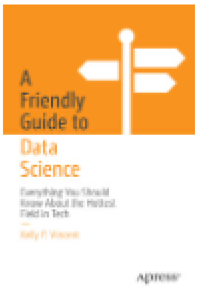

A Friendly Guide to Data Science: Everything You Should Know About the Hottest Field in Tech

Kelly P. Vincent

Published by Apress

5. Figuring Stuff Out: Data Analysis

Kelly P. Vincent[1]

Renton, WA, USA

Introduction

In previous chapters, we've talked about the most fundamental thing in data analysis and data science—the data itself—along with statistics, which is the foundation for everything else in data science. You obviously can't do anything in data science without data. But you also can't just take some raw data and start pulling out fascinating insights or predictions from it. There are a lot of steps to carry out before the data can yield valuable information. In fact, there are many ways that data can be explored, and the field of data analysis is devoted to understanding and working with data. Data analysis primarily involves slicing and dicing data in well-informed ways to extract meaning from it with analytical tools. These tools include programming languages, spreadsheets, and techniques from the statistics we covered in the previous three chapters, especially charts and other visualizations. Often data analysts stick to descriptive statistics, but some will delve into the more advanced statistics. Data analysts have been around for a long time, looking at what has happened before, explaining what it means, and sometimes using that information to help us understand the world and predict the future.

This sounds a lot like what data science can do, and that's not a coincidence. It can be hard to say where data analysis ends and data science begins. I think of the two being on a continuum with lots of overlap. Almost always, a data science project involves doing data analysis, especially at the beginning. Most data analysts' work does not dip into the more advanced data science world, but many data scientists do data analysis all the time. In fact, in order to be a good data scientist, you need to be a good data analyst first.

"Data analysis" is a loaded term that means a lot of different things to different people in the business world, but the simplest definition is that it is the process of investigating and analyzing data in order to better understand what that data represents. We will talk more about this in Chapter 7, but the more general label of "analytics" is divided into four types: descriptive, diagnostic, predictive, and prescriptive. As you can guess, descriptive basically just describes what's in the data, while diagnostic seeks to explain things by looking into the data. These two are the basic domain of data analysis, with data science more focused on the last two.

So data analysis work can involve a huge range of techniques, and if you look at job listings with the title “data analyst,” you will see a surprising range of skills required. In many cases, they are actually looking for a data scientist, asking for advanced programming skills, machine learning, and more. Perhaps they are trying to avoid paying a data scientist salary by using the other label, or (more likely) they don’t know what they want or need. In other cases, they are looking for a high school graduate with basic math skills who knows or can learn to use Excel. Most of the time, the title means something in the middle. This chapter is going to focus on that Goldilocks data analyst—someone who would normally be expected to have a college degree along with several other skills, but not be as technical as a typical data scientist.

You may wonder, if data analysis is its own field, why does it have a chapter in a book on data science? As I mentioned above, being a good data scientist involves being able to do good data analysis. Someone who’s a data analyst by job title may be more skilled in visualization or other things than a data scientist might be, but good data analysis done early in almost any data science project is critical to the project’s success. This is because of a fundamental fact of data science: you cannot do good data science if you do not understand the data. Data analysis has a set of approaches that bring about that understanding.

While this chapter is about data analysis and will refer to those performing this work as data analysts, this is just the name of the hat they’re wearing and equally applies to data scientists performing the data analysis part of their overall process. Additionally, data analysis can involve a lot of different approaches and tools, including those that fall under the label of statistics. This chapter will focus on the concepts, processes, and basic techniques of data analysis. It will start with a history of the field and two examples of data analysis in the real world. Then I’ll go over the four critical skill areas needed to be a good data analyst. Finally, I’ll cover the process that is generally followed to do data analysis work, a process called CRISP-DM (CRoss Industry Standard Process for Data Mining).

The History of Data Analysis

Data analysis has been around a long time, basically as long as data itself. The whole point of data is generally to understand whatever it represents better, and that’s what data analysis helps you do. Of course, the kinds of analyses the ancient Sumerians and Egyptians were doing were incredibly simplistic compared with what we do now, but it was still important. Figure **5-1** shows an account of commodities on an ancient Sumerian tablet from around 2,500 B.C.E. This one shows quantities of barley, flour, bread, and beer. The data from these and other tablets was used by rulers to assess taxes for their citizens.

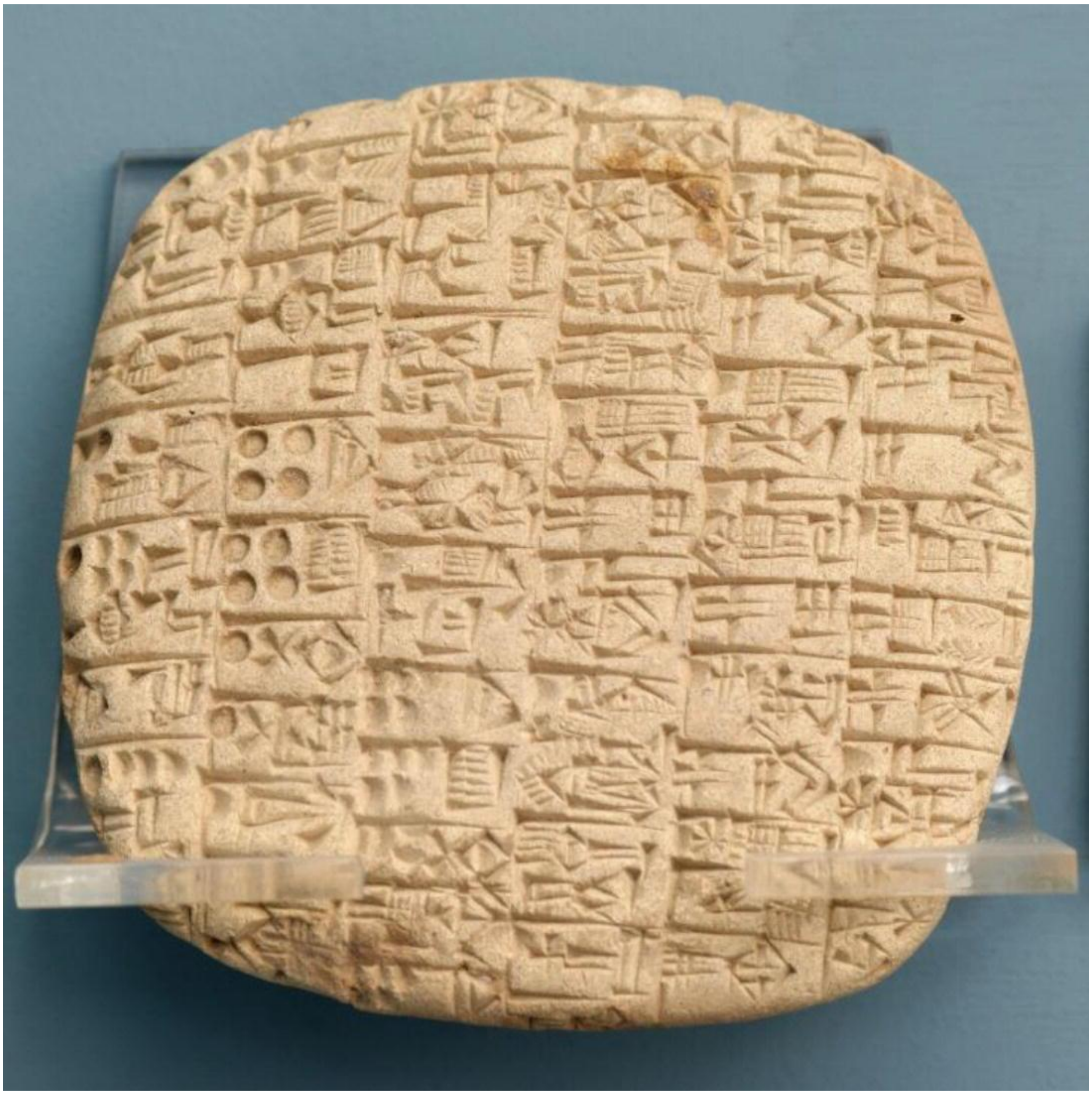

Figure 5-1

Ancient Sumerian tablet circa 2,500 B.C.E. tracking quantities of several commodities (barley, flour, bread, and beer). Source: Sumerian economic tablet IAM Š1005.jpg, https://commons.wikimedia.org/wiki/File:Sumerian_economic_tablet_IAM_%C5%A01005.jpg

While basic tasks with data have been done for a long time, data analysis as a modern field really took its first baby steps in the 1600s and didn't really learn to walk until the twentieth century. A lot of the important work in data analysis has been in visualizing the data in ways that help people understand the data and situation better. There are some pretty cool visualizations (especially maps and charts) that came out in the 1800s that we will see later in the book, with one simple original visualization in an example below (Figure 5-2). There is an entire chapter dedicated to visualization later in the book.

I'm going to talk about the computer and how it revolutionized data analysis next, but it's also worth looking at what data analysis can do before we understand how it's done. I'll share a couple of examples of real-world data analysis work. One of these is quite recent, but the other one was done more than 150 years ago and is still impressive today.

The Advent of the Computer

In the early days, analysis was always done on a rather small amount of data. It might be too much for people to keep it all in their heads, which is why visualization can be useful, but it was still a small enough amount that calculations could be done manually. There are definite limits to how much data was too onerous. We saw this in the previous chapters on statistics, where a sample size of 30 was "big enough" to use the more demanding Z-test over the t-test. Thirty data points sounds laughable to a modern data scientist, but things changed with the computer.

The computer has revolutionized a lot of things, and data is no different. The biggest change it enabled in data analysis is the ability to look at large amounts of data. This doesn't necessarily mean "big data," a term we mentioned in Chapter 1 and will revisit later. It simply means that everything doesn't have to be calculated by hand anymore, so we can use a lot more data. Early data analysis often involved collecting data points and running them through simple statistical techniques, perhaps starting by summing the numbers by literally adding the numbers together on paper or later with a simple calculator. Working with even hundreds of data points this way can get unwieldy, and as the data grows, these calculations soon become impossible.

US CENSUS TABULATING MACHINE

The US Census was getting very complicated by the last few decades of the 1800s. Hand counting people was too expensive and also error-prone. A machine was invented that helped speed up the process for the 1870 US Census, but it was still a mostly manual endeavor. The 1880 US Census was so difficult and time-consuming that they did not finish it until 1887. They needed something different. The first solution that came in was called a tabulating machine, and variations of that machine were used through 1940, after which the Census Bureau finally moved on to proper computers.

The main sign of this shift for regular people is the creation of statistics and analysis tools that allowed people to do their own analyses on the larger datasets that were difficult for manual work. One of the earliest was a product called SAS that's still used by some statisticians, especially in the insurance industry. It's a programming language with a custom, proprietary interface. It was first developed in the second half of the 1960s and is still evolving. Another product that came out in the early days is SPSS, which has primarily targeted people working in the social sciences, so it's used a lot in academia. By the 1970s, a lot of statistical work was done in one of the era's workhorse general programming languages, FORTRAN, which was difficult for less technical statisticians. The S programming language emerged in response to this, designed as an alternative to working with FORTRAN directly. People started using it, and then a version of S called S-PLUS came out in the late 1980s. S in general has been superseded by R, the modern open source and free statistical language based on S that was originally developed in the early 1990s. Nowadays, R is used by statisticians and data scientists and some data analysts, although most data scientists are switching to Python, which is a general-purpose programming language with a lot of statistical libraries that is often regarded as better than R because of its ease of use and performance.

OPEN SOURCE VS. PROPRIETARY TOOLS

With the exception of R and Python, the products mentioned above are proprietary, so they are very expensive and not accessible to most individuals. The proprietary tools are falling out of favor as people switch to open source R or Python. But a couple things that keep the proprietary products around are that they include guaranteed security and compliance, critical in many industries, and that companies that have code written in these languages or tools would have to rewrite everything in their chosen open source language. This is expensive, time-consuming, and not without risk (somebody could accidentally introduce a new bug), so it will be some time before these tools are abandoned, if they ever are. An additional reason that organizations sometimes stick to proprietary software is that because they are paying for it, they are entitled to customer support and can also influence future development of the software. Anyone who's gotten stuck with some code that isn't doing what they expect can understand the value of just being able to pick up the phone and call someone for help rather than hitting up Stack Overflow, the Internet's best free spot for technical questions. However, it is worth noting that there is customer support available for open source tools through some private companies (for a fee, of course).

Examples of Data Analysis in the Real World

Examples of data analysis are everywhere, but I've picked a couple interesting and very different examples to look at here. The first has to do with bringing baseball into the data age, and the second is an early win for public health.

Example 1: Moneyball

For decades, American baseball teams were staffed by scouts with "good instincts," who would go out into North American high schools and colleges to find young players with the aid of word of mouth. There was a strong tradition of scouts going by gut instinct and considering the potential of these young men with some consideration of traditional stats, like their hitting average, home runs, or RBIs (runs batted in). They would work with other team officials to prioritize a list for the draft to bring these players on with the intent of developing them into amazing players. The scouts were critical to this process.

In the mid-1990s, the owner of the Oakland A's died. He had been bankrolling expensive players with a philanthropic mindset, but the new owners were more practical and didn't want to spend so much on the team. As a consequence, the most expensive—that is, the best—players left for greener pastures. The team was no longer winning games, and nobody liked that. After a while, somebody in the organization had a different idea—what if they looked at more detailed stats of players where they currently played rather than trying to guess their potential? Some very committed baseball fans had been collecting and analyzing nontraditional stats for two decades, but the A's were the first official baseball organization to take this approach seriously.

This new approach, which came to be called Moneyball after the book about the approach by Michael Lewis, was embraced by several decision-makers in the organization, including the manager, Billy Beane. They started digging into more obscure statistics like on-base percentage and slugging percentage and looking at which stats led to real payoffs in the sport. This enabled them to identify undervalued and underutilized quality players already in the league as well as stronger draft picks, whom they were able to bring onto the team with very little investment. Beane then used further data analysis to inform other decisions, such as the best order for the batter lineup, and subsequently created a strong, winning team again for pennies on the dollar in the philanthropic era. As an example, the budget team won 20 consecutive games in 2002, breaking an American League record.

The rest of the baseball world took note, and soon other teams that still had a large budget were able to build up their rosters. Moneyball is widely considered to have completely changed the sport, and it's still an arms race situation, with every team trying to figure out a way to use data in new ways to get another small advantage. Now, it's considered a risk to not study player stats in this way.

Example 2: Stopping Cholera in London in 1854

We return to London for one of the most famous examples of data analysis solving a real problem, the work of John Snow during a cholera epidemic that gripped London in 1854. At that time, people did not understand how illness was spread, as the concept of germs did not exist yet. Instead, it was believed that "bad air" was the general cause of diseases spreading from person to person. Snow was ahead of his time in many ways, even revolutionizing medicine by popularizing the use of chloroform in surgery as the first anesthetic. But his major contribution to data analysis occurred when he made a map of London during the 1854 epidemic and added a bar representing each death from cholera at each home location. It was clear from this view where the cases were concentrated, and based on this, he identified a particular water pump on Broad Street as the source, because most of the victims were getting their water from this particular pump. You can see a zoomed-in view of the map Snow used, focusing on the area around the Broad Street pump in Figure 5-2. Rather amazingly, he convinced authorities to block access to that water pump immediately, and the epidemic soon petered out.

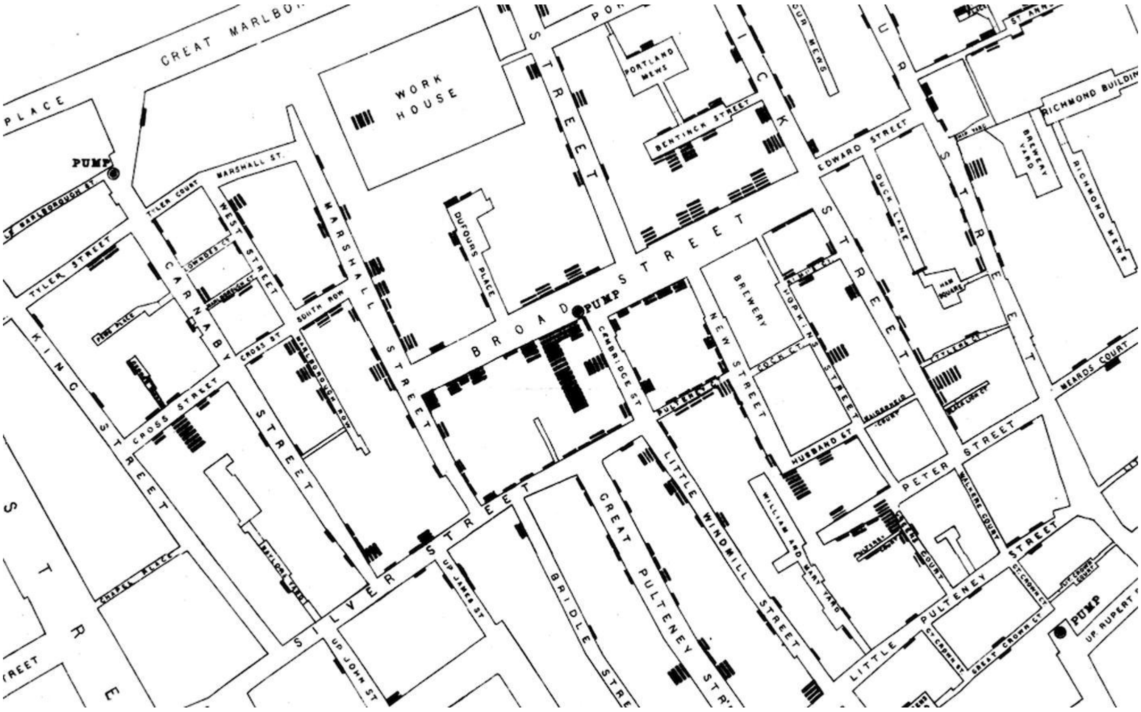

Figure 5-2

The map John Snow created showing deaths from the 1854 cholera epidemic. Source: A cropped area from "File:Snow-cholera-map-1.jpg," https://commons.wikimedia.org/wiki/File:Snow-cholera-map-1.jpg

On the map, all of the darker black marks are small bars indicating victims. Some are spread out, but it's pretty obvious from looking at this map that something about the location close to Broad Street was exposing people to the disease. There's a huge stack of deaths on Broad Street right next to a dot labeled PUMP. People living near there were getting their water from that pump. It was something we call domain knowledge, basically knowledge about the world the data comes from, that allowed Snow to conclude it was the water pump, instead of a cloud of "bad air" hovering over the area for some reason. Snow was a doctor and scientist, and even though germ theory wouldn't be established for several more years, there were doubts among experts about the bad air theory. Snow thought about the day-to-day lives of people living in this area and realized that they would all be getting water, so that was a potential source. When they shut down the Broad Street water pump, nobody was certain that the epidemic would be stopped. But it was, and that realization that water could harbor disease added to scientists' understanding of disease and the ways it could be spread.

Fundamental Skills for Data Analysts

I've talked a bit about what data analysis involves. Data analysts must have certain skills and attributes to be successful. Although some people have a natural aptitude for data analysis, it is a field most people can learn to be good at if they have an open mind, develop a data mindset, and work on the skills I'll talk about in this section. A range of skills—functional, technical, and soft—are necessary for a good data analyst. Additionally, domain knowledge is critical to doing good data analysis. We will look at the main skills below.

Functional Skills

Functional skills are pretty high level and are the ones that help you decide how to go about solving a problem. They involve both natural attributes and high-level ideas and skills learned in courses like science and math. The most obvious functional skill is logical and systematic thinking. You need to be able to work your way through logical steps to come to important conclusions and be able to back them up. Part of this is being aware of your biases so they don't impact your work. Although this is partially a soft skill (discussed below), you need to be able to listen to other people and understand their perspective even if it's different from your own. Related to this are organization and the ability to follow a process. All data analysts need to have a good foundation in math, and some roles require statistics knowledge as well. A good data analyst will also have natural creativity. You hear a lot about the value of out-of-the-box thinking or intellectual curiosity, and it is absolutely true that the most exciting insights often come from someone looking at things in a new way. One of the attributes people who work in data analysis and data science will often talk about is a *data mindset*—this is basically having an open mind, not jumping to conclusions, understanding that data is a representation of something, and respecting the data in whatever form it takes.

Technical Skills

While the primary technical skills depend on the specific role, all data analysts will need to have a general comfort level with computers that will enable them to learn any particular software their role requires. At a minimum, a comfort level with Microsoft Excel and Word (or the Google equivalents) will be needed, and most data analysts are expected to be quite experienced in Excel, including with many of the more advanced functionalities like pivot tables and v-lookups. Many analysts will need to have an understanding of database systems that they will be working with, and they are often expected to use Structured Query Language (SQL) to interact with those databases. Although it is not as common for data analysts as data scientists, many analysts will also do computer programming, usually in Python or R (this trend is on the rise, too). Sometimes data analysts will even code in VBA (Visual Basic for Applications, a programming language embedded in Microsoft Office products) in their Excel work.

Soft Skills

Unless you are working on an entirely solo project you intend to never share with other people, soft skills are also important. The term "soft skills" generally refers to the set of different abilities needed for interacting with other people in order to get work done. Most of these skills are required for virtually any job, but there are some that are specific to the technical work that data analysts and data scientists do.

Some of these have to do more with a mindset or attitude than specific skills. Being generally adaptable is crucial, as things often change during a data analysis project. Having a growth mindset, or an understanding that you always need to be developing your skills in data analysis, is important. The field is constantly changing, and you have to keep up to date with your knowledge and skills. Additionally, keeping in mind the impact that your work can have will help you make sure you are acting ethically.

Time management covers an important set of skills. You will usually be working on more than one thing at a time. You must be able to understand what tasks are most important and have a decent idea how long each thing will take, and when something is really due, so you can understand what you should be doing at any given time. The ability to prioritize is necessary in most jobs. One aspect of doing data analysis work is that sometimes you can't work on something because you are either waiting on someone else or you are stuck at a step that has to complete before you can do the next steps (for instance, you may be running code that takes an hour to finish, and you can't proceed on that project without the results). So you need to be able to work on multiple projects at once. You need to be able to understand how each of your tasks works so you can effectively prioritize and allocate the right amount of time to each in order to meet deadlines.

Another set of soft skills involves communication with a variety of different types of people. Usually, everyone who's expected to be users or beneficiaries of a project are called *stakeholders*. This includes technical people, like your deeply technical data analyst peers, and nontechnical people whose brains will shut down as soon as you mention the term "p-value"—and everything in between. Frequently, the work you are doing is ultimately for people who are not technical, so you need to be able to explain your work and conclusions in ways they can understand. You will often find that people don't trust you or your findings unless they feel like they can understand them. They generally do realize they aren't going to understand all the "fancy math," as we often say, but they still want to get the basic concepts behind the information you present. This is one of the reasons data analysts and scientists often prefer the simpler—and more explainable—solution over the more complicated/fancier one. This strategy will be discussed later in the book.

One of the common ways you interact with others is through presentations, so being able to create an easily followed presentation is valuable. Communication skills are not one-way, either—it is important to be able to listen and receive feedback, which can come both from your customers and your leadership. Unfortunately, conflict with customers and leadership sometimes arises, and you need to be able to stay calm and negotiate or simply listen to understand what is necessary, even in cases where you may disagree. There are times when standing up for what you believe is right is important, but other times it is more prudent to stay quiet in the moment (and sometimes indefinitely). In some environments, rocking the boat can get you in trouble and even retaliated against, so you should always consider your circumstances when going against the grain.

Companies sometimes have particular cultures that define communication styles—for instance, some may want to avoid conflict and prioritize people's feelings by avoiding direct speech, whereas others prefer to keep everything clear and in the open, so direct communication is favored. Knowing how to avoid hurting people's feelings while still getting your message across is valuable in the first case, and being able to be direct and clear without being mean is valuable in the latter case. It's actually been found that organizations that favor direct communication and don't shy from conflict, but still respect people's feelings while finding constructive ways to address the conflict, are more successful, but the indirect style is found more often than not.

Domain Knowledge

Another important area in data analysis is called domain knowledge, which basically just means expertise in the type of data you are working with. We'll talk about it in more depth in Chapter 10, but for now know that domain knowledge is considered one of the three pillars of data science and is also extremely important in data analysis. It's basically specialized knowledge and practical understanding of a particular "domain," an area such as financial data, gameplay data, website usage data, medical data, retail data, and so on. Usually, it implies deep knowledge of even more specific types of data within those areas. If you do not have a decent understanding of the data you are working on, you cannot do good data analysis—or good data science.

"FREE" DOMAIN KNOWLEDGE

Most of us wouldn't know where to start if we were handed a bunch of data about windmills in Europe. But a lot of the time we know about something just from having been involved in that world. For instance, college students will all be comfortable with data about students, classes, and instructors. We would know that there can be a class, like ENG 301 American Literature 1, that really is more of an idea, existing as a course description in the college catalog, for instance. A more tangible class would be one that made it on the schedule of classes for a given term. Section 002 of American Literature 1 is scheduled in the fall semester of the 2022–2023 school year, at 3:30 p.m. Tuesday and Thursday, with Alex Thorne as the instructor and eventually with a list of specific students enrolled. Generally, both students and instructors will be associated with multiple classes, and these have to fit in a schedule to avoid overlapping times and so on. These things seem obvious to most of us, but that is the nature of domain knowledge. When you have it, it seems so natural and obvious that you often don't even realize that the things you know are actual knowledge. This can get you far on the road to good data analysis in the right kind of data.

Domain knowledge has to come from somewhere. In the class example, it clearly comes from living the experience. But a lot of the time, you aren't going to already have it and will need to develop it. This is especially true in the business world. Sometimes you can find material online or in books to learn what you need to, but often the only way to learn what you need is to find someone who already has the knowledge and can help you learn it. Such people are often called *subject matter experts*, or SMEs. It is true that you can sometimes figure things out purely by looking into the data, but you have to be very careful with this and must always validate your assumptions with an expert.

One of the challenges of working with a subject matter expert is that they often don't know what they know. If they were to describe a process or a system, they would likely leave steps or assumptions out because they are second nature or instinctive for them, so they don't think they're worth mentioning. For instance, in the class example above, imagine if they forgot to mention that instructors teach multiple classes. If you were completely unfamiliar with how class scheduling works, you might make the mistake of assuming each instructor only has one class or that an instructor teaches all sections of a given class. It seems impossible that anyone could think that, but someone who has no experience with secondary or higher education might not know. These kinds of incorrect assumptions can be devastating in data analysis, leading you down completely wrong paths.

The nice thing is that domain knowledge is usually transferable to some degree. If you work in the retail space for one company, you might go to another retailer and find that you aren't starting from zero again.

CRISP-DM: The Data Analysis Process

Although data analysis projects are all a little different, there is a general process to manage such projects. We often call this the data analysis lifecycle. Different data analysts may use slightly different processes, but many teams use a process called *CRISP-DM* (CRoss Industry Standard Process for Data Mining) for both data analysis and data science with only slight differences between them. It has the following six steps:

1. Business understanding
2. Data understanding
3. Data preparation
4. Exploratory analysis and modeling
5. Validation and evaluation
6. Visualization and presentation

Although these steps appear linear, there is a lot of iteration within the process. We don't just finish one step and move on to the next and never look back. Figure **5-3** shows how iterative the process can be.

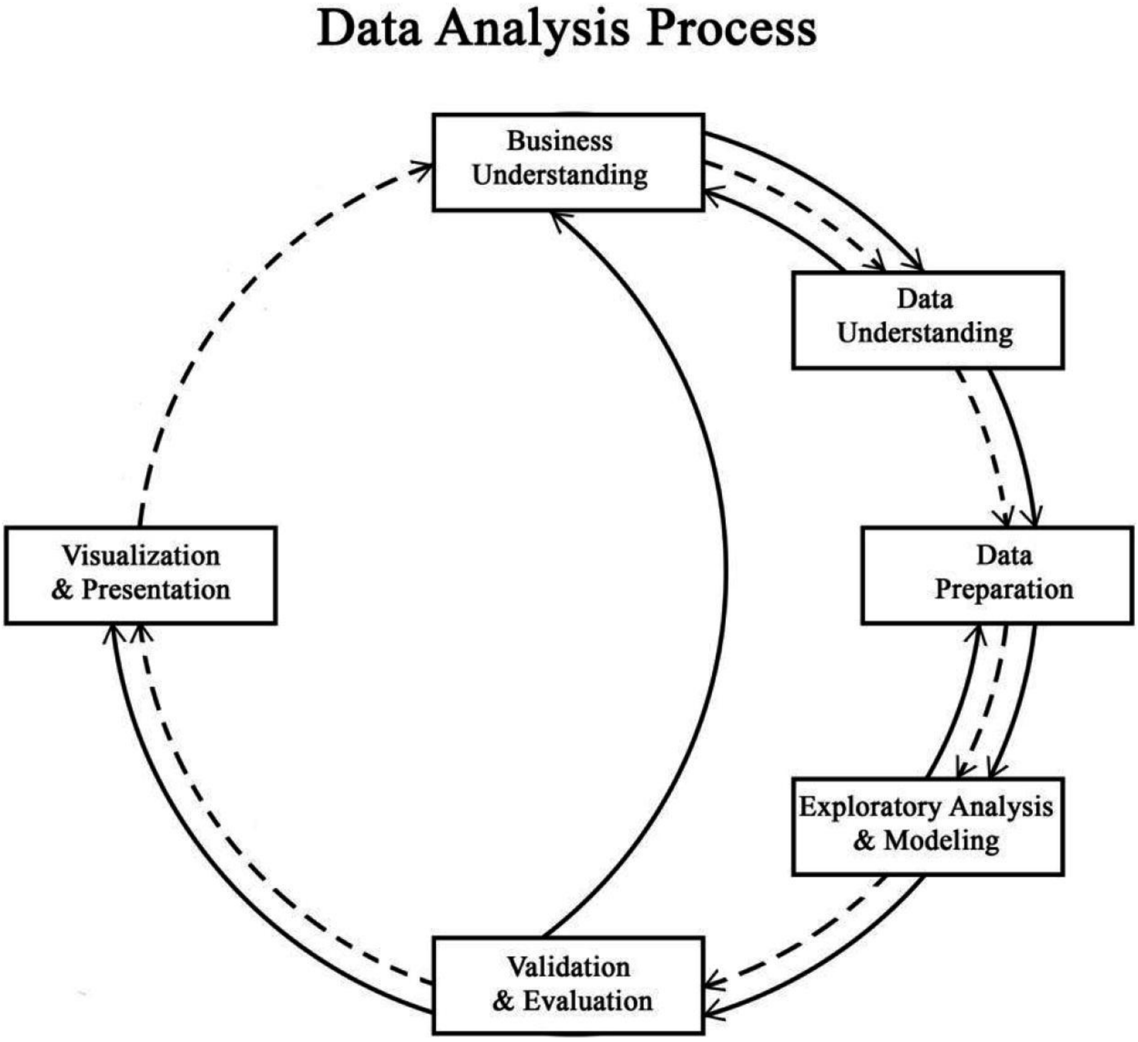

Figure 5-3

CRISP-DM data analysis process

We will go over the basics of each step here, but see Chapter 21 for more details on the process.

Business Understanding

We always have to start with business understanding, which means understanding what your customer is trying to learn and defining your research questions. In other words, what business problem are they trying to solve and how will you investigate that? For instance, a football team might want to know which of their defensive players to trade. This would imply you'd want to look at some performance metrics on all the players. You'll have to work with the business to identify and define those, and you'll also have to determine if you have data for those metrics. This step often occurs in conjunction with the next, data understanding, and you may find yourself going back and forth between them a lot in the beginning. Talking to customers to try to generate research questions is called requirements gathering, something done in a lot of different disciplines. But this will involve more than quickly asking them what they want. You will have to begin developing domain knowledge during this process, if you don't already have it.

Data Understanding

After you have a good understanding of what the customer is looking for, the next step is to understand the data that is available. This also implicitly includes finding the data in the first place. Most data analysis work is done by teams that have access to established data sources, so you will likely already have access to these common sources. Before you identify the specific data sources to use, you need to know—in a general sense—what data you need to answer the research questions you came up with in the previous step. Imagine in our example that you and the business have identified four things to track among the defensive players: tackles, sacks, interceptions, and fumbles.

Once you know what data you need, you look for data sources that contain that kind of information and then start investigating the specific data source(s). If you're lucky, you'll have a data engineering team that can give you exactly what you need, or there will be documentation like a data dictionary for your source(s), but often you won't be so fortunate. It's also not always clear what exact fields are important, so this can be an iterative process. It should also be mentioned that once you have obtained the data you want, you may find that it doesn't quite allow you to answer the research questions you came up with. You might tweak the questions, but you will likely need to check with the business to make sure your new versions still will give them what they want.

In the football example, imagine that we found sources that have each players' tackles, interceptions, and fumbles, but nothing that specifically contains sacks (a specific type of tackle, on a quarterback). However, you've found another table that has timestamped tackles with a record of what position was tackled but not the player who carried out the tackle. You look again at your original tackle data and see it has some times recorded. You conclude that it might be possible to line these two sources up, but you're not sure it can be done or how much time it will take. This is when you would discuss it with your business stakeholders—how important is the sacks figure, and is it worth the time investment to extract? You wouldn't necessarily be able to answer this before going to the next two stages, but it's appropriate to talk to the stakeholders first. They might just tell you sacks aren't that important for this question.

Note that like with the example, you'll be working with the data some at this stage, but you won't really dig into it until you're working on the data preparation.

Data Preparation

Once you are comfortable with your understanding of the business needs and believe you have the right data, the next step is data preparation, which can take up a huge proportion of the process. There will be a later chapter dedicated to this topic, but data prep involves many things. It's everything from cleaning up messy text fields that have trailing white space, making sure your data in numeric fields is all in the expected range, identifying and dealing with missing or null values, and much more. We'll go more in depth in Chapters 13 and 14.

If you're at a company with data engineers who have prepared nice, clean data for you in advance, lucky you. In that case you may be able to skip some of this stage, but more than likely you'll still have some work to do here.

If the project will involve any modeling, you'll also need to do feature engineering, where new features are created based on existing data source(s) in order to improve modeling or analysis. Knowing what to create usually relies on what's discovered in the next step, so there is some back-and-forth with these two, as well. Feature engineering is almost always necessary in data science, even when we have clean data from a data engineering team, but it's not needed as often in data analysis. Determining what features are needed is its own process and is based on both how the data looks and what techniques they're going to be used in.

It's worth mentioning that there can be a lot of back-and-forth between this step and the next, exploratory analysis and modeling, because you may find things in the next step that require you to do additional data prep or change some of what's been done.

In our football example, some of the data prep might be creating a new table with all of the basic stats with player names and replacing null values with 0 based on your stakeholders confirming that is the right business logic. You would also make sure to save all the numeric fields as numeric data types. You may still be wondering about the possibility of deriving sacks from the two data sources. You likely wouldn't have enough knowledge to do that yet and will rely on what you find in the next step to determine if it's possible and necessary.

Exploratory Analysis and Modeling

After you've got the data ready, it's time to start doing what you've been wanting to do the whole time: dig into the data. Exploratory analysis and modeling is often another big step. How big depends on how deep you go. But as mentioned, you will probably be going back to the data preparation step a few times after finding problems or little gotchas with the data that you missed when you did the cursory investigation in that step. In other cases, you may even have to go all the way back to data understanding. You may discover that you need to find entirely new data. This is totally normal and does not mean you've done something wrong.

Exploratory data analysis (EDA) is always the first major step in "looking at the data." This isn't a haphazard process of opening a data file in Excel and casually looking at it. Exploratory data analysis provides a framework for investigating data with an open mind. It isn't a rigid set of steps that must all be followed, but it helps guide you through your early analysis. The statistician John Tukey named this process in the 1970s, and data analysts have been following his guidelines since. EDA can be thought of as a mentality or attitude, too. We want to approach looking at the data with a curious mind cleared of assumptions and expectations, because we never know what we'll find.

So we know that EDA is important and relatively simplistic, but what exactly does it involve? Basically, EDA is descriptive statistics, as we covered in Chapter 2. We generate summary statistics and basic charts. Summary statistics tells us about "location" and "spread" and involves metrics like mean, median, mode, and standard deviation/variance. Typical charts include scatterplots, bar charts, histograms, box plots, and pie charts.

The initial EDA that we do often focuses on individual fields. The summary statistics will be done on individual fields, but it's not uncommon to make plots with multiple fields, especially with line charts and scatterplots. A variety of fairly simple charts and graphs are useful when doing EDA. These are often simply ways of seeing the measures of location and spread visually, which can be easier to interpret. Often things that are difficult to see in purely numeric data can be glaringly obvious when visualized. We might make a line chart of values in one field broken down by values in another field (with different lines for the breakdown field) or look at the distribution of values of a field in a histogram. Outliers may jump out at you.

One thing worth mentioning is that when you are creating charts as part of your EDA, you might not be super picky about including axis and chart labels. This is okay if they are for only you to see, but any time you are going to share a chart with someone else, even another colleague, it's best to label the axes at a minimum, and a chart title is always a good idea, too. If more than one type of data is included, a legend should be added. It might seem obvious to you what the axes and chart components represent when you are creating them, but it won't necessarily be obvious to other people—or even to you a few months down the road.

EDA is critical, and depending on the kind of project we're working on, there may be deeper analysis necessary like with a data analysis project, or we might proceed toward modeling directly like with a data science project. Or, even more likely, we might need to go back to the data preparation step before embarking on some more EDA. Learning about your data is usually an iterative process.

With the football data, you'd look at the distribution of the metrics we do have (tackles, interceptions, and fumbles), looking for outliers or other anomalies we didn't see during data prep. But we still have the question about deriving the sacks from the two sources. You would look into this closely, seeing if you can figure out how to line up the different times in the two files. You might have to do quite a bit of work to figure this out, which is why you want to make sure it's worth it to your stakeholders. But if you figure out how to do it, you'd move back to the data preparation step and add this derived feature to your table.

You may have bigger questions to answer after you've done your EDA, and by this point you should have a good sense for what variables are available to you in the data. You will mostly understand which features are reliable and accurate and what their limitations are. So you will have a sense for how far you can take your data in answering your questions, especially as they get more complex. Looking back over your questions and trying to create more is a good first step after EDA is done.

The exploratory analysis steps you will follow after doing EDA will depend entirely on the goals of your project. You will likely have research questions that haven't been answered yet, or you may have new requests from customers, both of which will guide you. The next steps would usually involve more complex ways of slicing and dicing the data to focus more on how different fields interact with each other and drilling down to understand the data that pertains to the specific research questions. Some of the work at this stage may be looking toward the visualization and presentation step, but not usually creating final deliverables yet.

While all of the steps discussed above are common in most data analysis projects, there are important advanced techniques that some data analysts will use. Some of these will overlap significantly with what people think of as data science's main techniques, which is why they're considered advanced here. We are not going to cover them since they will all be addressed in later chapters.

The most common advanced techniques that data analysts use are testing for correlation, basic hypothesis testing, significance testing, and other tests like t-tests and chi-squared tests, all of which are a part of statistics. Being able to understand these requires a basic comprehension of probability, as well as distributions, starting with the one most people have heard of—the normal distribution, otherwise known as the bell curve. You've seen all of this in Chapters 3 and 4.

Another common technique is linear regression, also based in statistics. This is visually a lot like drawing a line of best fit through dots in a scatterplot, but it is of course more complicated to create. It usually has more than two variables, which means it's difficult to visualize, but it's still easier to understand than a lot of other techniques. It is considered quite valuable despite being a relatively simple approach, because it is easy to explain and rather intuitive to most people.

You might be able to do a linear regression (or its sibling, logistic regression) if your stakeholders have given you a bit more info. Imagine they've rated some of the players between 1 and 10. You could create a linear regression model trained on the metrics you've been working with to calculate a rating for the remaining players. They might be able to use that to select a threshold, where anyone with a rating lower would be traded.

Any data analysis project (and most data science projects) will have exploratory analysis steps and some deeper analysis, but only a few have modeling. Modeling is most common in data science projects, but some more advanced data analysis projects may involve modeling, especially something like linear regression. Sometimes it may even involve modeling with machine learning. This overlap between data analysis and data science is just the nature of the field—the expectations and work vary tremendously job to job and project to project.

Validation and Evaluation

Once you've finished the analysis, you need to validate your work and evaluate your models if you have created some. This is where you "check your work," something you should never skimp on. You have to make sure that you haven't made any computational errors or used the data in any way that will lead to a misunderstanding of the real world. This is not trivial to do and often can take a lot of time.

While validation is about ensuring that the work that was attempted was done accurately, evaluation has us considering if the work we did is the right work and whether it answers the questions the stakeholders had. There are different techniques for evaluation, and peer reviewers will also be evaluating your approaches during review. In *peer reviews*, coworkers look at each other's work to identify problems before they become permanent. Peer reviewers will look at the steps you took to produce the analysis and any code you wrote to help you identify any shady assumptions, bad logic, or coding errors.

Although sometimes validation and evaluation aren't done until after the work is "done" (it's not truly done until we know it's right), it's also common to check things along the way, and many teams will review each other's work while it's still in progress. It's often a good idea to seek input from colleagues to make sure you're on the right track. Even the best data analysts and data scientists make mistakes along the way—often small, but not always. Peer reviewing can be a great way to avoid going down a rabbit hole. Data science in particular is a very collaborative field.

You'd need to validate all the transformations you did on the football data in the data preparation step and any of the charts you created or tests you ran in the previous step.

Visualization and Presentation

Finally, we are at the last step. This is all the way over on its own on the left side of the diagram because it’s quite different from most of the other steps. This is where you compile all your results and present them to your customers. Visualization is usually a huge part of a data analyst’s job. It’s standard to prepare a lot of charts during the exploratory analysis and modeling step, but this step is about presenting to your stakeholders. Visualizations are important, but often there won’t be too many because you don’t want to overwhelm the stakeholders with too much information.

Visualization is an important part of presentation, but usually there’s more to it. How this is done can vary widely at different companies and also with whom we’re presenting to, as well as the formality of the project. If this is a one-time project, you might be able to get away with showing some of the visualizations you created in the exploratory step. Some companies want everything in slide decks, so you might create a presentation that highlights your findings, including charts you created. Others might prefer Word documents. A lot of people are hung up on Excel and like to see the raw data, so you might create an Excel sheet with all your data and charts in it. (Excel is the bane of many data professionals’ existence.) More technical companies may use Python notebooks or create a web app. There is a chapter on visualization and presentation later in the book that will discuss these topics more.

If this work was foundational for providing ongoing information to stakeholders, it might involve creating a dashboard that’s automatically refreshed with updated data on some regular cadence. Some data analysts do create dashboards in tools like Tableau or Power BI and set up automatic data refreshes, but at a lot of companies there would be another team that would do that work, based on requirements you give them.

With your football results, a logical thing to show would be a table containing player ratings and their other metrics, ranked high to low. Other interesting trends or views could easily have been found during exploration.

Key Takeaways and Next Up

This chapter introduced data analysis, both a field in its own right and a subset of the work most data scientists do. We covered the major types of skills data analysis requires, including functional and technical skills like ways of thinking and specific tools or techniques, soft skills like communication and other people skills, and domain knowledge, which is having a breadth of knowledge about the world that given data represents. Good data analysis is systematic and generally follows a process model called CRISP-DM, which outlines six major steps: business understanding, data understanding, data preparation, exploratory analysis and modeling, validation and evaluation, and visualization and presentation. The major tasks in each of these steps were covered, although we will dig into each more deeply in later chapters.

Coming next is the chapter on data science, where we’ll talk about how it developed as a relatively new field, address the values that define data science, talk about how organizations see data science, and finally talk about data scientist qualities and roles.

PRACTITIONER PROFILE: SANDIP THANKI

Name: Sandip Thanki

Job Title: Data scientist

Industry: Academia

Years of Experience: 13

Education:

- PhD Physics and Astronomy
- MS Physics and Astronomy
- BS Physics and Astronomy

The opinions expressed here are Sandip's and not any of his employers', past or present.

Background

Sandip Thanki is a physicist focusing on astronomy by training, earning a series of degrees culminating in a PhD in Physics and Astronomy. He studied stars and began his career as a professor. After becoming a department chair, one of his colleagues was a dean who worked with data a lot. He often pulled data for her, and he ended working with it on his own as he became curious. He moved into another role that required even more data work, and he loved it.

Work

Although Sandip started out working with student data as an academic administrator, he considers himself a true data scientist. He jokes that it's not that different from when he was working in astronomy—the job is the same as back then, saying that it's just the table attributes that have changed—now rows are students and columns are GPA, course credits, and financial aid status, while before rows were stars with columns of brightness and size. But this job is also different because he can actually do work that can change students' lives for the better, which he finds incredibly rewarding.

His job involves a lot of different activities, and he does a lot of reporting and answering ad hoc queries from a huge variety of stakeholders. Although it might not be obvious, his work often has high stakes because hundreds of thousands—even millions—of dollars can be on the line through grant proposals and other reports that have legal ramifications. He's always careful and methodical, but for those projects he triple-checks his work.

Sound Bites

Favorite Parts of the Job: Sandip loves the fact that his work can make a positive difference in people's lives. He also loves that every day is different and he never knows what interesting questions people will ask him.

Least Favorite Parts of the Job: He finds some of the queries he gets boring and basically meaningless when they're just being used to check mundane boxes. These are things like generating percentages of different ethnicities to get dumped into some report, rather than to drive efforts to improve opportunities for disadvantaged students.

Favorite Project: Sandip has lots of relatively small efforts that he's proud of. One involved digging into data to identify students who had stopped attending college even though they had been successful while attending and were close to completion. The simple SQL query he ran will lead to people's lives being changed after outreach efforts can help pull some of those students back into school and help them graduate.

How Education Ties to the Real World: The astronomy data he worked with wasn't totally clean, but next to data on people it was pristine. With students, there are so many ways to break things down and much more room for error in the data.

Skills Used Most: People skills are hugely important. The data science department is often intimidating to people, so being helpful—especially prompt and transparent—really helps build trust. Often stakeholders don't know what they need. He has learned to suss out the real requirements when people ask him for information—he knows how to ask the right questions so they can, in turn, ask the right questions of him. The last critical skill is methodical thinking and behaviors like taking good notes, documenting your work, and organizing everything. The ability to refer back and even reuse prior work is a huge time-saver.

Primary Tools Used Currently: SQL daily, Tableau weekly, and R and Python occasionally

Future of Data Analysis and Data Science: Sandip thinks that data is currently underutilized and more will be used to benefit people in the future once we've figured out better ways of anonymizing it. For instance, maybe it would be possible to warn drivers that an erratic driver is approaching them.

What Makes a Good Data Analyst: Patience, keeping the goal in mind, and being highly organized.

His Tip for Prospective Data Analysts and Data Scientists: His top two are (1) keep your stakeholders in mind at all times and (2) value simplicity over complexity. Only put a few charts on a dashboard and share spreadsheets with no more than a couple sheets. People are often turned off by complexity.

Sandip is a data scientist working in academia to help faculty and staff know how to best help students succeed.