Chapter 2: Beginning with a Question

“My job was to find questions about baseball that have objective answers, that’s all that I do, that’s all that I’ve done.”

-- Bill James, Sabermetrician

“To most people statistics means plugging numbers into an advanced calculator that spits out values, without much thought involved. Those people don’t work with data.” (“a life in statistics: Nathan Yau”, significance magazine).

It is natural to assume that the first step in data analysis is getting the data to analyze but 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. But before we get into what makes data analysis good, we want to discuss what skills contribute to a good data analysis. We know there is a natural allure to data science and everyone wants to achieve that sexy mystique (oh we know) surrounding data analysts, so we will begin this chapter by talking about how we achieve that mystique. There are multiple disciplines and skills that come together in the science of data analysis and we will walk through each one.

The two most important personality traits of a data analyst are 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 a mound of 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 worth every moment.

Once those pearls are discovered and cleaned up, they must be communicated to others. Make no mistake, the complexity of both our environment and analysis is difficult to convey to others. Often times it takes a combination of words, numbers and pictures to communicate the insights in data and that’s the personality trait that’s helpful, being able to condense complexity into a paragraph, table and/or graphic. Curiosity is especially important because doing data analysis well requires gaining and maintaining a broad mixture of skills. It takes some dedication and motivation to develop the other skills needed and with a healthy dose of curiosity the skills seem to develop effortlessly in an individual.

While it may be difficult to create an exhaustive list of skills to be a good data analyst[[1]](#footnote-1), we are going to cover the following skills/domains that data analysts benefit from knowing within information security: **domain expertise** (setting and maintaining purpose to the analysis), **data management** (being able to prepare, store and access data), **programming** (the glue that connects data to the analysis), **statistics** (to avoid being lied to by data) and **visualization** (communicating the analysis graphically).

Not all of these skills are required to be present in a single individual. While smaller shops may seek a single individual to cover these areas, as the work load increases it is entirely possible (and a lot more feasible) to spread these tasks across several individuals (or even individual teams). Wherever these skills come from, when they come together we create an environment where data analysis has its best chance at success. 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 contribute a significant piece of the puzzle.

Domain Expertise

Saying that a data scientist needs domain expertise should go without saying. It may seem obvious when we lay it out like this, but this cannot be talked about enough and there are some important points to discuss about the benefits and pitfalls within domain expertise. On one hand, domain expertise sets the context and purpose for data analysis and can help prevent or identify spurious results in the analysis. On the other hand, domain experts may push away data science, either as a perceived threat to their authority or because it may challenge long-held beliefs, viewed as conventional wisdom.

Creating a good research question, as we’ll see in the next section, is one of if not the most critical steps in any data-driven analysis. A good research question in information security requires an understanding of how systems work (and how they can break), how the attackers think and act and the tools they’ll be using and deploying. Attempting to develop the research question without good knowledge of the environment may lead to superficial and/or irrelevant findings. Therefore knowledge of information security is very important when forming that research question. But the research question also needs to be asked in a way that the data can answer. Asking questions about hypothetical constructs like “how much risk…” do not lend themselves well to data-driven analysis (but we’ll discuss research questions later in this chapter). The point here is that just domain expertise without knowledge of good data collection practices can result in chasing ghosts and doom the analysis before it even starts. Therefore, good data analysis requires knowledge of both information security and statistics so it has a valid purpose and is set up to make an impact.

Once the question is set and the data is gathered, domain expertise also brings context to data and analysis. Without an understanding of what the data represents and the relationships within them, problems can arise during the data munging and the analysis. At best, the lack of context may just waste time, at worst it could produce spurious results that would fail even a basic sniff test. Regardless, if the domain expert and data scientist is not the same person, countless back and forth communications (and sometimes some frustrations) can be generated here as the data scientist learns from the expert about the characteristics within the data that affect the analysis and the outcomes.

Finally, an understanding of the domain is essential in interpreting and assigning significance to the results. As we’ll see later in the book, statistics can only show correlation and can never prove causation. Causal relationships can only be established by an expert in the environment, and then only based on their understanding, observations and experience.

While we’d struggle to understand data without expertise in information security, it’s that same expertise that can present a challenge to good data analysis. Aside from any political or power struggle that may be come up, some people may see data science as silly threat, trying to replace their work and experience with models and formulas. This objection is not only misplaced, but also counter-productive. Statistics and all the related fields only have value in context and then as a supporting role. We need to mentally split expertise within an environment (like how we protect information systems) from the expertise of data analysis. We should not approach a statistician to present on identity management any more then we should approach an information security practitioner to collect and analyze data.

There is one more pitfall we have to discuss and it’s a bit of a challenge to write about knowing that the target audience for this book is the experts in the field of information security. There are times when expertise is built on some assumptions that appear logical, but later prove to be false upon closer inspection. Which is just a fancy way to say that experts can be wrong and even the possibility of that can cause friction. For example, we often hear that passwords should always be of a certain length and pull from multiple characters sets, but is this good advice? Florêncio and Herley from Microsoft Research collected data from 75 different websites and concluded a restrictive password policy “causes considerable inconvenience for negligible security improvement.” (http://research.microsoft.com/pubs/132623/WhereDoSecurityPoliciesComeFrom.pdf) As a data scientist, we must not only have awareness that these assumptions exist and are sometimes deep-rooted in the culture but that we must also have the confidence to challenge the conventional wisdom that has been built up in the industry over the years.

Programming Skills

As much as we’d like to portray data science as a glamorous pursuit of truth and knowledge, honestly it can get a little messy. Okay, that’s an understatement. Working with data is a lot more uncertain and messy then people think and unfortunately the mess usually appears early on when we collect and prepare the data. This is something that many classes in statistics never prepare their students for. The professors hand out rather nice and neat data sets ready to be imported into the analysis tool du jour. But once we leave the comfort of the classroom we quickly realize that the world is a disorganized and messy place and data are a reflection of the world.

This is the first cold-hard lesson in data science: data comes to us in a wide range of formats, states and overall quality causing us to spend a healthy portion of our time cleaning and preparing the data for analysis. This is where the ability to whip together a script comes in very handy. Learning even basic programing skills opens up a whole range of possibilities when we are working with data. It frees us to accept any form of data and munge it into whatever format we (or the analysis software) would want. There is certainly a large collection of data conversion tools available that can come in handy, but they certainly cannot anticipate or handle everything we will come across. To be really effective while working with data, we need to adapt to our data, not vice versa.

Our data may be embedded in semi-structured log files or maybe it needs to be scraped from a website, or in really bad cases, data comes in an overly complex and thoroughly confusing format known as an XML document. Somehow this data must be collected, coaxed and massaged into a format that supports further analysis. While this could be done with a lot of patience and a text editor, learning a programming language is way more efficient in the long run.

Most every modern language will support basic data manipulation tasks, but the scripting languages (python, R, perl) are used more often in data analysis then their compiled counter parts (Java, C). The programming language is somewhat irrelevant, as the results and a happy analyst matter way more then picking a “best” language. Whatever gets the job done is the best language to use. We prefer using Python for the cleaning and converting data (or perhaps some Perl if we’re feeling nostalgic, or want some unique package) and then R and/or Python 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 7, but web languages are not typically involved in the preparation and analysis.

There is a tool worth mentioning, which we will label as a “gateway tool” between a text editor and programming known as the spreadsheet (MS Excel, OpenOffice Calc). It allows non-programmers to do some amazing things and get some quick and accessible results. While spreadsheets have their own sets of challenges and draw-backs, they also have some benefits. If the amount of data are not too large 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. It does well at quick one-shot tasks. But if there is a repeating analytic task or model that is used consistently, moving to a 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 and 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 or quick data sets, but trust us, you will (more often then you think) have to go back to the source data and re-clean it. Either you got another day of log files, or you realized you could pull another data point from the source data, or you identified an error in the process. Something, somewhere will cause you to go back to the source and repeat the data cleaning and conversion. Leveraging a spreadsheet means a lot more clicking, while write a script to convert enable consistent and easy re-execution of the process.

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” (<http://files.shareholder.com/downloads/ONE/2532388207x0x628656/4cb574a0-0bf5-4728-9582-625e4519b5ab/Task_Force_Report.pdf>) in which they investigate the loss of $6 billion in trades. In an appendix they have this:

“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. In addition, many of the tranches were less liquid, and therefore, the same price was given for those tranches on multiple consecutive days, leading the model to convey a lack of volatility. While there was some effort to map less liquid instruments to more liquid ones (i.e., calculate price changes in the less liquid instruments derived from price changes in more liquid ones), this effort was not organized or consistent.”

“…the model was approved despite observed operational problems. The Model Review Group noted that the VaR computation was being done on spreadsheets using a manual process and it was therefore “error prone” and “not easily scalable.” …” (p. 105)

“CIO’s implementation of the model was flawed. CIO relied on the model creator, who reported to the front office, to operate the model. Data were uploaded manually without sufficient quality control. Spreadsheet-based calculations were conducted with insufficient controls and frequent formula and code changes were made. Inadequate information technology resources were devoted to the process. Contrary to the action plan contained in the model approval, the process was never automated.”

As with any complex system, catastrophe requires multiple failures[[2]](#footnote-2). We cannot point to their use of a difficult-to-understand spreadsheet as the primary cause, but certainly it appears to have been a contributory factor.

Throughout this book, we will have examples in a few different languages, but don’t think data analysis is limited to these languages or tools. They serve as an example of what’s possible and how we may go about solving specific problems within data analysis. Excel makes the list because it is fairly ubiquitous these days and many people attempt to leverage it already for data analysis. Python and R make the list because they are open source efforts and free to download and they are incredibly robust languages with many convenient features that make data analysis pleasurable.

One last note, programming is not just for data preparation, though we end up investing a lot of time and energy at that stage. Many of the languages we mentioned have robust data analysis features built into (or onto) the language. So the same programming language we used for data preparation can be used for the analysis itself. For example, the R language was developed by statisticians for performing data analysis. Python, with the addition of packages like NumPy, SciPy and Pandas offers a rich and comparable data analysis environment as well.

But we’re not done there, just 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 here). 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 that in later chapters.

Data Management

If there was one skill we may be able to skimp on, it’s data management, but we would do so at a high cost. Within information security our data points can quickly multiply and if we don’t learn to manage our data the strain of the ever-expanding data sets will take its toll on our efficiency and effectiveness. When we start out with simple data sets, we can leverage spreadsheets and text editors. We outgrow that stage quickly though and we must move into 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 we haven’t quite reached that painful point.

There comes a tipping point, either in the complexity of the data or the amount of data that we must move to a more robust infrastructure for our data. Let’s not be fooled here, the relational databases of yesteryear were 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 simply help make the analysis more efficient and repeatable. There have been analysis efforts that we’ve installed a local database and imported our data for a one-time project. No longer are databases so unwieldy that they can’t help with the smaller efforts as well.

When we talk about data management skills, we naturally jump to databases. We want to have enough knowledge to install a Mongo or CouchDB, dump our data in and leverage that for our analysis. However data management is more than databases. It also requires some work on data quality and data integrity. We want to be sure the data we are working with is not inadvertently modified or corrupted in some way. Also, we work in information security and we’d be negligent if we didn’t talk about that for a bit here.

There seems to be a pattern in technology: a passionate need drives a handful of geniuses to work their tail 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 a malicious user. Of course as the technology evolves and grow, security usually comes with more experience (and failures).

This focus-on-function is the same thing we are seeing in our current “data revolution” and new brand of NoSQL databases and data management tools. While the authentication and security features are far better then the early days of UNIX, they typically do not compare to the security and features of the more established relational databases. We will not dwell on this point though. Just keep in mind that data management is just as much about data security and it is making the data accessible and usable.

Statistics

If hacking skills are the gears of a watch, then math and statistics dictate how all those gears come together to give them purpose and meaning (okay, the analogy was stretched a bit there). We need a way to learn from the data we collect, but more importantly, we also need a way to know when the data may be misleading and when we cannot learn from the data. As with R.A. Fisher’s story at Rothamsted in chapter 1, we may have a lot of data, and perhaps some metrics were derived from it, but without understanding the lessons from statistics, we may have undue confidence in the data.

Generations of people have collected data and tried to learn from it. Nathan Yau, author and maintainer of flowingdata.com, summarized the common misperception of statistics, “To most people statistics means plugging numbers into an advanced calculator that spits out values, without much thought involved. Those people don’t work with data.” (“a life in statistics: Nathan Yau”, significance magazine). This makes a lot of sense if we stop and think about it. The world is messy and it produces messy data. There are complex interdependencies and variations not just in the world but also in our measurements of the world. It makes sense that the majority of statistical methods generate probabilistic statements. Intuitively, a simple “advanced calculator” would simplify the world to the point of being useless, which sadly is how some

hacking (media definition)

All of this is done within some domain, in our case, we’re talking about a rather environment of

The majority of statistics has advanced by new generations of statisticsians showing that the previous methods

developing slightly better ways not to be deceived by the data and hopefully more accurate methods

Statistics is not a precise science, despite its use of mathematics and that’s because the world i

combined with If we stop and think about it, there is no way

Most people are familiar with the basic descriptive statistics (mean, median, etc). But when it comes to

The uninitiated, especially in engineering-based industries like computer science, see statistics as a collection of math formulas and applying statistics is just plugging in the right numbers to get the right answer. This type of thinking paints statistics as a cold unthinking science and nothing could be further from the truth. Most people naturally understand that the real world is filled with complex interactions and natural variations. On top of that, we have variation in our measurements and measuring everything is often impractical. Therefore, an essential concept within statistics is probability.

One of the common questions within statistics is asking how likely our observations are given some assumption. Notice the word “likely” in that statement. The outcome of basic statistics is either to describe the data we’ve seen, which is not statements in probability but are limited in their applicability (and we will see in chapter 4), or we want to make inferential statements about a larger population from limited observations. Descriptive statistics is limited in applicability to scope of the observations we made, while inferential statistics is only capable of making probabilistic statements.

But statistics is more than applied mathematics and models. Statistics also helps us tackle challenges with data collection and how we design our experiments. Understanding how the data was generated and collected help us determine how much confidence we can place on our results. Poor data collection techniques all but negate any inferential outcomes. put into the data representing the world

limit the inferences we can make or in some cases the data is coming from and how it was generated how our analysis are constructed and the data collected can influence how much confidence we should have in the inference we can pull from the data.

Even though we separated out statistics into its own domain, this function is for all practically purposes, exclusively done with computers (wish someone would tell the educational system that little secret). So the same language that we used to prepare the data, may be the same platform we use to perform the analysis.

The choice of programming language is not all that important though scripting languages (python, perl, R) appear to be more prevalent over compiled languages (Java, C). Though most every modern language has its own strengths when it comes to data manipulation, our personal favorites are python and R.

Python is a

. . leads to the ability to programming is an important skill to develop.

Knowledge of programming, however rudimentary

Being able to scrape a web site, transfer

Having at least cursory knowledge of regular expressions (for pattern matching in text)

, and often has a combination of challenges that must be addressed before analysis can begin. Sometimes the (mention python, perl, regex).

The tasks to complete within a data analysis will range from situation to situation. On one hand, every data analyst will have to be comfortable handling and manipulation data (chapter 3) and knowledge (however basic) of programming can serve an analyst well. Also, some data outgrow operations in memory and require a a more robust data management platform (databases are covered in chapter 6).

However, being able to manipulate and transform data is the key among them. We are leaving that definition vague because the tasks are decidedly vague. Obviously we have simple data manipulation functions

Perhaps it is to understand how something is functioning now, or to project how it could be done better, or to compare disparate entities out what is our understanding of

* + aim to find out what makes up some phenomenon: substance, function and rationale (why).
  + comparative - relationships (relation, equiivalance, difference)
  + explanatory - correlation, conditionality, causality
  + normative - how something should be done (best case world)

Perhaps it is to support a critical decision or to define to the uncertainty within a critical decision or define supporting a decision Analysts that lose sight of the larger context do so with disastrous results. It is essential that data analysis occurs with the so

analysis should always be driven by

Need good transition here

Now let’s move on to the stages within data analysis:

Four stages of data analysis:

Goals and Questions (set context, questions, purpose)

Acquire Data (gathering and prepare data)

Analyze (data analysis, generate conclusion)

Learn (learning from feedback)

Picking a Question

This first stage is hands down the most critical part of any data analysis effort. The entire point of this stage is summarized in the opening quote from Bill James. We don’t opening quote to this chapter is from Bill James, who’s work influenced Michael Lewis’, *Moneyball*.

the most critical and … framing stage is where the “why” and the “what” of the analyses are defined and most analyses are doomed in this stage because it is rushed through or skipped altogether.. For example, figure 2.1 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 and actions that this data addresses are of little interest to the typical organization. Outcomes like figure 2.1 are the result of a poor frame selection and/or skipping the frame selection altogether.

Putting some thought towards the what and why will set up how we proceed through the rest of the stages. Continuing on with the spam example, even though organizations don’t care how much spam is successfully blocked, they may be keenly interested in how many valid emails are blocked as spam or how much time is lost dealing with spam that bypasses our filtering. After thinking about the purpose and context of the analysis we would have concluded that the logs from the email filter may not help our analyses and we should seek other sources the data to address that frame.

A good frame for data analysis will establish both a purpose and a context in which the analysis is performed. The purpose is the specific goal of the analysis, while the context accounts for domain-specific information otherwise lacking. For example, let’s say we want to learn more about the challenges with phishing emails. Just hearing the “what” and skipping any context, we may craft some phishing emails to see how many people are duped by our cleverness. We could then conclude phishing is indeed a problem and we should do something about it. But if we add context of a limited budget for user awareness training and we can only target a subset of employees. Now the data collection and analysis takes on a dramatically different direction. Now we have to understand attributes of who is falling victim instead of how many people clicked. It is essential that we establish and understand both the purpose and context of the analysis before we seek any data.

It may also be worthwhile to understand that any frame that is set will not cover similar situations. It is usually necessary to consider secondary questions or “further” areas to research. Often times data collection is costly and/or time consuming and it’s much easier to add one more question to a survey or pull out one more field in context from the data source then it is to go back later for the data.

frames simplify and organize, frames put a limit on the complexity. a challenge of over-framing - over-simplification, framing out possible off-shoots or discovery paths.do not define the problem with out thought about the assumptions, may overlook options.

Frames are limiting, which are both good and bad things. How a frame is set, will ultimately determine the range of possible conclusions that may be reached.

Careful margining frames, look at separate problems as separate analysis (see below on supporting future though). Not choosing a frame just means that a poorly thought out frame has been chosen (a frame has been chosen with little to no thought).

What proportion of viruses are caught by AV? (notice the proportion, meaning we want to know “out of how many” within the frame.)

Just choosing to count signature matches may be limiting, could we also correlate the payload of the virus?

We do not cover this step further in this book, however that should not diminish the importance of this step.

Framing does not mean draw a conclusion, expect to be proven wrong, expect to be consistently wrong. (may have a quote here)

(from Wikipedia: The Research Question serves two purposes:

1. it determines where and what kind of research the writer will be looking for and
2. it identifies the specific objectives the study or paper will address.

From me:

* The frame should be set at the outset, but not set in stone, it must be fluid and able to updated and modified as challenges in data collection or analysis arise.

From “Constructin Research Questions”:

* “Some researchers aim for prediction and explanation while others search for understanding”
* “A fundamental step in all theory development is the formulation of carefully grounded research questions.”
* “If we do not pose innovative research questions, it is less likely that our research oefforts will generate interesting and influential theories”
* “the primary discussion revolves arouns how to formulate feasible research question in a particular sequential order. We are advised to first define the topic (for example. leadership, adult vocationsl learning, diversity among male engineers, middle-class status ansxiety in UK higher ed ), then to clarify the domain of the research, that is, what objects should be studied (individuals, social interaction and so on), state a purpose and finally to decide the type of research questions such as descriptive, explanatory and prescriptive questions.
* Others considering contextual issues, such as how various stakeholders, may influence the formaultion of research questions.
* “from gap-spotting to problemization”
* “with the risk of stating the obvious, therefore, we think it is critical to spell out an answer, one which is simple but fundamental: **questions are the core ingredient in all knowledge development**.” (p. 10)
* Quote: Gadamer “the path of all knowledge leads through the question” (1994, 363) - follow up on this.
  + Gadamer, H.G. (1994) Truth and Method. New York: continuum. (first published in 1960)
* “It is important to note that research questions can be understood in terms of different levels and with more or less precision and focus. … we see research questions not necessarily as very detailed questions or as specific objectives (close to testing an hypothesis). Instead, we regard research questions as setting the somewhat broader intellectual motive of a study, whether it is empirical and/or theoretical, that is, the rationale and direction of a study.”
* Questions provide the necessary starting point and path for all forms of knowledge development. It is by asking questions that we are able to generate knowledge about things. Similarily, asking questions forms the basis of every kind of research investigation.
* not all questions are research questions. Questions must be “researchable” or “investigable” - “they can be investigated scientifically and answered empirically” (p. 11, good stuff here)
* “It is important to note that some questions, which appear to be genuinely open, are in fact ‘closed’ question in the sense that they do not open up the subject matter and instead preserve it. The prime example of closed questions is found in educational settings where the teacher pretends to ask students a genuinely open question, but already has a ready-made answer.” … “however a research questions can never be completely open…”
* Types of research questions: descriptive, comparative, explanatory, and normative.
  + “a central observation made by Dillon is that these questions are to a large extent hierarchically related to each other in the sense that descriptive questions are the most basic, followed by comparative questions, and then explanatory and normative questions”
  + Descriptive - aim to find out what makes up some phenomenon: substance, function and rationale (why).
  + comparative - relationships (relation, equiivalance, difference)
  + explanatory - correlation, conditionality, causality
  + normative - how something should be done (best case world)

Acquiring Data

Once we know what the “what” and “why” of the analysis, it’s time to seek the data that would best serve the frame of the analysis. The data collection is sometimes relatively straightforward. If we want to know (for some strange reason), the proportion of spam emails we block that offer discount prescriptions we can look at all of all of the blocked spam (known as the *population*) and count how many were in the prescription category. This method allows us to count and *describe* the proportion (*descriptive statistics is discussed later in this chapter)*. But if we want to know something a bit more useful like how much spam makes it past our filter, we may have a bigger challenge. We will not be able to gather the population of all emails in order to count the proportion of spam. Instead we could take a *sample* of emails and infer an estimate based on how many were spam (inferential statistics will be discussed in chapter 4). Perhaps we could take a sample of users and have them self-report how many spam messages they receive.

Once we move away from being able to count everything, we run into the need for a sample and how we gather the data will greatly influence the level of confidence we should have in the resulting analysis. For example, if we want to ask users about their level of spam and we ask those sitting near us in the I.T. department, we would be introducing bias into the sample because, as everyone knows, I.T. people are not representative of typical users. We will be discussing sample selection in chapter 4.

Let’s assume we’ve identified where we can gather data from and we’ve figured how the best method of collecting that data. Chances are really good that the data is imperfect and this is where the real world diverges from textbooks. Usually in statistics classes the sample data has either been completely prepared and is free of defects, or there is a single step to prepare it (ya know, as an exercise). In re

al life we have incomplete data, we have redundant data, and we also must cope with the problem of normalizing data since a single entity could be represented in multiple ways. The topic of data preparation will be covered in chapter 3.

1. Identify sources of data
2. Identify method of data collection
3. clean/munge the data

select a handful of people

want to answer and why we need it answered we can seek sources of data that support that direction. Again, it is critical that we don’t start with data and seek answers without a clear purpose. But

Taking a “encircling” approach

clean data is essential

protecting privacy and “sum of the whole is greater than the parts” effect of data aggregation

Sometimes data doesn’t come from systems but from stochastic models and/or just plain thinking.

What am I trying to do? What am I trying to decide or learn?

What can I meaure/monitor/observe to help that decision?

If I gather this data, will it represent what I’m measuring?

what I’m asking?

looking for data is like running a race by but even before the first datum

Exploratory Data Analysis

This is a slight variation on the four stages where the context or frame of the data analysis is not specifically defined before we begin. In this case we do begin with the data, but not to arrive at a conclusion but to arrive back at the framing of the analysis. In other words, exploratory data analysis helps us understand what questioned are answered are in the data, and we’d want to then use that to see if any of those questions are useful. If that exploration

of the data analyses is not known and before we are able to establish the frame of analysis we have to understand what’s in the data to begin with. So *prior to* establishing the frame of our analyses we may explore the data in an attempt to discover answers to questions we didn’t know to ask.

Acquire

Analyze

Frame

Acquire (if needed)

Analyze (refining)

Learn

Where is time spent? (acquire)

Exploratory data analysis … employes a variety of techniques to:

* Uncover underlying structure
* Extract important variables
* Detect outliers and anomalies
* Test underlying assumptions
* Develop parsimonious models
* Determine optional factor setetings
* <http://www.unitedbiosource.com/pdfs/webinars/20121031-exploratory-wasiakr.pdf>

1. For example, we may argue that playing a musical instrument helps teach the creative and critical thinking necessary for good analysis. But alas, that did not make it on the broad list of skills we cover here. [↑](#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)