CHAPTER 1

The Analytical Data Life Cycle

Like all things, there is a beginning and an ending in every journey. The same can be said about your data. Thus, all data have a life cycle—from inception to end of life and the analytical data life cycle is no different. In my interactions with customers, they tend to relate to four stages (data exploration, data preparation, model development, and model deployment) as the framework for managing the analytical data life cycle. Each stage is critical, as it supports the entire life cycle linearly. For example, model development cannot happen effectively if you do not prepare and explore the data beforehand. Figure 1.1 illustrates the analytical data life cycle.

Each phase of the lifecycle requires a specific role within the organization. For example, the IT's role is to get all data in one place. Business analysts step in during the data exploration and data preparation processes. Data scientists, data modelers, and statisticians are often involved in the model development stage. Finally, business analysts and/or IT can be a part of the model deployment process. Let's examine each stage of the analytical data life cycle.

STAGE 1: DATA EXPLORATION

The first and very critical stage is data exploration. Data exploration is the process that summarizes the characteristics of the data and extracts knowledge from the data. This process is typically conducted by a business analyst who wants to explore:

- What the data look like
- What variables are in the data set
- Whether there are any missing observations

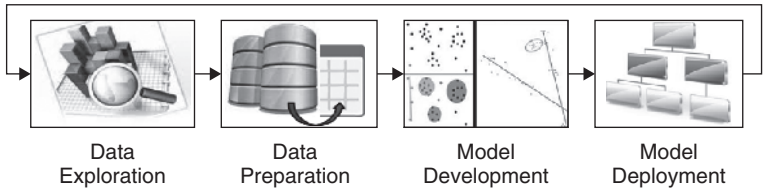

Figure 1.1 Analytical data life cycle

- How the data are related
- What are some of the data patterns
- Does the data fit with other data being explored?
- Do you have all of the data that you need for analysis?

An initial exploration of the data helps to explain these common inquiries. It also permits analysts to become more familiar and intimate with the data that they want to analyze.

The data exploration process normally involves a data visualization tool. In recent years, data visualization tools have become very popular among business analysts for data exploration purposes because they provide an eye-catching user interface that allows users to quickly and easily view most of the important features of the data. From this step, users can identify variables that are likely good candidates to explore and provide value to the other data that you are interested in for analysis. Data visualization tools offer many attractive features, and one of them is the ability to display the data graphically—for example, scatter plots or bar charts/pie charts. With the graphical displays of the data, users can determine if two or more variables correlate and whether they are relevant for further in-depth analysis.

The data exploration stage is critical. Customers who have opted to skip this stage tend to experience many issues in the later phases of the analytical life cycle. One of the best practices is to explore all your data directly in the database that allows the users to know the data before extracting for analysis, eliminate redundancy, and remove irrelevant data for analytics. The ability to quickly extract knowledge from large complex data sets provides an advantage for the data preparation stage.

STAGE 2: DATA PREPARATION

The second stage of the analytical life cycle is data preparation. Data preparation is the process of collecting, integrating, and aggregating the data into one file or data table for use in analytics. This process can be very tedious and cumbersome due some of the following challenges:

- Combining data from numerous sources
- Handling inconsistent or nonstandardized data
- Cleaning dirty data

- Integrating data that was manually entered
- Dealing with structured and semistructured data
- Value of the data

Customers that I have dealt with spend as much as 85 percent of their time preparing the data in the stage of the life cycle. The data preparation normally involves an IT specialist working closely with a business analyst to thoroughly understand their data needs. They say that preparing data generally involves fixing any errors (typically from human and/or machine input), filling in nulls and/or incomplete data, and merging/joining data from various sources or data formats. These activities consume many resources and personnel hours.

Data preparation is often directed to harmonize, enrich, and standardize your data in the database. In a common scenario, you may have multiple values that are used in a data set to represent the same value. An example of this is seen with U.S. states—where various values may be commonly used to represent the same state. A state like North Carolina could be represented by "NC," "N.C.," "N. Carolina,"' or "North Carolina," to name a few. A data preparation tool could be leveraged in this example to identify an incorrect number of distinctive values (in the case of U.S. states, a unique count greater than 50 would raise a flag, as there are only 50 states in the United States). These values would then need to be standardized to use only an acceptable or standard abbreviation or only full spelling in every row.

Data preparation creates the right data for the model development process. Without the right data, you may be developing an incomplete data model on which to make your decisions. In a worst-case scenario where you have the incorrect data for the analytic data model, you will get erroneous results that send you down the path of a devastating decision. Bringing all the data from different sources and ensuring that the data are cleansed and integrated are the core building blocks to a complete analytical data model for decision support.

STAGE 3: MODEL DEVELOPMENT

Now that you have explored and prepared the data, it is time to develop the analytical data model. Before discussing the model development cycle, it is worthwhile to provide business pains faced

by many organizations that develop a large number of analytical data models. Data models can take days, weeks, and even months to complete. The complexity is due to the availability of the data, the time it takes to generate the analytical data model, and the fact that models can be too large to maintain and in a constant state of decay.

To add to the complexity, model development involves many team members—data modelers, data architects, data scientists, business analyst, validation testers, and model scoring officers. Many organizations are challenged with the process of signing off on the development, validation, storage, and retirement of the data model. Model decay is another challenge that organizations encounter, so they need to constantly know how old the model is, who developed the model, and who is using the model for what application. The ability to version-control the model over time is another critical business need that includes event logging, tracking changes to the data attributes, and understanding how the model form and usage evolve over time. It also addresses what to do with the retired models—possibly archiving them for auditability, traceability, and regulatory compliance.

The use of an analytical data model varies from customer to customer. It is dependent on the industry or vertical that you are in; for example, you might have to adhere to regulations such as Sarbanes-Oxley or Basel II. Customers commonly leverage their analytical data models to examine

- Customer retention
- Customer attrition/churn
- Marketing response
- Consumer loyalty and offers
- Fraud detection
- Credit scoring
- Risk management
- Lifetime value
- Path to purchase
- Drug development
- Clinical trials
- Anti-money laundering

- Demand forecasting
- Loss prevention

If you are in the banking/financial industry, you may develop a data model that looks at time since last payment, number of missed payments, ratio of accrued interest, or formal loan cancellation to analyze risk of a loan default application. For retail and telecommunications, you may want to develop a data model that looks at customer acquisition/churn and cross-sell opportunities that rely on response to previous promotions within some time period and/or through some channels to enhance customer experience. Regardless of the data model type or industry, the data used in the analytical data model must be up to date and available during the lifetime of the model development and scoring processes.

Analytical data models have the ability to uncover hidden opportunities and are considered to be the fundamental success of a business. The use of analytics is increasing at an exponential rate, and organizations are developing analytical data models to enable data-driven decisions. Once models are built, deploying the models provides the outputs (results) that are driving many operational processes throughout the organizations.

STAGE 4: MODEL DEPLOYMENT

Once the model is built, it is time to deploy the model. Deploying a model often implicates scoring of the analytical data model. The process of executing a model to make predictions about behavior that has yet to happen is called *scoring*. Thus, the output of the model that is often the prediction is called a *score*. Scores can be in any form—from numbers to strings to entire data structures. The most common scores are numbers such as

- Probability of responding to a particular promotional offer
- Risk of an applicant defaulting on a loan
- Propensity to pay off a debt
- Likelihood a customer leave/churn
- Probability to buy a product

Scoring as part of the model deployment stage is the unglamorous pillar of analytical data life cycle. It is not as thrilling or exciting as the model development stage, where you may incorporate a neural network or a regression algorithm. Without the scoring and model deployment, the analytical data model is shelfware and is pretty useless. At the end of the day, however, scoring your analytical data model will reveal the information to enable you to make data-driven decisions.

The application that is used to execute the scoring process is typically simpler than the ones used to develop the models. This is because the statistical and analytical functions and optimization procedures that were used to build the model are no longer needed; all that is required is a piece of software that can evaluate mathematical functions on a set of data inputs from the analytical data model.

The scoring process invokes a software application (often called the *scoring engine*), which then takes an analytical data model and a data set to produce a set of scores for the records in the data set. There are three common approaches to scoring an analytical data model:

1. A scoring engine software application that is separate from the model-development application
2. A scoring engine that is part of the model-development application
3. A scoring engine that is produced by executing the data model code (e.g., SAS, C++, or Java) that is output by the model development application

The type of model generated will depend on the model development software that is used. Some software can produce multiple types of models, whereas others will generate only a single type. In the first two approaches, the scoring engine is a software application that needs to be run by the user. It might have a graphical user interface or it might be a command line program, in which the user specifies the input parameters by typing them onto a console interface when the program is run. There are usually three inputs to the scoring engine: the model that is to be run, the data to be scored, and the location where the output scores should be put.

In the third approach of the scoring engine, the model acts as its own scoring engine. After the model development software generates

the model, it will need to be compiled into an executable form. This step is usually done manually and often requires technical knowledge of system and programming level details. The main reason to use a compiled model is to increase performance because a compiled data model will usually run significantly faster than a model that requires a separate scoring engine.

The analysts often use a model development software application that generates model-scoring code in a particular programming language. Perhaps due to company policy or data compliance, the IT department scoring officer might convert the scoring code to another language. Code conversion introduces the potential for loss in translation, which results in costly errors. A single error in the model-scoring logic results or the data attribute selection can easily deliver an incorrect output, which can cost the company millions of dollars. Converting the scoring algorithm is usually a slow manual process producing thousands of lines of code. It is best to avoid this scenario, and customers should consider selecting the model development and deployment software application that is harmonious and compatible.

END-TO-END PROCESS

I have defined the stages and characteristics of the analytical data life cycle. Figure 1.2 shows what technologies are best suited for each stage.

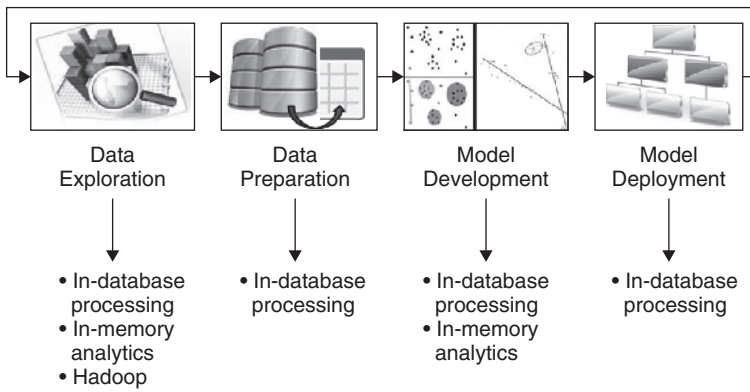

Figure 1.2 Technologies for the analytical data life cycle

In the next few chapters, I will go into details of how and why you should consider using these technologies in each of the stages. In addition, I will share anecdotes from customers who discover value in performance, economics, and governance with each technology at different stages. Each technology enables you to analyze your data faster and allows you to crawl, walk, sprint, and run through the journey of the analytical data life cycle. Your journey starts now.