Chapter 16

People

16.1 THAT QUESTION AGAIN – WHAT IS DATA ANALYTICS?

Very early in Chapter 1, we discussed the meaning of the term "data analytics" and some of its variants such as business intelligence, business analytics, and data science. The debate about the remit of these fields is not within the scope of this book. What we *can* say is that the debate has been intense for several years and there is still a lack of consensus. This reflects the reality that "data" is a very broad area that has grown in recent years with the explosion of data availability. It is like asking what can be done with concrete when you do not have a definition of a civil engineer or architect.

If you think of oft-touted data science success stories, you probably think of Internet companies, social media companies, and digital companies. All of these organizations are growing immense customer bases through the innovative use of data and the creation of software and hardware to glean insights from that data. If you work in other industries, different "data analyst" roles may spring to mind. You may work with quantitative analysts in a financial services setting. You may think of actuaries who work with statistics in the insurance industry. I personally know geologists who deal with significant volumes of core sample data and run complex parameterized simulations across it. Companies have been mining data about customer behaviors for over a decade (Linden et al., 2003).

With these examples, you see a spectrum of activities. At one end of this spectrum, you find what can be considered data engineering activities. Data has to be put somewhere. Engineering activities involve the storage, management, and maintenance of data, and access to that data by the team. This may also include preparations applied to that data to facilitate these activities and support subsequent data analyses.

After engineering, you find data wrangling. As the name suggests, this is the common fight with the data that many analysts bemoan. Data must be cleaned. Derived variables are calculated. The data is reshaped to suit a particular analysis. Unstructured content is enriched with entities.

Once the data has been reshaped, there is sometimes a modeling phase. The reality is that on many Guerrilla Analytics projects, it is a sufficient win given project timelines to be able to summarize and report on the data at

all – never mind building sophisticated models. However, when a model is required, it happens in this phase. A hypothesis is agreed and an appropriate model is selected, built, and evaluated. The model is then used to optimize a process and make predictions about the process.

Finally, the analytics results must be communicated to the customer. The format of this communication depends on the customer's requirements. It might be a conversation, a workshop, or a written report. Increasingly, it is a web application or dashboard where the customers can interact with the analytics to produce their own insights.

In the context of Guerrilla Analytics, a data analyst is anybody working with data to produce insight. If that data is an unstructured mass of website visits, then perhaps you use the current in vogue Big Data tools. If your data arrives with you in a fairly clean state and your job is to produce sophisticated models then you are more focused on the modeling process and reporting. If you are confronted with spreadsheets you probably spend a lot of time wrangling the data to get insight from it.

What makes you a Guerrilla Analyst is that you have to produce insight in dynamic circumstances while being quite highly constrained and facing frequent disruptions. This chapter focuses on the general skills of a Guerrilla Analyst rather than specific skills such as SQL, Hadoop, machine learning, or some of the other myriad technologies, languages, and techniques you may encounter.

16.2 GUERRILLA ANALYTICS SKILLS

Drew Conway's "Data Science Venn Diagram" (Conway, 2013) is an interesting perspective on the skills required in modern data analytics. The three expertise areas he identifies are as follows:

- **"Hacking" or programming:** This emphasizes the ability to access data and to think algorithmically. Data comes in a huge variety of forms and is stored in many ways. Team members need the skills to deal with this. They need to be able to get at this data and clean and manipulate it. This does not mean being an expert in computer science who can design compilers. Indeed it does not mean being an expert in database design and maintenance. It is a little bit of everything needed to get the job done. They need to recognize common database structures. They need to model data sensibly. Once data has been shaped into a good format, analytics is usually performed. This is where knowledge of algorithms and awareness of computer science topics such as complexity and data structures is advantageous. An appropriate data structure and an algorithm design can be all the difference between a feasible execution time and analyses that take hours to finish. The former is obviously the priority in Guerrilla Analytics.
- **Substantive expertise:** This is what is often termed "domain knowledge." The more your team knows about the business domain they are working in, the better placed they are to find, understand, and add value to the data they

must work with. A geologist estimating yields from a resource knows the order of magnitude of a sensible result. A marketing analyst should know the typical take-up rates of a particular type of campaign. Without this substantive expertise, the analyst is placing blind faith in their program code and their tools.

- **"Math and statistics" knowledge:** What Drew calls math and stats means some awareness of statistics as would be used in many machine learning and statistical analyses. If you want to get some insights from data beyond a ranking and a table of counts, you will need to apply a statistical analysis or machine-learning algorithm. Typical tried and trusted methods include regression, neural networks, Bayesian networks, decision trees, clustering, and association rule mining. To use these algorithms successfully, you need to understand the conditions under which they operate, the appropriate situations in which to use them, and the techniques for evaluating and tuning their performance.

There is currently much debate over whether these diverse skill sets can be found in a single individual. This is certainly less likely when you go further and include the additional skill sets your team will require on a variety of Guerrilla Analytics projects.

- **Communication:** Communicating with customers who are not data literate is critical. It is also important that team members can work constructively with one another.
- **Visualization:** Developing minimal, insightful, and beautiful visualizations to tell a story can often be the difference between analytics being understood and used by the customer or being discarded.
- **Software engineering**: Team members need an understanding of version control, build tools, documentation, specifications, and deployment. Without these techniques, the complexity of analyzing data can quickly turn into chaos. Second, if the team is ever to effectively consolidate its knowledge and increase its effectiveness, team members should appreciate how to build tools that can be reused. A sophisticated fuzzy match algorithm's value to the team is limited if it can only ever be used by its creator.
- **The data environment:** A lot of data resides in databases and most Guerrilla Analytics is done in a DME. While database tuning and management is a specialized skill, team members do need to understand how to set up database users, permissions, logs, and table spaces as well as how to connect to a database. Web applications are ubiquitous now. They are the front-end to many data sources and they are increasingly the presentation medium for analytics reporting. Team members need a working understanding of how web applications are structured, deployed, and developed.
- **Mindset:** Team members need to be able to prioritize work to add value incrementally. They obviously need a Guerrilla Analytics mindset where they can focus on data provenance despite project disruptions.

Your approach to building a team should be to find as many of these skills as possible in an individual team member while making sure that all skills are covered across your team. Let us now consider what each of these skill set areas involves in some more detail.

16.3 PROGRAMMING

16.3.1 Data Manipulation

It is simply impossible to do Guerrilla Analytics without advanced data manipulation skills. Data comes in a wide variety of forms and shapes and these are rarely the form and shape the analyst requires for their work. Data manipulation involves being able to quickly design short data flows that reshape data and calculate necessary derived data fields and datasets.

The Guerrilla Analytics environment puts a strong emphasis on this skill because very often there is no time to build data flows with sophisticated supporting tools. Some examples of this "data gymnastics" are as follows.

- Append together a variety of datasets that not only have some common fields but also differing fields.
- Identify the first and the last record within some partition of the data. For example, identify the first and the last system log record for a given user within a given day.
- Identify fuzzy duplicates across an arbitrary subset of fields in a dataset.

There are a large number of such patterns encountered in a Guerrilla Analytics project. These are covered in detail in the appendix "Data Gymnastics."

The Guerrilla Analyst must be able to dive into data, recognize these patterns, and quickly implement them with program code. This brings us to a related point.

To manipulate data with program code your team will need a data manipulation language. There is a variety of languages to choose from. SQL is necessary given its ubiquity in the relational database world. Other DMEs come with their own domain-specific language such as R (Crawley, 2007). As with anything, each language has its strengths and weaknesses. Assess these languages based on the types of data manipulation you need to do most often. Pick a minimum set of languages and get the team skilled up in their use rather than trying to cover all data manipulation scenarios.

16.3.2 Command-Line Scripting

You will encounter large data files and large numbers of data files. These files often cannot be opened with conventional text editors. Even if you could open them, you might need to do something a bit more sophisticated than read their contents. You might like to count the number of lines, sample some part of the file, sum a column of data, or append and sort multiple files. You might have to

do this for a series of folders which all contain files of various dates and names. A command prompt and its associated scripting language are very useful in these scenarios where quick file manipulation is required.

At a minimum, your Guerrilla Analysts should be aware of command-line capabilities to do the following.

- **Iterate through files.** Iterate through a tree of folders to a given depth finding files with a given name pattern or other file property such as date stamp.
- **Append files.** Append one or more files together to produce a larger file.
- **Split files.** Conversely, split a file into a number of chunks based on line count, size, or some other rule.
- **Sort contents.** Sort a file by one of the columns in its content.
- **Find patterns.** Find a pattern within a file or count the occurrences of that pattern. For example, find all United States postcodes in a file.
- **Create samples.** Sample data extracted from between two given row numbers in a file. For example, extract rows 20,560 to row 20,600 from a file.
- **Find and replace.** Find and replace particular patterns within a file. For example, find the text "CUSTOMER: <FIRST NAME> <LAST NAME>" and replace with "CUSTOMER: XXX ZZZ" so that customer names are masked.
- **Compression.** Compress and decompress a file.
- **Character encoding.** Change the character encoding of a file.

These are some examples that you will encounter frequently. The more familiar and comfortable the team is with the command line, the better. It is a powerful, fast, and lightweight solution to data wrangling problems and is always available when more sophisticated tools may not be due to Guerrilla Analytics restrictions.

16.3.3 File Formats

There are several very common data file formats. Analysts need to understand what these formats look like and how to extract content from them. Comma-Separated Value (CSV) files are probably the most common format. XML and increasingly JSON are unavoidable. Programming languages such as Python (Lutz, 2009) provide functionality for iterating through and parsing these formats. Make sure that your team is able to work with these file formats and does not waste time writing file parsing code that already exists.

16.3.4 Data Visualization Language

Visualizing data is critical when exploring new data and when presenting data analytics to customers. Some languages have visualization libraries that allow plots of data to be produced from program code. Dashboard tools allow analysts to quickly prototype interactive data dashboards. Some combination of both

approaches is required. Code-driven visualizations have all the usual advantages of version control, repeatability, and automation. Dashboard tools suit a more interactive approach to discovering relationships in data.

16.4 SUBSTANTIVE EXPERTISE

This is a vague and wide-ranging skill. It is best understood in terms of examples. How much better is a forensic data analyst if they have accounting experience, understand accounting rules, and know the main types of fraud? How much better is a data analyst in investment banking if they have worked in a trading environment? What about a data analyst who begins a customer analytics job having worked in marketing?

Few data analysts have a large amount of hands-on experience in a domain in addition to being skillful Guerrilla Analysts – there just is not enough time to be an expert in two or more fields. However, there is undoubtedly an advantage to the analyst who gains as much experience as possible in a domain to better understand the business drivers and challenges. Substantive expertise helps focus analyses quickly and avoids time being wasted in understanding irrelevant data patterns or pursuing useless analyses.

16.5 COMMUNICATION

It may seem obvious, but communication is a key skill for a Guerrilla Analyst. It is sometimes said that everything we do is about influencing others (Block, 2011). This is true for the Guerrilla Analyst as much as anybody else in a business environment. What use is an analysis if you cannot explain it and persuade a customer to use it? But communication is also important internally within a team. A Guerrilla Analytics environment cannot support lone rangers. There is too much going on in parallel and too much is changing for a go-it-alone attitude. A Guerrilla Analyst who communicates well also knows when to inform their team of issues, gives feedback constructively, and challenges team decisions with the right tone and language. It is important that the Guerrilla Analyst be able to communicate with the following audiences.

- **The customer.** You must be able to describe your analyses and their importance in simple business terms. A customer rarely cares about the sophisticated algorithms behind results. They only care about the bottom line and how your analyses support the decision they need to make. Know when to impress with knowledge of decision trees, data contortions, and data velocity stats, and know when to tailor communication at the right level.
- **Team members.** In a healthy Guerrilla Analytics team, there will be disagreements between team members and challenges to a particular approach that you are advocating. It is important to be able to understand where these challenges are coming from and present a reasoned non-emotional

argument for your position. Teams damage themselves irreparably because of arguments over technology and process. A good communicator can influence their technical peers, superiors and reports, and get their message across.

- **Management.** The higher up people go in management, the more removed they become from the details of Guerrilla Analytics. Over time they may forget how difficult it is to estimate analytics jobs before getting stuck into the data. This worsens as technology moves on. They forget how surprises are always lurking in every dataset. This causes problems when management expectations of delivery are not in tune with reality. A Guerrilla Analyst must be able to identify and communicate issues early and present incremental approaches to a problem. This makes management easier because managers have frequent visibility of progress and a "safety net" of delivery they can rely on even if the more advanced analyses and hypotheses do not come to fruition.

How do you improve communication skills in your team? The simple answer is practice. You must incentivize them to read voraciously, write in journals and blogs, and seek out opportunities to communicate analytics to a wide variety of audiences using a variety of media. While a peer review and feedback helps, there are also many self-publication channels that anybody can use such as blogs, twitter, and video websites.

16.6 "MATHS AND STATS"

This is another incredibly broad area that is impossible to cover in one book. There are simply hundreds of available statistical tests and modeling techniques. It would be arrogant to suggest that a data analyst could become expert in what is the full-time professional domain of statisticians and machine-learning experts. However, it is necessary and possible to be familiar with how these techniques are used and the right questions to ask of an expert when choosing an appropriate technique.

A Guerrilla Analyst involved in modeling should be familiar with the following fundamental concepts:

- **Independent variables.** These are the inputs to a statistical model.
- **Dependent variables.** These are the outputs from a model – the things that are predicted.
- **Transformations.** How to change and transform variables appropriately to make them suitable for an analysis.
- **Choice of variables.** How to assess whether variables should be included in a model.
- **Performance.** The concept of false positives and false negatives, and how to measure algorithm performance.
- **Testing.** Model validation as discussed in Chapter 15.

16.7 VISUALIZATION

Visualization is hugely important in communicating analytics results. Trends, changes, proportions, comparisons, connectivity—all are best communicated or supported with visualization. The ongoing increases in the volume and complexity of data only emphasize the importance of good visualization.

Excellent books by Nathan Yau (2013) and Edward Tufte (1990), and the ACM article "A Tour Through the Visualization Zoo" (Heer et al., 2010) are inspirational introductions to visualization.

For a Guerrilla Analyst, the key is to be able to recognize what they are trying to communicate and choose the appropriate visualization technique quickly. Here is a summary of typical visualizations that the team should be familiar with.

- **Changes over time:** Here you want to show how one or more quantities change over some time period. Bar charts and line charts (perhaps with multiple series) are a good option.
- **Static proportions:** In many cases you want to show the breakdown of a population. If the population is static, a pie chart or a tree map is a good option. A bar chart is also useful if various categories are being compared.
- **Changing proportions:** When data proportions change over time, an area chart or a stacked bar chart with time on the horizontal axis are good options.
- **Correlation:** This is when you want to demonstrate relationships between variables. For example, when variable A increases then variable B is seen to decrease. The scatter plot is the classic way to demonstrate correlations.
- **Distribution:** Sometimes you want to know how a population breaks down into buckets. This is a distribution and is best represented by a histogram.
- **Comparisons:** In these cases you want to compare a variable in terms of its distribution and key statistics such as median, mean, and range. A box plot offers a concise but information-dense way of doing this.
- **Outliers:** Here you are looking at particular variable values that are significantly different from the rest of the population. Again the box plot is very useful here. If outliers as a combination of variables are of interest, then a scatter plot quickly demonstrates interesting cases.
- **Geography:** If locations are of interest then the obvious solution is a map. You can color regions, or place bar and pie charts at locations to bring out variables at that particular location.
- **Connectivity:** Connectivity often arises when you look at networks and relationships. For example, you might consider email or phone call traffic, or connections between web pages. In these cases a network diagram is best.
- **Hierarchy:** The canonical way to represent a hierarchy is with a tree diagram.
- **Change in rank:** In some cases you would like to know how the rankings of items have changed between two points in time. This is where a slope chart comes into play.

- **Sequential additions and subtractions:** In some scenarios you would like to understand the cumulative effect of sequentially introduced additions and subtractions. This is where a waterfall chart is very useful (Rasie, 1999).

16.8 SOFTWARE ENGINEERING

A Guerrilla Analytics team needs to be able to do more than code against data. They are also creating sophisticated data and service builds. This type of coding is closer to traditional programming, but there is a surprising lack of knowledge of software engineering amongst data analysts. Software engineers have been thinking hard about problems of workflow management, version control, testing, coding conventions, configuration, and many other project operational challenges that data analysts ignore or struggle with. The following sections summarize the key software engineering skills that your Guerrilla Analytics team needs to know.

16.8.1 Source Code Control

Source code version control is probably the most important and fundamental aspect of software engineering. Guerrilla Analysts are not building million line code bases with support for multiple active releases. Guerrilla Analysts are however dealing with another type of complexity – that of changing data and requirements, as well as outputs that span several languages and formats. Version control is critical across the Guerrilla Analytics workflow in the maintenance of data provenance. A Guerrilla Analyst would benefit from understanding:

- **Concept of workspaces and code repositories:** The basic set up common across all source code control, which is that of a repository and workspaces. The repository centralizes and manages all the team's code. Individual workspaces are where a team member works on their copy of the code.
- **Repository/depot:** The repository is where files and their version history are stored. It may be on a separate server or can be local to the project files.
- **Checkout:** How to take a copy of the latest code or a specific version of the code from a repository into a local workspace.
- **Commit:** How to put code changes made in the local workspace back into the repository.
- **Update/sync:** How to pull the latest code file version from the repository into a local workspace.
- **Conflict:** How to recognize conflicts when more than one user has modified a file in their respective workspaces and how to resolve those conflicts by stepping through a conflict report.
- **Trunk/mainline/baseline:** The main code development.
- **Branch/fork:** Branching or forking involves taking a copy of the main/trunk code at a point in time. That branch of code can then develop independently. Branching is useful when you want to create a release for a customer.

- **Integration:** This is the process of taking changes made in a branch and merging them back into the mainline/trunk.
- **Tag/label:** This is a snapshot across all versions of files at a point in time. A tag/label can be equivalent to a version in the sense of a software product. Tags are generally named in some descriptive way and perhaps following a convention.

There are many tools for handling source code control (Collins-Sussman et al., 2008; Loeliger and McCullough, 2012). Choose one tool and get the team trained in its use.

16.8.2 Deployment Environments

The concept of local, test, and production environments is fundamental to software engineering.

- The local environment is where the developer tests their own changes to the data and code.
- The test environment is where all local changes from several developers are brought together and tested.
- The production environment is where tested code is rolled out for the customer.

Similar principles can be usefully applied in Guerrilla Analytics, especially when consolidating knowledge in builds. Your team needs to have this basic discipline and awareness when producing code that is deployed to customers.

16.8.3 Testing

Testing is another well-established area of software engineering. Some methodologies even advise writing tests before actually writing application code. Testing Guerrilla Analytics work has its own peculiarities – so much so that several chapters of this book are devoted to testing.

16.9 MINDSET

Finally, a Guerrilla Analyst needs to have the right mindset. Fast moving, poorly understood data with changing requirements can be an intimidating and frustrating environment to work in. Database tuning, algorithms, software releases, pattern matching – it can feel like you are a jack of all trades and master of none. The most successful Guerrilla Analysts will have the following mindset.

- **Curiosity:** They must relish tearing data apart, and twisting it inside out to find insights. Many of their lines of attack will lead to nothing but the Guerrilla Analyst must enjoy that exploratory process rather than expect a specification of what they need to do.

- **Passion:** They must care about presenting results in a beautiful and insightful way so that they influence customers. Fancy algorithms and data gymnastics are irrelevant to telling a data story. The Guerrilla Analyst must want to explain their analyses and improve them.
- **Discipline:** Maintaining data provenance requires discipline. An analyst needs to keep versioning, testing, conventions, and quality code in mind despite a sometimes chaotic project environment. Hacking around datasets carelessly without regard for the data provenance implications creates technical debt.
- **Patience:** Data is like a box of chocolates – you never know what you're going to get (Zemeckis, 1994). Some might naively think it is just a matter of "add another column." The reality is that there are always surprises lurking. A Guerrilla Analyst needs to be able to deal with the unexpected setbacks and persevere with wrangling insight out of the data they have been dealt.

16.10 WRAP UP

In this chapter, you have learned about the skill sets needed in Guerrilla Analytics. In particular, you should now understand the following.

- Data analytics is a wide ranging diverse set of activities. What defines a Guerrilla Analyst is that they have to produce insight in dynamic circumstances while being quite highly constrained.
- It is unlikely that any individual would have all the required skills of a guerrilla data analyst; however, you should ensure your team overall has as many of these skills as possible.
- The main skill areas for a Guerrilla Analyst are:
  - **Data wrangling:** The ability to quickly manipulate data according to frequently encountered patterns.
  - **Programming:** The ability to automate routine tasks, write and understand algorithms, understand the many technical environments and systems they will encounter.
  - **Substantive expertise:** Knowing and understanding the problem domain from the business perspective.
  - **Communication:** The ability to communicate results and issues within the team and with the customer.
  - **Maths and stats:** Some knowledge of statistical models and machine learning, how to choose appropriate approaches and how to evaluate them.
  - **Visualization:** The ability to choose and implement the right data visualization to communicate about data.
  - **Software engineering:** Knowledge of version control, deployment, and testing as appropriate for data analytics work.
  - **Mindset:** The right combination of curiosity, patience, discipline, and passion to deal with analytics setbacks and extract insights from complex data.