porting from data science languages to production system languages introduces errors in the models and obstacles to maintaining those models.

People
: Having two distinct teams with different priorities for and experiences with regards to computational systems and data processing leads to organizational dissonance and reduces the impact of both teams.

Successfully Deploying Open Data Science

In the new world of Open Data Science there are solutions that help mitigate these legacy deployment challenges. Most significantly, the technical aspects of deploying live-running data science assets can, today, be addressed.

Assets to Deploy

There are a number of analytics assets that can be deployed as part of an Open Data Science environment:

Machine learning models
: Machine-learning models can be embedded in applications. Recommendation systems, such as those commonly seen on ecommerce websites that recommend other buying options based on your past history, contain machine learning algorithms customized for particular data formats and desired outputs.

Interactive data applications or dashboards for business users
: Interactive visual analysis tools deployed onto corporate intranets are common in most organizations today. Some popular proprietary systems are Tableau or Qlik, while Shiny provides this capability for R in the Open Data Science ecosystem. Data science web applications are enhanced when they incorporate high-quality interactive visualizations as part of their interface.

Pipelines and batch processes
: Entire data science workflows can be established in a way that can be packaged and shared, or deployed into production systems to scale the workflow for parallel processing.

Processes to Deploy

Data science models exposed as network services
: Using RESTful APIs, discussed earlier, data science models can be deployed in such a way that they can be integrated into other applications. Amazon Lambda is an example of one such network deployment system for simple models.

Web-based applications
: Entire web-based applications may be developed. Frameworks that facilitate the rapid deployment of simple interactive web-apps are well established, such as Java "portlets," IBM Web Sphere Portal, or Heroku. However, a new generation is now emerging to serve the Open Data Science community, such as the RStudio Shiny Server. In Python both Flask and Django provide similar "app"-oriented extensible web frameworks, allowing a data science team to focus on the business analytics, leaving the core capabilities around authentication, session management, data, and security to the common framework.

Traditional client-server applications
: Some data science applications require a heavy-weight custom client that connects to a network-based server. These continue to be present in special situations in the Open Data Science world. The deployment of both the client and server components needs to be coordinated and managed across an organization.

Open Data Science Deployment: Not Your Daddy's DevOps

In summary, running a DevOps data science team is not like running a DevOps environment. Data scientists are not and should not be a part of operations. Instead, they need to concentrate on making better models and performing better analyses.

To enable this, enterprises face three factors when considering analytics deployment:

- Data science teams focusing on analytics and not operations
- Data science assets that need to be managed and distributed across an organization

- Data science processes that need to be deployed into production and then managed

To address these, many of the strategies of the now-popular DevOps style of rapid provisioning can effectively be applied; however, it is also necessary to have an Open Data Science environment that can facilitate the transitions between individual analysts, centralized data lab environments, and large-scale automated cluster deployments used in production systems.

CHAPTER 12

The Data Science Lifecycle

With the rise of data science as a business-critical capability, enterprises are creating and deploying data science models as applications that require regular upkeep as data shifts over time. This is due to the changing data inputs and the insights gained from using the model over time. Many organizations include feedback loops or quality measures that deliver real-time or near-real-time reports on the efficacy of a particular model, allowing them to observe when outputs of the model deteriorate. In this way, a handful of initial models can quickly be refined by Open Data Science teams into "model factories" where tens to hundreds of deployed models may be "live" at any given time. These are then coupled to the results generated by these models, and it is clear that model management quickly becomes a critical requirement of the Open Data Science environment.

In this final chapter, we will explore why models have to be continuously evaluated as part of a data science lifecycle and what can be done to combat "data model drift."

Models As Living, Breathing Entities

In the course of day-to-day business, many quantitative models are created, often without clear visibility on their number, variations, and origins. Many, if not most, are good only for temporary or one-off scenarios; however, it can be hard to predict in advance which will survive, be enhanced, and promoted to wider use.

Imagine a scenario where an executive contacts the analytics team and says, "Shipping costs are going through the roof, can you figure out what the problem is?" A traditional business analyst would capture historical information in an Excel spreadsheet, then work to create a simple analysis model to see where and when the extra costs came from. This reveals that the shipping contract changed to a different service provider in the middle of the last quarter, resulting in higher packaging and shipping costs. Good—problem solved.

This spreadsheet-based model might work for a while to track shipping costs and provide visibility if costs vary significantly again, but as the business continues to grow, the contract may be revised or the service provider changed again. And that would be the end of that model, and all that work.

This scenario illustrates that data science models are living, breathing entities that require oversight. We need to start thinking of our data science models as important as our most valuable capital assets. After all, they impact how our products and services are sold, how demand relates to revenue, and how we forecast our costs.

You need to control who gets to use these models, who has permission to modify them—and who decides when it's time to retire them.

These latter points are critical because data science models can change. In the scenario described earlier, the first service provider may have been based on the East Coast where the company was located; however, as the business expands to California and a larger revenue, base the shipping cost model is no longer accurate. Service providers change, costs change, and the old cost models become increasingly inaccurate. This is a "model drift," where the model has drifted away from providing accurate estimates for the system it was designed for (shipping costs).

That's what lifecycle management means—what you do when the model no longer fits the data. It establishes reviews on a periodic or even continuous basis to ensure the model is still accurate.

The Data Science Lifecycle

The data science lifecycle is a combination of two closely related capabilities that help enterprises manage their data science assets:

Data science asset governance
: Data science models are high value assets—expensive to produce and capable of delivering high-value to organizations. They need to be tracked just as other high-value corporate assets. Controls are required to determine who has access to the assets, and who has the rights to modify or delete them.

Data science lifecycle management
: Data science models lose the power of their impact over time as data and business change. Therefore, the models must be continuously reevaluated or replaced to maximize their effectiveness

Benefits of Managing the Data Science Lifecycle

There are a number of benefits for having a solid data science lifecycle in place. Two are especially important:

Reusability
: Keeping close track of data models and sharing them freely throughout the organization allows them to be leveraged for more than just one purpose. This also helps build upon previous work to quickly deliver value in the form of new data model assets.

Continuously improved results
: Data science models are powerful. They can and do impact the bottom line. Alternatively, stale or inaccurate models can lead to missed opportunities, lost revenue, increased costs, and risk of noncompliance.

Data Science Asset Governance

The data science platform you choose becomes the central repository for all data science assets. It retains information about each data science model asset, including the following:

Goals
: What is the business purpose of the model? What are the expected results? What are the characteristics of the model (for example, coefficients for linear models, rules for decision trees, or goodness-of-fit measures)?

Authorization information
: Who created the model? Who approved the model? Who requested the model? Who activated, suspended, or archived the model?

Provenance information
: When was the model originally created? When and what have been the subsequent revisions to the model? What is the most recent version of the model?

Compute context
: What is the deployment platform? What is the configuration of the deployment platform? What resources are available on the deployment platform? When has the model been used, and what was the performance of the model?

Data lineage
: What are the source(s) of data? What transformations have been applied to the source(s) data? Where is the transformed data stored?

Model Lifecycle Management

Model lifecycle management is the process of periodically or continuously reevaluating models over a period of time to reassess the accuracy of the model considering changing conditions. In this process, the model is evaluated to determine if the accuracy has drifted over time due to changes in data, business rules, or other conditions.

Experienced data scientists and analysts develop the discipline to incorporate monitoring strategies into their model development workflow and expect to monitor their models' performance over time, refreshing models when necessary. Identifying expected results and the parameters for "normal" behavior early on can alert you to model drift. Filters to catch outliers and anomalies in the model output can further provide indicators of model drift—hopefully before something catastrophic happens.

As a data science professional leading the adoption of Open Data Science within your organization, one of your key responsibilities may be the delivery and operation of reliable predictive models. In light of this, it is essential to understand your ongoing model output, performance, and quality in order to identify drifting or misbehaving models at an early stage. With appropriate visibility into

model performance, you will be equipped to intervene preemptively, performing course corrections with models before they deteriorate and undermine the organization's analytics information flow.

In order to be equipped to regularly evaluate model performance and effectively embed alerts in the operational system, you have several possible strategies. The most common is the "champion challenger" methodology.

The Champion-Challenger Model

With this technique the deployed model is compared with new "challenger" models, and the model with the best results *usually*—note this disclaimer—becomes the new "champion" model and replaces the previous champion.

New challenger models are created under one of two conditions: either an alert that one of the data quality controls senses model drift, or based on a routine model test schedule—say, once a month or once a quarter (best practice would be continuous monitoring, but that is not always possible).

The way it works is simple: the new model is created based on the most recent data. Performing A/B testing against the current champion model—using backward-facing data, since the outcome is now known—provides two alternatives where one will generate the most accurate outcomes.

Getting Over Hurdle Rates

The model that wins is only usually declared the champion winner because of something called *hurdle rates*. Hurdle rates are the costs of switching models. After all, there are real costs involved in making the change, as different systems may be affected, various groups within the organization mobilized, and possibly a formal model review process initiated. This is especially relevant in regulated industries, where there could be significant compliance costs to moving a new model into production.

Other Data Science Model Evaluation Rates

In addition to the champion-challenger method of ensuring the accuracy of data models, a number of other model evaluation tech-

niques exist to measure the accuracy, gains, and the credibility of the model.

Accuracy
: The percentage of predictions the model makes that are correct.

Gains
: Comparison of the business results of using the model versus the results of not using the model. In effect, this measures the performance of the model.

Credibility
: Measurement compares the training data used when creating the model with current (new) data to determine the predictive quality under operational conditions (in contrast to training conditions).

Testing techniques can be static, with manual tests typically done once but which can be implemented periodically. However, they can be automated and implemented so that the testing is a closed loop with continuous reevaluation. This is called *Continuous Integration and Continuous Deployment* (CICD), and in this type of environment an adaptive controlled system is used to determine the model that best fits selected criteria and is automatically promoted to the production system. Ultimately, CICD is what most companies aspire to.

Keeping Your Models Relevant

As data science becomes a critical part of your go-to-market business strategy, you will be creating and deploying more and more data science models as applications. These need to be refreshed regularly. That's because data can change and shift over time, which affects the accuracy of your models. Using good data model governance and data lifecycle management techniques, you can keep your models relevant even in today's fast-moving business environment.

About the Authors

Michele Chambers is an entrepreneurial executive with over 25 years of industry experience. Prior to working at Continuum Analytics, Michele held executive leadership roles at database and analytic companies, Netezza, IBM, Revolution Analytics, MemSQL, and RapidMiner. In her career, Michele has been responsible for strategy, sales, marketing, product management, channels, and business development. Michele is a regular speaker at analytic conferences, including Gartner and Strata, and has books published by Wiley and Pearson FT Press on Big Data and modern analytics.

Christine Doig is a senior data scientist at Continuum Analytics, where she has worked, among other projects, on MEMEX, a DARPA-funded project helping stop human trafficking through Open Data Science. She has 5+ years of experience in analytics, operations research, and machine learning in a variety of industries, including energy, manufacturing, and banking. Christine loves empowering people through open source technologies. Prior to working at Continuum Analytics, she held technical positions at Procter and Gamble and Bluecap Management Consulting. Christine is a regular speaker and trainer at open source conferences such as PyCon, EuroPython, SciPy, and PyData, among others.

Christine holds an M.S. in Industrial Engineering from the Polytechnic University of Catalonia in Barcelona.

Ian Stokes-Rees is a computational scientist who has had the opportunity to work on some of the biggest "Big Data" problems there are over the past decade. He loves Python and promotes it at every opportunity. Ian's greatest interest is in enabling communication, collaboration, and discovery through numbers, narratives, and interactive visualizations made possible by high performance computing infrastructure.

Ian's love of computers started with an Apple II and Logo at the age of 8. In pre–public-Internet days, he ran a BBS in the Toronto area and studied Electrical Engineering at the University of Waterloo. Highlights since then include several years at a tech startup in the UK, a PhD at Oxford working on the CERN LHCb experiment, two years at research centers in France, four years of postdoctoral research in computational structural biology (proteins! viral capsids!

lipid bilayers!) at Harvard Medical School, and a year at the Harvard School of Engineering as a lecturer in computational science.

Today Ian is a Product Marketing Manager at Continuum Analytics where he helps shape the future of Open Data Science.