Data-Centric Systems and Applications

Antonio Badia
Computer Engineering & Computer Science
University of Louisville
Louisville, KY, USA

ISSN 2197-9723 ISSN 2197-974X (electronic)
Data-Centric Systems and Applications
ISBN 978-3-030-57591-5 ISBN 978-3-030-57592-2 (eBook)
https://doi.org/10.1007/978-3-030-57592-2

This Springer imprint is published by the registered company Springer Nature Switzerland AG.
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

have given a list of known names so that readers with different backgrounds can relate what is in here with what they already know.

The goal of the book is to introduce some basic concepts to a wide variety of readers and provide them a good foundation on which they can build. After going through this book, readers should be able to profitably learn more about Data Mining, Machine Learning, and database management from more advanced textbooks and courses. It is my hope that most of them feel that they have been given a springboard from which they are in a good position to dive deeper into the fascinating world of data analysis.

Louisville, KY, USA Antonio Badia
July 2020

Contents

Chapter 1
The Data Life Cycle

It is sometimes said that "data is the new oil." This is true in several ways: in particular, data, like oil, needs to be processed before it is useful. Crude oil undergoes a complex refining procedure as the substance that comes out of wells is transformed into several products, mostly fuels (but also many other useful by-products, from asphalt to wax). A complex infrastructure, from pipelines to refineries, supports this process. In a similar way, raw data must be thoroughly treated before it can be used for anything. Unfortunately, there is not a big and sophisticated infrastructure to support data processing. There are many tools that support some of the steps in the process, but it is still up to every practitioner to learn them and combine them appropriately.

In this chapter, we introduce the stages through which data passes as it is refined, analyzed, and finally disposed of. The collection of stages is usually called the *data life cycle*, inspired by the idea that data is 'born' when it is captured or generated and goes through several stages until it reaches 'maturity' (is ready for analysis) and finally an end-of-life, at which point it is deleted or archived. Data analysis, which is the focus of Data Mining and Machine Learning books and courses, is but one step in this process. The other steps are equally important and often neglected.

The main purpose of this chapter is to introduce a framework that will help organize the contents of the rest of the book. As part of this, it introduces some basic concepts and terms that are used in the following chapters. In particular, it provides a classification of the most common types of *datasets* and *data domains* that will be useful for later work. We will come back to these topics throughout the book, so the reader is well served to start here, even though SQL itself does not appear until the next chapter. Also, for readers who are new to data analysis, this chapter provides a basic outline of the field.

A. Badia, *SQL for Data Science*, Data-Centric Systems and Applications,
https://doi.org/10.1007/978-3-030-57592-2_1

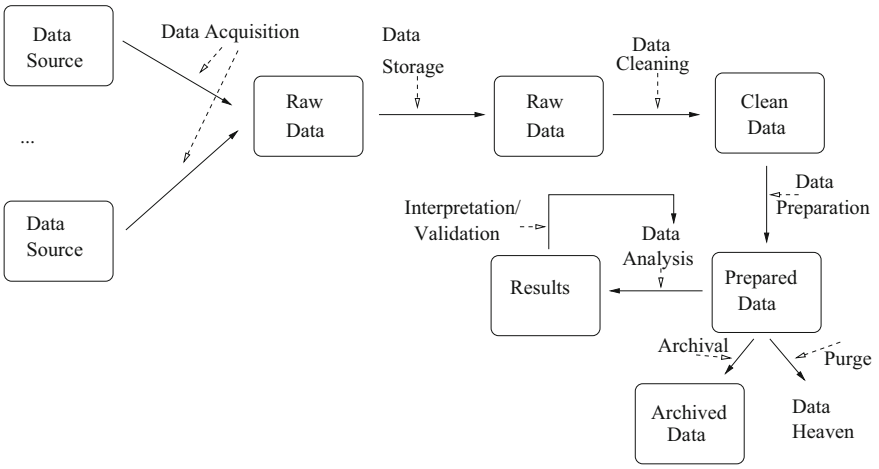

Fig. 1.1 The data life cycle

1.1 Stages and Operations in the Data Life Cycle

The term *data life cycle* refers both to the transformations applied to data and to the states that data goes through as a result of these transformations. While there is not, unfortunately, general agreement on the exact details of what is involved at each transformation and state, or how to refer to them, there is a wide consensus on the basic outlines. The states of the cycle can be summarized as follows:

$$\text{Raw data} \rightarrow \text{cleaned data} \rightarrow \text{prepared data} \rightarrow \text{data + results} \rightarrow \text{archived data}$$

The arrows here indicate precedence; that is, raw data comes first, and cleaned data is extracted from it, and so on. The activities are usually described as follows:[1]

$$\text{Data Acquisition/capture} \rightarrow \text{data storage} \rightarrow \text{data cleaning/wrangling/enrichment} \rightarrow \text{data analysis} \rightarrow \text{data archival/preservation}$$

Again, the arrows indicate precedence; data acquisition/capture happens first, followed by data storage, and so on.

The diagram in Fig. 1.1 shows the activities and stages together. We now describe each part in more detail.

The first activity in data analytics is to acquire, collect, or gather data. This happens in different ways. Sometimes existing sources of data are known and

[1] Some activities are given different names in different contexts.

accessible, sometimes a prior step that uncovers sources of relevant data[2] must be carried out. What we obtain as the result of this step is called *raw data*.

It is very important to understand that "raw" refers to the fact that this is the data before any processing has been applied to it, but does not indicate that this data is "neutral" or "unfiltered." In statistics, the domain of study is called the *population*, and the data collected about the domain is called the *sample*. It is understood that the sample is always a subset of the whole population and may vary in size from a very small part to a substantial one. However, the sample is never the population, and the fact that sometimes we have a large amount of data should not fool us into believing otherwise. For analysis of the sample to provide information about the population, the sample must be *representative* of the population. For this to happen, the sample must be chosen at random from elements of the population which are equally likely to be selected. It is very typical in data science that the data is collected in an *opportunistic* manner, i.e. data is collected because it is (easily) available. Furthermore, in science data usually comes from experiments, i.e. a setting where certain features are controlled, while a lot of data currently collected is *observational*, i.e. derived from uncontrolled settings. There are always some decisions as to what/when/how to collect data. Thus, raw data should not be considered as an absolute source of truth, but carefully analyzed.

When data comes in, we can have two different situations. Sometimes datasets come with a description of the data they contain; this description is called *metadata* (metadata is described in some detail in Sect. 1.4). Sometimes the dataset comes without any indication of what the data is about, or a very poor one. In either situation, the first step to take is *Exploratory Data Analysis (EDA)* (also called *data profiling*). In this step, we try to learn the basic characteristics of the data and whatever objects or events or observations it describes. If there is metadata, we check the dataset against it, trying to validate what we have been told—and augment it, if possible. If there is not metadata, this is the moment to start gathering it. This is a crucial step, as it will help us build our understanding of the data and guide further work. This step involves activities like classifying the dataset, getting an idea of the attributes involved, and for each attribute, getting an idea of data distribution through *visualization* techniques, or *descriptive statistics* tools, like histograms and measures of centrality or dispersion.[3]

We use the knowledge gained in EDA to determine whether data is correct and complete, at least for current purposes. Most of the time, it will not be, so once we have determined what problems the data has, we try to fix them. There are often issues that need to be dealt with: the data may contain errors or omissions, or it may not be in the right format for analysis. There are many *sources* of errors: manual (unreliable) data entry; changes in layout (for records); variations in measurement, scale, or format (for values); changes in how default or missing values are marked; or outdated values (called "gaps" in time series). Many of these issues can only be

[2]This step is referred to as *source discovery*.

[3]Readers not familiar with these notions will be introduced to the basics in Chap. 3.

addressed by changing the data gathering or acquisition phase, while others have to be fixed once data is acquired.[4] The tools and techniques used to fix these problems are usually called *data cleaning* (or *data cleansing*, *data wrangling*, *data munging*, among others). The issues faced, and the typical operations used, include

- Finding and handling *missing values*. Such values may be explicitly or implicitly denoted. Explicitly denoted missing values are usually identified with a marker like 'NULL,' 'NA' (or "N.A.," for "Not Available") or similar; but different datasets may use different conventions. Implicit missing values are denoted by the absence of a value instead of by a marker. Because of this variety, finding missing values is not always easy. Handling the absence of values can be accomplished simply by deleting incomplete data, but there are also several techniques to *impute* a missing value, using other related values in the dataset. For example, assume that we have a dataset describing people, including their weight in pounds. We realize that sometimes the weight is missing. We could look for the weight of people with similar age, height, etc. in the dataset and use such values to fill in for the missing ones.
- Finding and handling *outliers*. Outliers are data values that have characteristics that are very different from the characteristics of most other data values in a set. For example, assume that in the people dataset we also have their height in feet. This is a value that usually lies in the 4.5–6.5 range; anyone below or above is considered very short or very tall. A value of 7.5 is possible, but suspicious; it could be the result of an error in measurement or data entry. As this example shows, finding outliers (and determining when an outlier is a legitimate value or an error) may be context-dependent and extremely hard.
- Finding and handling *duplicate data*. When two pieces of the dataset refer to the same real-world item (entity, fact, event, or observation), we say the data *contains duplicates*. We usually want to get rid of duplicate data, since it could bias (or otherwise negatively influence) the analysis. Just like dealing with outliers, this is also a complex task, since it is usually very hard to come up with ways to determine when duplicate data exists. Using again the example of the people dataset, it is probably not smart to assume that two records with the same name refer to the same person; some names are very common and we could have two people that happen to share the same name. Perhaps if two records have the same name and address, that would do—although we can imagine cases where this rule does not work, like a mother and a daughter with the same name living together. Maybe name, address, and age will work? Many times, the possibility of duplication depends on the context; for instance, if our dataset comes from children in a certain school, first and last name and age will usually do to determine duplication; but if the dataset comes from a whole city, this may not be enough.

[4] The overall management of issues in data is sometimes called *Data Quality*; see Sect. 1.4.

The result of these activities is usually referred to as *clean data*, as in 'data that has been cleaned and fixed.' While cleaning the data is a necessary pre-requisite for any type of analysis, at this point the data is still not ready to be analyzed. This is because different types of analysis may require different additional treatment. Therefore, another step, usually called *data pre-processing* or *data preparation* is carried out in order to prepare the data for analysis. Typical tasks of this step include:

- Transformations to put data values in a certain format or within a certain frame of reference. This involves operations like *normalization, scaling*, or *standardization*.[5]
- Transformations that change the data value from one type to another, like *discretization* or *binarization*.
- Transformations that change the structure of the dataset, like *pivoting* or *(de)normalization*. Most data analysis tools assume that datasets are organized in a certain format, called *tabular data*; datasets not in this format need to be restructured. We describe tabular data in the next section and discuss how to restructure datasets in Sect. 3.4.

Data is now finally ready for analysis. Many techniques have been developed for this step, mostly under the rubric of Statistics, Data Mining, and Machine Learning. These techniques are explained in detail in many other books and courses; in this book we explain a selected few in detail (including an implementation in SQL) in Chap. 4.

Once data has been analyzed, the results of the analysis are usually examined to see if they confirm or disprove any hypothesis that the researcher/investigator may have in mind. The results sometimes generate further questions and produce a cycle of further (or alternative) data analysis. They can also force a rethinking of assumptions and may lead to alternative ways of pre-processing the data. This is why there is a *loop* in Fig. 1.1, indicating that this may become an *iterative* process.

Finally, once the cycle of analysis is considered complete, the results themselves are stored, and a decision must be taken about the data. The data is either *purged* (deleted) or *archived*, that is, stored in some long-term storage system in case it is useful in the future. In many cases, the data is *published* so it can be shared with other researchers. This enables others to reproduce an analysis, to make sure that the results obtained are correct. The publication also allows the data to be reused for different analyses. Whenever data is published, it is very important that it be accompanied by its metadata, so that others can understand the meaning of the dataset (what exactly it is describing) as well as its scope and limitations. If the data was cleaned and pre-processed, those activities should also be part of the metadata. In any case, data (like oil) should be disposed of carefully.

[5] Again, readers not familiar with these should wait until their introduction in the next chapter.