Chapter 1: Unleashing The Securing Power Of Data

“It ain’t so much the things we don't know that get us into trouble. It's the things we know that just ain't so.”

Josh Billings, Humorist

This book isn’t really about data analysis and visualization. Yes, most every section will be focused on those topics, but being able to perform good data analysis and produce informative visualizations is just a means to an end. We never (okay, rarely) analyze data for the sheer joy of analyzing data. We analyze data and create visualizations to gain new perspectives, see relationships we didn’t know existed or to simply discover new information we didn’t have before. In short, we do data analysis and visualizations to learn, and that is what this book about. We want to learn how our information systems are functioning, or more importantly how they are failing and what we can do about it.

The cyber-world is just too large, has too many components and has grown far too complex to simply rely on our intuition. We believe it is only by augmenting and supporting our natural intuition with the science of data analysis that we will be able to maintain and protect our ever-growing and increasingly complex infrastructure. We are not advocating replacing people with algorithms, we are advocating arming people with algorithms so they can learn more and we can do a better job. The data contains information, and we believe we can learn better with the information in the data than without it.

This book will focus on using real data -- the types of data most of the readers may have come across in their work. But rather than focus on huge discoveries in the data we have made the decision to focus on the process and less on the result. As a result of that decision, the use cases are intended to be exemplary and introductory rather than knock-your-socks-off cool. Our goal here is to teach you the reader new ways of looking at and learning from data. Therefore, the analysis contained here is intended to be new ground in terms of technique for the reader, not necessarily in conclusion.

A brief history of learning from data

We have a rich history of learning how to learn from data and then doing great things with that knowledge. By looking backwards and understanding where we are coming from may help establish the context.

19th Century Data Analysis

Prior to the twentieth century, the use of data and statistics was still relatively undeveloped. While great strides were made in the 18th century, much of the scientific research of the day would calculate basic statistics and use those as evidence for the validity of the hypothesis. The inability to draw clear conclusions from noisy data (and almost all real data is more or less noisy) made much of the scientific debates more about opinions of the data than the data itself. One such fierce debate[[1]](#footnote-1) in the 19th century was between two medical professionals in which they debated (both with data) the cause of cholera, a bacterial infection that was often fatal for the victim.

The cholera outbreak in 1849 London was especially brutal claiming over 14 thousand lives in a single year. The cause of the illness was unknown at that time and two competing theories from two competing researchers emerged. Dr. William Farr a well-respected and established epidemiologist argued that cholera was caused by air pollution created by decomposing and unsanitary matter (officially called miasma theory). Dr. John Snow, also a successful epidemiologist who was not as widely known as Farr, put forth the theory that cholera was spread by consuming water that was contaminated by a “special animal poison” (this was prior to the discovery of germs) and the two debated for years.

Farr published the “Report on the mortality of cholera in England 1848-49” in 1852 in which he included a table of data with eight possible explanatory variables collected from the 38 registration districts of London. In the paper, Farr had done some relatively simple (by todays standards) statistics and established a relationship between the average elevation of the district and cholera deaths (lower areas had more deaths). While there was also a relationship between cholera deaths and the source of drinking water (another one of the eight variables he gathered), he had concluded that it was not nearly as significant as the elevation. Farr’s theory had data and logic and was accepted by his peers and it was adopted as fact of the day.

Dr. John Snow was passionate and vocal about his disbelief in Farr’s theory and relentless in proving his own theory. It’s said he even collected data by going door to door during the cholera outbreak in the Soho district of 1854. It was from that outbreak and his collected data that he made his now infamous map in Figure 1. The hand drawn map of the Soho district included little tick marks at the addresses where cholera deaths were reported. Overlaying the location of water pumps where residents got their drinking water showed a rather obvious clustering around the water pump on Broad Street. With his map and his passionate pleas, the city did allow the pump handle to be removed and the epidemic in that region subsided but this wasn’t enough to convince his critics. The cause of Cholera was heavily debated even beyond John Snow’s death in 1858.

The cholera debate included data and visualization techniques (long before computers) yet neither had had been able to convince the opposition. The debate between Snow and Farr was re-examined in 2003 when statisticians in the UK evaluated the data Farr published in 1852 with modern methods. They found that the data Farr pointed to as proof of an air born cause actually supported Snow’s position. They concluded that if modern statistical methods were available to Farr, the data he collected would have changed his conclusion. The good news of course, is that these statistical methods are available to us today.

20th Century Data Analysis

A few years before Farr and Snow debated cholera, an agricultural research station north of London at Rothamsted, began conducting experiments on the effects of fertilizer on crop yield. They spent decades conducting experiments and collecting data on various things such as crop yield, soil measurements and weather variables. Like many modern-day logging implementations, they gathered the data and diligently stored it, but they were unable extract the full value from it. Finally, in 1919 they hired a brilliant young statistician named Ronald Aylmer Fisher to pour through more than seventy years of data and help them understand it. Fisher quickly ran into a challenge with the data being confounded and he found it difficult to isolate the effect of the fertilizer from other effects such as weather or soil quality. This challenge would lead Fisher towards discoveries that would help forever change not just the world of statistics, but the most every scientific field in the twentieth century.

What Fisher discovered (among many revolutionary contributions to statistics) is that if an experiment was designed correctly, the influence of various effects could not just be separated, but also measured and their influence calculated. With a properly designed experiment he was able to isolate the effects of weather, soil quality and other factors so they could compare the effects of various fertilizer mixtures. And this work was not limited to just agriculture, the same techniques Fisher developed at Rothamsted are still widely in use today in everything from medical trials to archaeology dig sites. Fisher’s work, and the work of his peers helped revolutionized science in the twentieth century. No longer could scientists simply collect and present their data as evidence of their claim as they had in the eighteenth century. They now had the tools to design robust experiments and new techniques to model how the variables affected their experiment and observations.

The world of science now included statistical models and much of the statistical and science education has focused on learning, developing and testing these models and the assumptions behind them. Most every statistical problem started out with the question “what’s the model?” and ended with the model populated to allow description and even prediction using the model. This represented a huge leap forward and has enabled research never before possible. If it weren’t for computers becoming ubiquitous, the world would probably still consider these techniques to be modern. But computers are ubiquitous and they have enabled a whole new approach to data analysis that was both impossible and unfathomable prior to the technology.

21st Century Data Analysis

It’s difficult to pull out any single event or person that captures where data analysis is today like Farr and Fisher capture the previous stages of data analysis. The first glimpse at what was on the horizon came from John Tukey who wrote in 1962 that data analysis should be thought of as different from statistics (though analysis leveraged statistics). He stated that data analysis must draw from science more than mathematics (can you see the term “data science” in there?) Tukey was not only an accomplished statistician having contributed numerous procedures and techniques to the field, but he was also an early proponent of visualization techniques for the purpose of describing and exploring the data. We will come back to some of Tukey’s work later in this chapter.

Let’s jump ahead to a paper written in 2001 by Leo Breiman, a statistician who focused on machine learning algorithms (which we will focus on in chapter 10). In the paper he describes a new culture of data analysis that does not focus on defining a data model *of nature* but instead derives an algorithmic model *from nature*. This new culture has evolved within computer science and engineering largely outside (or perhaps along side) traditional statistics. New approaches are being born from the practical problems created by the information age, which has brought large quantities of complex and noisy data. The revolutionary idea that Breiman outlined in this paper is that models should be judged on their predictive accuracy instead of validating the model with traditional statistical tests (which are not without value by the way).

At face value we may think of testing “predictive accuracy” by gathering data today and see how it predicts the world of tomorrow, but that’s not what the idea is about. We want to split the data of today into two data sets. We use the first data set to generate (or “train”) an algorithm and then we can validate (or “test”) its predictive accuracy on the second data set. To increase the power of this approach, we can iterate through this process multiple times, splitting our data into various training and test sets, generating and validating as we go. This approach is not well suited to small data sets, but works remarkably well with modern data sets.

There are several main differences between data analysis in the modern information age and the agricultural fields of Rothamsted. First, there is a large difference in the available sample size. “Classic” statistical techniques were largely limited by what the computers of the day could handle (“computers” were the people hired to “compute” all day long). With generally smaller samples, generating a training and test was impractical. However our modern environments are recording hundreds of variables generated across thousands of systems and large sample sizes are the norm, not the exception. Secondly, for many environments and industries, a properly designed experiment is unlikely if not completely impossible. We cannot divide our networks into control and test groups, nor would we want to test the efficacy of a web application firewall by only protecting a portion of a critical application. One effect of these environmental limits is a much higher noise-to-signal ratio in our data. The techniques within machine learning (and the related field of data mining) have evolved with modern data in mind. Finally, knowledge of statistics is just one skill of many that contribute to successful data analysis in the 21st century. With that in mind, let’s spend some time looking at the various skills and attributes that support a good data analysis.

Gathering data analysis skills

We know there is a natural allure to data science and everyone wants to achieve that sexy mystique surrounding security data analysis (oh yeah, we know). While we have focused on this concept of data analysis so far, it takes more than just analytic skills to create that mystique that everyone is seeking. We want to combine statistics and data analysis with visualization techniques, then leverage the computing power and mix with a healthy dose of domain (information security) knowledge. And all of that begins not with products or tools, but with our own skills and abilities.

Before we get to the skills though, there are a couple underlying personality traits we see in data analysts that want to discuss: curiosity and communication. Working with data can at times be a bit like how we imagine archeology: spending hour after hour with small tools in the hope of uncovering even the tiniest of insights in the dirt. So it is with data analysis: pearls of wisdom are nestled deep within data just waiting to be discovered and presented to an eagerly awaiting audience. It is only with that sense of wonder and curiosity that the hours spent cleaning and preparing data are not just tolerable, but somehow exciting and worth every moment. Because there is that moment, when we are able to turn a light on in an otherwise dark room, when we can describe some phenomenon or point in a direction to travel and that’s what we’re after. We are uncovering those tiny moments of enlightenment hidden in plain site for those who know where to look.

Once we turn that light on, we have to be able to bring others into the room for the discovery to have value. Otherwise, we will have constructed a house that nobody lives in. It’s not enough to point at our work and say, “see!” We have to step back and think of the best way to communicate our discovery. The complexity present in both our systems and analysis make it difficult to convey the results in a way that everyone will understand what we have discovered. Often times it takes a combination of words, numbers and pictures to communicate the insights in data. Even then some people will take away nothing and others will take away too much. But there is still a need to condense this complexity into a paragraph, table and/or graphic. Only then can we bring the value from the data to where it is desperately needed within information security.

While we could spend the entire book to create an exhaustive list of skills to be a good security data scientist, we are going to cover the following skills/domains that a data scientist will benefit from knowing within information security: **domain expertise** (setting and maintaining purpose to the analysis), **data management** (being able to prepare, store and maintain data), **programming** (the glue that connects data to analysis), **statistics** (to learn from the data) and **visualization** (communicating the results effectively). It’d be easy for us to label each one of these skills as the most important, but in reality, the whole is greater than the sum of its parts. Each of these contributes a significant and important piece to the workings of security data science.

Domain Expertise

Saying that a data scientist needs domain expertise should go without saying and it may seem obvious when we lay it out like this, but data analysis is most meaningful when performed with a higher purpose in mind. It is our experience with information security that will guide the direction of the analysis, provide context to the data and help apply meaning to the results. In other words, domain expertise is beneficial in the beginning, middle and end of all of our data analysis efforts.

If you are reading this book, it is probably safe for us to assume that you already see value in moving towards a data-driven approach in information security. Therefore, rather than spend the effort discussing the benefits of domain expertise in data analysis, we will cover some objections readers may likely encounter as other domain experts (or skeptical leadership) are brought into the data analysis effort.

**“People are smarter than models.”**  There are those who hold the opinion that people will always outperform algorithms (or statistics, or models) and there is some truth to this. Teaching a machine, for example, to catch a fly ball is a remarkably challenging and it may never outperform a person catching a fly ball. But as Kahneman and Klein point out in their 2009 paper, determining when people will outperform algorithms is heavily dependent on the environment for the task. If the environment is complex and feedback is delayed or ambiguous, algorithms will generally and relatively consistently outperform human judgment. So the question then becomes how complex is the security of our information systems and how clear is our feedback? When we make a change, or add a control, how much feedback do we receive on how well it is actually protecting the information asset?

The result is that information security occurs in a very complex environment, but that doesn’t mean we put all our eggs in the algorithm basket. What it does mean is that we should have some healthy skepticism about any approach that relies on purely human judgment and we should seek ways to augment and support that expertise. Notice how we worded that last sentence because the distinction we are making here is important. We do not want to compare algorithms to human judgment. We do not want to set up an either-or choice. We do, however, want to compare human judgment combined with algorithms and data analysis against human judgment alone. We do not want to remove the human element, but we want to be skeptical of unsupported opinion. In our complex environment, it is the combination of human intuition and data analysis that will produce the best results and create the best opportunity for learning and securing our infrastructure.

**“It’s just lying with statistics”** is meant to express a general distrust in statistics and data analysis, which are often abused and misused (and in some cases flat out made up) for the sake of serving some ulterior motive. In a way, this distrust is grounded in our collective knowledge of just how easy it is to social engineer people. However, we are in a different situation since our motive is (or should be) to learn from the data. We are sitting on mounds of data that hold information and patterns just waiting to be discovered. Not leveraging data analysis because statistics are misused is like not using a car because they are used as get-away vehicles. We need to be comfortable with adding statistics to our information security toolkit.

This is not to say that data analysis is infallible. There may be times when the analysis provides the wrong answer. Perhaps through poor data collection, under-trained analysts, a mistake in the process or simply using Excel (couldn’t resist) can lead to a misleading conclusion. But what we should see is simply less mistakes when we apply the rigor of data analysis with our expertise. Again, we arrive at the combination of data analysis and expertise.

**“This ain’t rocket science.”** This statement has two implications, first it says that whatever the problem is we’re trying to solve, we should be able to solve it with common sense. But this concern is goes back to our first point, thinking that people sitting around a conference table looking at a complex environment should be able to solve the (complex) problem without the need for data analysis. But as we saw, we should pull up a chair for the data as well as we are generally better off with than without it.

The second implication to the statement is that data analysis is too complicated and will cost too much (in either time, money or resources). This point is simply misinformed and the objection is more likely to be a concern about an uncomfortable change in practices than a concern about time spent with data analysis. Many of the tools are open-source (if the organization is averse to open-source there are plenty of commercial solutions out there as well) and the only real commitment is in the time to learn some of the basic techniques and methods we are showing in this book. The actual analysis itself can be fairly quick. At one point, I (Jay) was presenting the results of some data analysis and someone made a comment about how it would’ve been cool to see a specific relationship in the data. While still in the meeting, I was able to fulfill the request and create a descriptive visualization while sitting at the table. It was a quick and simple task and helped answer questions in the room.

**“We don’t have the data.”**  An alternate form of this objection is saying we don’t have actuarial-quality data (which is more prevalent when we start talking about risk analysis). Data-detractors argue that anything less than perfect data is nigh worthless to us and prevents us creating well-designed experiments. This statement is untrue and quite harmful to progress, for if we just wait around for perfect data, we will always be waiting and many learning opportunities will be missed. More importantly and to the heart of this objection, we don’t *need* perfect data; we just need methods to learn from the messy data we do have. As Douglas Hubbard wrote, “*The fact is that we often have more data than we think, we need less data than we think, and getting more data through observation is simpler than we think*.” So, generally speaking, data for security analysis absolutely exists, often times it is just waiting to be collected. And, even somewhat sketchy data can be collected with a few alterations and analyzed with modern data analysis methods that have evolved to work with noisy, incomplete and imperfect data.

**“But we will fall off the edge of the world.”** There is one last point we should discuss and it’s not so much an objection to data analysis, but an obstacle in data analysis. When we are seen as a domain expert, we are expected to provide answers with confidence. Which is fine, but a conflict arises within data analysis when confidence is confused with certainty. Data analysis requires just enough self-awareness and humility to create space for doubt in the things we think we know. Even though we may confidently state that passwords should be so many characters long with so much complexity, the reality is we just don’t know where the balance is between usability and security. Confidence needs to be balanced with humility and the ability to update our belief based on new evidence. This obstacle in data analysis is not just limited to the primary analyst. Other domain experts involved in the analysis will have to come face to face with their own humility. Not everyone will want to hear that his or her world isn’t flat.

type="caseStudy"

AES-256-bit keys are twice as good as AES-128, right?

One natural assumption about AES-256 bit keys is that they are twice as long as AES-128 bit keys, and, we’ve been around information security people when they force a project to use 256-bit keys because they are “twice as good”. Well, let’s look into the math for a bit. First, we are talking about bits here, and while 256 bits is twice as many bits as 128, it will have 2128 times more keys. But, let’s break out our slide rules and work through an exercise to try to answer a simple question: if we had access to the world’s fastest super-computer, how many 128-bit keys could we crack?

The world’s fastest super computer (at the time of this writing) is the *Tianhe-2* in China, which does around 34 petaflops (34 x 1015 floating point operations per second). If we assume it takes one operation to generate a key and one operation to test it (this is an absurd and conservative assumption), we can test an amazing 34 x 1014 keys per second. But a 128-bit key has 3.4 x 1038 possibilities, which means after a full year of cracking 128-bit keys, we will have exhausted 3 x 10-13 percent of the key space. Even if we run the super computer for a thousand years, we will only have searched 0.000000000003% of all the possible keys (and spent a fortune on electricity).

**To put this simply, the probability of brute-force cracking a 128-bit key is already so infinitesimally small we could easily round off that probability to zero.** But let’s be professional here and say, “moving from a 128-bit key to a 256 is moving the probability from really-super-duper-infinitesimally-small to really-super-duper-infinitesimally-small x 2128”.

Programming Skills

As much as we’d like to portray data science as a glamorous pursuit of truth and knowledge, as we’ve said, it can get a little messy. Okay, that’s an understatement. Working with data is a great deal more uncertain and unkempt than people think and, unfortunately, the mess usually appears early on when we are attempting to collect and prepare the data. This is something that many classes in statistics never prepare their students for. Professors hand out rather nice and neat data sets ready to be imported into the analysis tool du jour. Once we leave the comfort of the classroom we quickly realize that the world is a disorganized and chaotic place and data (and our analyses) are a reflection of that fact.

This is a cold-hard lesson in data science: data comes to us in a wide range of formats, states and overall quality. It may be embedded in unstructured or semi-structured log files. It may need to be scraped from a website. Or, in extreme cases, data may come in an overly complex and thoroughly frustrating format known as XML. Somehow, we must find a way to collect, coax, combine and massage what we’re given into a format that supports further analysis. While this could be done with the patience of Heracles, a text editor and judicious use of summer interns, the ability to whip together a script to do the work will provide more functionality, flexibility and efficiency in the long run. Learning even basic programing skills opens up a whole range of possibilities when we are working with data. It frees us to accept multiple forms of data and manipulate it into whatever formats work best with the analysis software we have. While there are certainly a large collection of handy data conversion tools available (appendix A), they cannot anticipate or handle everything we will come across. To be truly effective while working with data, we need to adapt to the data in our world, not vice versa.

Any modern language will support basic data manipulation tasks, but scripting languages such as Python or R appear to be used slightly more often in data analysis than their compiled counterparts (Java and C). However, the programming language is somewhat irrelevant, as the end results (and a happy analyst) are more important than picking any “best” language. Whatever gets the job done with the least amount of effort is the best language to use. We generally will flip between Python (pandas) and R for cleaning and converting data (or perhaps some Perl if we’re feeling nostalgic) and then R and/or pandas for the analysis and visualization. Learning web-centric languages like HTML, CSS and JavaScript will help create interactive visualizations for the web, as we’ll see in chapter 12, but web languages are not typically involved in the preparation and analysis of data.

There is a tool worth mentioning in this section, which we will label as a “gateway tool” between a text editor and programming known as the spreadsheet (e.g. Microsoft Excel, OpenOffice Calc). Spreadsheets allow non-programmers to do some amazing things and get some quick and accessible results. Though spreadsheets have their own sets of challenges and drawbacks, they also have some benefits. If the data are not too large or complex and the task is not deciding the future of the world economy (see case study), then Excel may be the best tool for the job. We would strongly suggest seeing Excel as a temporary solution though. It does well at quick one-shot tasks. But if there is a repeating analytic task or model that is used over and over, moving to some type of structured programming language is highly recommended.

As a cleaning tool, spreadsheets seem like a very good solution at first (especially for those who have developed some skill with them). But spreadsheets are event-driven, meaning they work through clicking, typing and dragging. If we want to apply a conversion to a row of data, we have to click to select the row and apply a conversion. This works for small data sets or quick tasks, but trust us, you will (more often than you think) have to go back to the source data and re-clean it. Either another day of log files needs to be processed, or we realize we should have pulled another relationship from the source data, or (gasp) we identified an error in the cleaning process. Something, somewhere, and probably more than once, will cause us to go back to the source and repeat the data cleaning and conversion. Leveraging a spreadsheet means a lot more clicking, while writing a script enables an easy, flexible and consistent execution of the cleaning process each time it runs.

type="caseStudy"

The Limits of Spreadsheets

On January 16th, 2013, J.P. Morgan issued a report to shareholders titled “Report of JPMorgan Chase & Co. Management Task Force Regarding 2012 CIO Losses” in which they investigate the loss of $6 billion in trades. They perform a detailed examination of the breakdown and describe the spreadsheet as a contributory factor. “During the review process, additional operational issues became apparent. For example, the model operated through a series of Excel spreadsheets, which had to be completed manually, by a process of copying and pasting data from one spreadsheet to another.” They also have uncovered a huge challenge with spreadsheets and that is consistency and integrity of the computations made in the data. “Data were uploaded manually without sufficient quality control. Spreadsheet-based calculations were conducted with insufficient controls and frequent formula and code changes were made.” They continue on and label the Excel-based model as “error prone” and “not easily scalable”. As with any complex system, catastrophe requires multiple failures[[2]](#footnote-2). We cannot point to their use of an “error prone” spreadsheet as the primary cause, but certainly it appears to have contributed in the loss of $6 billion.

After the data is ready for analysis, we can continue to benefit from understanding how to program as many of the languages we mentioned have robust data analysis features built into (or onto) the language. For example, statisticians developed the R language specifically for the purpose of performing data analysis. Python, with the addition of packages like NumPy, SciPy and pandas, offers a rich and comparable data analysis environment. But, preparing and analyzing the data is not enough. We also need to communicate our results and one of the most effective methods for that is data visualization (of which we devote several chapters to in this book). Again, Excel has the ability to produce graphics and with judicial modification of the default settings, good visualization can be done with Excel. However, in our opinion, flexibility and detail in data visualization is best achieved through programming. Both Python and R have some feature-rich packages for generating and exporting data visualization and we will cover some examples in later chapters. But in many cases, we may just combine all of these steps and functions in the same script. We can write one script to grab the source data, manipulate/clean it, and run the analysis on it and then visualize the results.

Data Management

If there was one skill we could hold off on learning, it’s data management, but we would only be able to dismiss it for a while. Within information security (as well as most other disciplines) our data can quickly multiply. If we don’t learn to manage it, the strain of ever-expanding data will take its toll on our efficiency and effectiveness. As we mentioned, we can leverage spreadsheets for the simple analyses. We will quickly outgrow that stage and should be resolved to expanding our repertoire to programing languages and simple formats like comma-separated value (CSV) files. At this point, we may see some benefits by moving our data into a database, but it still may not be necessary.

As the data repository grows, we reach a tipping point, either through the complexity of the data or the volume of data and moving to a more robust data management solution is going to be inevitable. There is a misconception that the large relational databases of yesteryear are reserved for the biggest of our projects, but that is no longer a helpful mindset. Many of the database systems we discuss in Chapter 6 can be installed on a desktop and help make the analysis more efficient and scalable. Once data management skills become more natural, their benefit can be used on even the smallest of projects. We’ve installed a local database and imported our data for some smaller one-time projects.

When we talk about data management skills, we naturally focus in on databases. We want to have enough knowledge to install a relational or NoSQL database to dump our data in and leverage that for our analysis. However data management is more than databases. Data management is also about managing the quality and integrity of the data. We want to be sure the data we are working with are not inadvertently modified or corrupted, either through misconfiguration of a tool or rushing through the cleaning and conversion a bit to quick. Whatever the reason, it doesn’t hurt to have some checks that keep an eye on data quality and integrity, especially over long-term data analysis efforts (e.g. metrics). It’s the like the concept of unit tests in software development where the smallest piece of testable code in an application are isolated from the larger body of code and checked to determine whether it behaves exactly as expected. We may want to automate some integrity checking of data after any new import or conversion and especially if the data analysis has sufficient efficacy to be performed regularly and used as a metric or control.

Finally, we work in information security and we’d be negligent if we didn’t talk about the security of the data for a bit here. But let’s take a step back for some context first. There seems to be a pattern repeating in our history: some passionate need drives a handful of geniuses to work their tails off to produce an elegant solution, but the security of their system is not their primary concern, meeting the functional need is. As an example, when the UNIX platform was first developed it was intended to be a shared (but closed) platform for multiple users who use the platform for programs they would write[[3]](#footnote-3). As a result, most of the authentication and permissions were constructed to protect the system from unintentional errors in their programs, and not from a malicious user. The point here is that “young” technology typically places an emphasis on functionality over security.

With the fast-paced and passionate push of the current data revolution we are definitely seeing the more emphasis on functionality and less on the security. New data management platforms such as Hadoop or NoSQL environments were not designed with many of the security policies or compliance requirements of most enterprise networks. The result is a distributed computing platform with some difficult security challenges. The authentication and security features are far better than the early days of UNIX, they typically do not compare to the security and features of the more established relational databases. We won’t focus too much on this point, but whatever data management platform is chosen, don’t assume the security is built in.

Statistics

Perhaps we are a little biased here, but picking up some skills around statistics will improve most every aspect of your life. Not only will it change the way to see and learn from the world around you, but it will also make you more interesting and probably even a bit more attractive to those around you. Seriously, though, statistics—as we are discussing it as a single skill here—is a very broad topic and quite a deep well to drink from. We use the term to describe the varied collection of techniques and methods that have evolved (and continue to evolve) to attempt to learn from data. These skills include the classic statistical approaches as well as newer techniques like data mining and machine learning. Luckily we can learn from the successes and mistakes of the generations of rather brilliant people who have worked with data very similar to ours, even if their calculations were performed with pen and paper versus transistors. Regardless of your personal belief in the utility of statistics and data analysis when it comes to information security, there is a vast amount of evidence showing it’s significant influence and benefit to most every other field of science.

Aside from the obvious “learning from data” approach, there are a few perhaps more subtle reasons to focus on improving our skills within statistics. First, even though data never lies, it is far too easy to be tricked by the data. We, as heuristic beings, are very capable of pulling out patterns and meaning from the world around us. Usually the ability to see subtle connections and patterns is helpful to us, and we use that skill on a daily basis. However, that skill can also mislead and we may think we see patterns and connections where none exist. Understanding statistics can raise awareness of this and the tactics can help minimize incorrect conclusions. Secondly, even though we just said data never lies, the way it’s generated and collected can create deceptive data, just as asking for the opinions of those around us may mistakenly confirm our own opinion since we naturally surround ourselves with like-minded people. Data itself may not be deceptive, but it’s quite easy to think the data means something it does not as in the story of the 1936 election polling.

type="caseStudy"

The magazine Literary Digest ran a large public opinion poll in an attempt to predict the 1936 presidential race. They gathered names from a variety of sources including the telephone directory, club memberships and magazine subscriptions. They ended up with over 2 million responses and predicted a clear winner: Alfred Landon (for those not up on their American history, the democratic candidate, Theodore Roosevelt, won that election carrying 46 states). The problem with the Literary Digest poll began long before a single response was collected or counted. Their trouble began with where they went looking for the data. Remember the year was 1936 and the great depression in the U.S. hadn’t let up yet and they ended up polling people with phones, club memberships and magazine subscriptions. They systematically polled the middle and upper class, which generally leaned towards Landon, and arrived at an answer that was mathematically correct and yet completely wrong.

The data did not lie. If they wanted to know which presidential candidate would get the most votes among Americans with a phone, club membership or magazine subscription, the data told an accurate story. However they weren’t looking for that story. They wanted to know about all registered voters in the United States, but through their selection of sources they introduced bias into their sample and drew meaning from the data that simply did not exist.

The fact that they had an unprecedented 2 million responses did not help improve the accuracy of their poll. Gathering more data with the same system flaw just generates a larger sample with the bias. To drive that point home, in the same 1936 election, a young man named George Gallup had gathered a relatively small sample of just 50,000 voters but he applied a much more representative sampling method and correctly predicted Franklin Roosevelt as the winner of the 1936 elections. The Literary Digest closed its doors a few years later but Gallup, Inc. is now an international organization, still conducting surveys and gathering data.

Statistics is not just a collection of tools; it is a collection of toolboxes each with their own set of tools. We can begin with descriptive statistics, which attempt to simplify the data into numbers that describe aspects of the data. For example, we can calculate the center of the data by calculating the mean, mode or median, we describe how spread out the data is with the standard deviation, or the symmetry of the data with skew or describe the width of peak with the kurtosis. But anytime we simplify the data, we will lose some level of detail and this is where visualization can serve us well. With visualizations, we create a single representation, or message, that can be simple and represent every data point, without simplification. We could think of this type of visualization as being a “descriptive visualization” since it is doing nothing more than simply describing the data to the viewer.

Aside from the challenge of over-simplifying, descriptive statistics is also limited to only describing the data we collect. It is not correct to simply scan a few systems, calculate the mean number of vulnerabilities and announce that the statistic describes all the systems in the environment. Inferential statistics helps us go beyond just describing our observations and enables us to make statements about a larger population given a smaller representative sample from that population. The key word there is “representative”. Statistics will teach us about the “design of experiments” (thanks to Fisher) and this will help us gather data so that we reduce the probability of being misled by the data. We want to have confidence that the samples we collect are representative of the whole and that lesson has been learned many times before we entered the scene.

We should also approach statistics with a healthy degree of respect and humility. As we slide more and more into the depths of applied mathematics, we will realize how easy it is to find meaning where none exists (technically called a “type I error”). But what is more important to understand here is that this error can occur with or without data. Even if we don’t fill a single cell in an Excel spreadsheet, we can make this mistake. The best tools in the toolbox are designed to limit the chance of these types of errors, but statistics alone is not enough. We need the combination of experience and data to decrease the chance of being misled. Errors can and will occur even with this combination, but we can reduce the frequency of these errors by applying the rigor and methods within statistics, to place ourselves in a much better position to learn from the mistakes when they do occur.

Having now built up the application of statistics on a pedestal, we should point out that much can be learned from data without advanced statistical techniques. We briefly mentioned “descriptive visualization” above. Take some time to look around at many of visualizations out there, they are generally not built from statistical models, but are just describing some set of data and showing the relationships therein. Snow’s map of the areas around the water pump on Broad Street in Figure 1.1 did not involve logistic regression or machine learning, it was just a visual description of the relationship between address and deaths. There is no doubt that we can improve our ability to secure our information assets with simple statistical methods and descriptive visualizations. All it takes is the patience to ask a question, gather the evidence, make sense of it and communicate it to others.

Visualization (a.k.a. Communication)

The final skill is what we are labeling “visualization” but really it is about the skill of communication. There are multiple ways to classify the types of visualizations out there, but for our discussion we want to talk about two general types of visualization, which are separated by who we want to read and interpret the visualization. The distinction we are making here is quite simple: 1) visualizing for ourselves or 2) everyone else. For example, Figure 2.2 shows four common plots used to diagnose the fit of a linear regression model (which we will run in chapter 5) and let’s face it, these are quite ugly and we would not include these in our next presentation to the board of directors. This type of visualization serves to provide information to the analyst while working with the data, or in this case about a data model.

Figure 2.2 Diagnostic plots for regression model of bot infections [FILENAME 793725c02f002]

These graphs are generated as a way to understand certain relationships and attributes of the model. They communicate from the data to the analyst and is used visually inspect for anomalies, strength of relationships or other aspects of the data for the purpose of understanding it better. Very little effort is spent on making these attractive or presentable since they are part of the analysis, not the result.

The other type of visualization exists to communicate from the analyst to one or more other people and serve to explain the story (or the lack of a story) the analyst uncovered in the data. These are typically intended to be attractive and carry a clear message, as it is a communication tool for non-analysts. Figure 2-3 (which we will learn how to generate in chapter 5) is derived from the same data as figure 2.2 but is intended for a completely different audience. Therefore, it is cleaner and we can pull a message for each of the 48 continental states from this one picture.

Combining the Skills

The skills we have listed here are what we want in order to make the analysis run smoother and improve what we can learn from the data while reducing our chances of being misled by the data. While we may have portrayed these skills as being in a single person, that is not a requirement. As the data grow and the demands for analysis gets more embedded into the culture, spreading the load among multiple individuals with partial combinations these skills will help lighten the load. Moreover, if you are just beginning to build your security data science team, you may be setting yourself up for an impossible task if you try to find even one individual with all of these skills. Take the time to talk through each of these points with candidates to ensure there is at least some element of each of the skills we talked through here.

Centering on a Question

It is natural to assume that the first step in data analysis is getting the data to analyze but true data analysis is never performed for its own sake. It is always performed within a larger context and understanding that context is the key to a successful data analysis. Just jumping in and grabbing data is like running a race without knowing where the finish line is. We want to have a good concept of what we’re trying to learn from the data. Therefore, every good data analysis project begins by setting a goal and creating one or more **research questions**. If you have ever come across a visualization or research and thought, “yeah, but so what?” that reaction is probably caused by the lack of a well-prepared research question in the analysis. Remember, the purpose of data analysis is to learn from our environment and learning can be done with or without data (with varying degrees of success). Creating and following a good research question is a component of *good* *learning*, not just of good data analysis. Without a well-formed question guiding the analysis, we may waste time and energy seeking convenient answers in the data or worse: *we may end up answering a question nobody was asking in the first place*.

For example, figure 2.3 shows the amount and categories of spam blocked at an organization during a given month. Thanks to the logs generated by an email filtering system, it is entirely possible to collect and show this information. However, the questions this data answers (and whatever subsequent actions it may drive) are of little interest to the typical organization. It’s hard to imagine someone looking at this graphic and thinking, “let’s understand why travel spam was up in December.” Outcomes like figure 2.3 are the result of a poor question selection and/or skipping a question altogether—it is data analysis for the sake of analyzing data and does not help to inform us about our environment in any meaningful way.

Figure 2.1 Amount of Spam by Category: The result of a poor research question [793725c02f001]

A good research question around spam might be, “*How much time do employees spend on spam that is not blocked by the spam filter*?” Just counting how much spam is blocked has little value since it will have no contextual meaning (i.e. nobody can internalize the effective difference between one-thousand or five-thousand spam emails). What we want to know is the impact spam has on employee productivity. While “productivity” may be a challenge to measure directly we can flip that around and just assume it is impossible to be productive when employees are reading and deleting spam. Therefore, what we really want to measure is time as it is directly spent dealing with unfiltered spam.

Now that we’ve framed the question like this, it’s pretty clear to see that we will not look to our spam filter logs to answer this spam-related question. We really don’t care that thousands of emails were blocked at the perimeter or even what proportion of spam is blocked. With a research question in hand, we now know to collect a measurement of employee time. Perhaps we would look for any logging from the email clients of events when users select the “mark as spam” option. Or perhaps, it’s important enough to warrant running a short survey in which we select a sample of users and ask them to record amount of spam and time spent going through them for some limited period of time. Either way, the context and purpose of the analysis is being set by the research question, not the data.

Steps to Creating a Good Research Question

Creating a good research question is relatively straightforward, but requires a bit of practice, critical thinking and discipline. Most research questions will serve as a pivot point for a decision or action (or inaction). Knowing the context of the result may also help determine what to collect. Going back to the spam example, maybe we learn there is some tolerance for wasted time. If so, maybe we don’t need to how much time is wasted, but if the time spent dealing with spam is simply above or below that tolerance. Planning the analysis with that information could change how data is sought and/or simplify data storage and analysis.

Usually we begin with some topic already in mind. Perhaps we are looking for the benefit of implementing a specific technology or we are trying to protect a specific asset or data type, or simply trying to increase our visibility into a network segment. Even if we just have a general sense of direction, we can begin by coming up with a series of questions or things we’d like to know about it. Once there is a good list of questions, we can whittle those down to one or just a few related questions. Now the fun really begins because we have to make those questions objective.

Let’s work through a simple example. The Human Resources department has submitted a proposal to expose an interactive menu for the company cafeteria to the Internet. While this may raise all sorts of questions around controls, processes and procedures, suppose the core security-oriented decision of the proposal is limited to either allowing authentication with the corporate username and password, or investing in a more expensive two-factor authentication mechanism. We may brainstorm a question like “How much risk does single factor authentication represent?” Or perhaps, “How effective is two-factor authentication?” These types of questions are really nice and squishy for the initial phase of forming a research question, but not well suited to serious analysis. We would struggle to collect evidence of “risk” or “effectiveness” in these questions. So we transform them to be more specific and measurable as an approach to inform the decisions or actions in context. Perhaps we start by asking how many services require single-factor versus dual-factor authentication. We’d also like to know how many of those services have had their authentication system attacked and with what success and so on. Perhaps we have access to a honeypot and can research and create a profile of internet-based brute force attempts. Perhaps we can look at the corporate instance of Microsoft Outlook Web Access and create a profile of authentication-based attacks on that asset. These are all good questions that are very answerable with data analysis with outcomes that can will help support a decision.

Exploratory Data Analysis

Now that we’ve laid out how a good data analysis should begin, lets talk about how things will generally occur in the real world. We’d love to start each day with a hot, caffeinated beverage, a clear research question and a bucket of clean data, but in reality we usually have to settle for just the hot, caffeinated beverage. Often times we do start off with data and a vague question like, “is there anything useful in this data?” This brings us back to John Tukey (remember we mentioned him earlier this chapter). He pioneered a process he called “exploratory data analysis” or EDA. It’s the process of walking around barefoot in the data, perhaps even rolling around a bit in it. We do this to learn about the variables in the data, their significance and relationships to other variables. Tukey developed a whole range of techniques to increase our visibility into and our understanding of the data, including the elegantly simple **stem and leaf plot**, the **five-number summary** and the helpful **box plot** diagram. All of which are explained or used later in the book.

Once we get comfortable with the data we’ll naturally start to ask some question of it. However, and this is important, we always want to circle back and form a proper research question. As Tukey said in his 1977 book, “Exploratory data analysis can never be the whole story” and refers to EDA as the foundation stone and the first step in data analysis. He also said, “Exploratory data analysis is an attitude, a state of flexibility, a willingness to look for those things that we believe are not there, as well as those we believe to be there.” With that in mind, most of the use cases in this book will be approached with exploratory analysis. We will take an iterative approach and learn as we walk around in the data. In the end though, we need to remember that data analysis is done to find an answer to a question worthy of asking.

1. And worthy of a bonified Hollywood plot as well: http://snowthemovie.com/ [↑](#footnote-ref-1)
2. See Richard Cook’s “How Complex Systems Fail” for a brief and wonderful dissection of this topic. http://www.ctlab.org/documents/How%20Complex%20Systems%20Fail.pdf [↑](#footnote-ref-2)
3. <http://www.cse.psu.edu/~tjaeger/cse443-s12/docs/ch4.pdf> and one of the first solutions for the UNIX platform was to simply store the users passwords in a clear text file on the system: <https://info.aiaa.org/tac/isg/SOFTC/Public%20Documents/Technical%20Working%20Groups/Cyber%20Security/Password%20Security%20A%20case%20Study.pdf> [↑](#footnote-ref-3)