available to the project, and various types of resources (e.g., people and skills, computing, time, and funding),
4) be armed with a few pearls of wisdom and lessons learned in order to help maximize the success of her or his next analytics project,
5) understand the significance of methodology to the practice of analytics within operations research and other disciplines.

5.2 Macro-Solution Methodologies for the Analytics Practitioner

As described in the Introduction, a macro-solution methodology is comprised of general steps for an analytics project, while a micro-methodology is specific to a particular type of technical solution. In this section, we describe macro-methodology options available to the analytics practictioner.

Since a macro-methodology provides a high-level project path and structure, that is, steps and a potential sequence for practitioners to follow, practitioners can use it as an aid to project planning and activity estimation. Within the steps of a macro-methodology, specific micro-methodologies may be identified and planned, aiding practitioners in the identification of specific technical skills and even named resources that they will need in order to solve the problem.

Four general macro-methodology categories are covered in this section:

A. The scientific research methodology
B. The operations research project methodology
C. The cross-industry standard process for data mining (CRISP-DM) methodology
D. The software engineering methodology

We reiterate here that there is some overlap in these methodologies and that the most important message for the practitioner is to follow a macro-solution methodology. In fact, even a hybrid will do.

5.2.1 The Scientific Research Methodology

The scientific research methodology, also known as the *scientific method* [5], has very early roots in science and inquiry. While formally credited to Francis Bacon, its inspiration likely dates back to the time of the ancient Greeks and the famous scholar and philosopher Aristotle [6].

This methodology has served humankind well over the years, in one form or another, and has been particularly embraced by the scientific disciplines where theories often are born from interesting initial observations. In the early days, and even until more recently (i.e., within the last 20 years–merely a blip in historical time!), a plethora of digital data was not available for researchers to

study; data were a scarce resource and were expensive to obtain. Most data were planned, that is, collected from human observation, and then treated as a limited, valuable resource. Because of its value both to researchers' eventual conclusions and to the generalizations that they are able to make based upon their findings, the scientific methodology related to data collection has evolved into a specialty in and of itself within applied statistics: experimental design. In fact, many modern-day graduate education programs in the United States require that students take a course related to research methodology either as a prerequisite for graduate admission or as part of their graduate coursework so that graduate students learn well-established systematic steps for research, sometimes specifically for setting up experiments and handling data, to support their MS or PHD thesis. Often, this type of requirement is not uncommon in social sciences, education, engineering, mathematics, computer science, and so on–that is, these requirements are not limited strictly to the sciences.

The general steps of the scientific method, with annotations to show their alignment with a typical analytics project, are the following:

A.1. *Form the Research Question(s).* This step is the one that usually kicks off a project involving the scientific method. However, as already noted, these types of projects may be inspired by some interesting initial observation. In applying this step to an analytics project in practice, the research questions may also relate to an underlying problem statement, which typically forms the preface for the project.

A.2. *State One or More Hypotheses.* In its most specific form, this step may involve stating the actual statistic that will be estimated and tested: for example, $H_0 : \mu_1 = \mu_2$, that is, that two treatment means are the same. (Note that a *treatment mean* is the average of observations from an experiment with a set of common inputs, that is, fixed independent variable values are the *treatment*.) Interpreted more broadly, the hypotheses to test imply the specific techniques that will be applied. For example, the hypothesis that two means are identical implies that some specific techniques of experimental design, data collection, statistical estimation, and hypothesis testing will be applied. However, one might also consider more general project hypotheses: for example, we suspect that cost, quality of service, and peer pressure are the most significant reasons that cell phone customers change their service providers frequently. These types of hypotheses imply specific techniques in churn modeling.

A.3. *Examine and Refine the Research Question and Hypotheses.* In this step, the investigating team tries to tune up the output of the first two steps of the scientific method. Historically, this is done to make sure that the planning going forward is done in the most efficient and credible way, so that ultimately, the costly manual data collection leads to usable data and scientifically sound conclusions–otherwise, the entire research project

becomes suspect and a waste of time (not to mention, the discrediting of any conclusions or general theory that the team is trying to prove). This step is not much different in today's data-rich world: Practitioners should still want to make sure they are asking the right questions, that is, setting up the hypotheses to test so that the results they hope to get will not be challenged, while trying to ensure that this is all done as cost-effectively and in as timely a manner as possible. In today's world, because of the abundance of digital data, this sometimes means exploration on small or representative data sets. This can lead to the identification of additional data needed (including derivatives of the available data), as well as adjustments to the questions and hypotheses based on improved understanding of the underlying problem and the addition of preliminary insights. Notice the carry forward of "problem understanding" that happens naturally in this step. In fact, it is good to consider the acceptable conclusion of this step as one where the underlying problem being addressed can be well enough articulated that stakeholders, sponsors, and project personnel all agree. Some preliminary model building, to support the "examination" aspect of this step, may occur here.

A.4. *Investigate, Collect Data, and Test the Hypotheses.* In traditional science and application of the scientific method, this meant the actual steps of performing experiments, collecting and recording observations, and actually performing the tests (which were usually statistically based). Applied to analytics projects, this macro-methodology step means preparing the final data, modeling, and observing the results of the model.

A.5. *Perform Analysis and Conclude the General Result.* In this step, we perform the final analysis. In traditional science, does the analysis support the hypotheses? Can we draw general conclusions such as the statement of a theory? In analytics projects, this is the actual application of the techniques to the data and the drawing of general conclusions.

As is evident here, the scientific method is a naturally iterative process designed to be adaptive and to support systematic progress that gets more and more specific as new knowledge is learned. When followed and documented, it allows others to replicate a study in an attempt to validate (or refute) its results. Note that reproducibility is a critical issue in scientific discovery and is emerging as an important concern with respect to data-dependent methods in analytics (see Refs [7,8]).

Peer review in research publication often assumes that some derivative of the scientific method has been followed. In fact, some research journals mandate that submitted papers follow a specific outline that coincides closely with the scientific method steps. For example, see Ref. [9], which recommends the following outline: Introduction, Methods, Results, and Discussion (IMRAD). While the scientific method and IMRAD for reporting may not eliminate the problem of false

discovery (see, for example, Refs [10,11]), they can increase the chances of a study being replicated, which in turn seems to reduce the probability of false findings as argued by Ioannidis [12].

Because of this relationship to scientific publishing, and to research in general, the scientific method is recommended for analytics professionals who plan eventually to present the findings of their work at a professional conference or who might like the option of eventually publishing in a peer-reviewed journal. This methodology is also recommended for analytics projects that are embedded within research, particularly those where masters and doctoral theses are required, or in any research project where a significant amount of exploration (on data) is expected and a new theory is anticipated. In summary, the scientific method is a solid choice for research-and-discovery-leaning analytics projects as well as any engagement that is data exploratory in nature.

5.2.2 The Operations Research Project Methodology

Throughout this chapter, analytics solution methodology is taken to mean the approach used to solve a problem that involves the use of data. It is worth bringing this point up in this section again because, as mentioned in the Introduction, our perspective assumes an INFORMS audience. Thus, we are biased toward these methodology descriptions for analytics projects that will be applying some operations research/management science techniques. While it was natural to start this macro-section with the oldest, most established, mother of all exploratory methodologies (the scientific method of the last section), it is natural to turn our attention next to the macro-method established in the OR/MS practitioner community.

In general, one may find some variant of this project structure in introductory chapters of just about any OR/MS textbook, such as Ref. [13], which is in its fourth edition, or Ref. [14], which was in its seventh edition in 2002. (There have been later editions, which Dr. Hillier published alone and with other authors after the passing of Dr. Lieberman.)

Most generally, the OR project methodology steps include some form of the following progression:

B.1. *Define the Problem and Collect Data.* As most seasoned analytics and OR practitioners know, problem statements are generally not crisply articulated in the way we have been used to seeing them in school math classes. In fact, as noted earlier, sponsors and stakeholders may have disparate and sometimes conflicting views on what the problem really *is*. Sometimes, some exploratory study of existing data, observing the real-world system (if it exists), and interviewing actors and users of the system helps researchers to gain the system and data understanding needed for them to clarify what the problem is that should be solved by the project. The work involved in

this step should not be underestimated, as it can be crucial to later steps in the validation and in the acceptance/adoption/implementation of the project's results. It is a good idea to document assumptions, system and data understanding, exploratory analyses, and even conversations with actors, sponsors, and other stakeholders. Finding consensus about a written problem statement, or a collection of statements, can be critical to the success of the project and the study, so it is worth it to spend time on this, review it, and attempt to build broad consensus for a documented problem statement.

Collecting data is a key part of early OR project methodology, and is intriciately coupled with the problem definitition step, as noted in Ref. [14]. In modern analytics projects, data collection generally means identifying and unifying digital data sources, such as transactional (event) data (e.g., from an SAP system), entity attribute data, process description data, and so on. Moving data from the system of record and transforming it into direct insights or reforming it for model input parameters are important steps that may be overlooked or under-estimated in terms of effort needed.

As noted earlier, we live in a world where "solutions" are sexy and "problems" are not–further adding to the challenge and importance of this step. In comparison with the scientific method of the previous section, this step intersects most closely with the activities and purposes of **A.1**, **A.2**, and **A.3**.

B.2. *Build a Model.* There are many options for this step, depending on the type of problem being solved and on the objective behind solving it. For example, if we are seeking improved understanding, the model may be descriptive in nature, and the techniques may be those of statistical inference. If we are trying to support a complex decision, such as where to build a new firehouse and how to staff it, then we may build descriptive models to analyze current urban demand patterns; we may build predictive models that take those outputs to project future demand; and then we may build an optimization model to locate the facility so that future demand is best served. Much of this step is based on available data, as well as on available tools and skills, which sometimes means we choose to build the models that we are most familiar with or that we have the skills to support. This step most intersects with the activities and purposes of **A.3**, although it is not an exact mapping.

B.3. *Find and Develop a Solution.* In OR, this traditionally has meant the work of solving the equations or doing the math that finds the solution, designing the algorithm, and coming up with a computer code to implement the algorithm. There are many variants of this step today because the models may be derived fully from data or logic, and the micro-methods for finding the solutions can be specific to the technique. However, the common denominator here has to do with the algorithm, or in some cases, the heuristic: It is the recipe for taking the data, assumptions, and so on. and converting it to a useable result, however that is done. Computer code just

helps us to do that most efficiently. This step intersects most closely with the activities and purposes of **A.4**, although it is only partial in mapping. As we shall see in a later section, this step interlocks with micro-solution methodologies that can constitute the details of this macro-step.

B.4. *Test (Verify) and Validate.* This step is actually a whole bunch of activities. Testing and verifying are often used interchangeably in software development, and since we often program (i.e., "implement") our model solution (algorithm, heuristic, process, model, etc.), the interchange works here in the OR project methodology. The act of testing, or verifying, is making sure that whatever it is you made and are calling the model or solution is actually doing what you think it is doing. This is different from validation, which is making sure a model is representative of whatever you are trying to mimic, for example, a real-world system or process and a decision-making scenario. Validation asks the following question: Does the model behave as if it were the real system? There are entire areas of research devoted to these topics, not just in the analytics and OR fields, but in statistics and software engineering as well. They all are better because of the cross learning that has happened. For example, statistical methods can be used to generate and verify test cases. In validation, statistical methods are often used in rigorous simulation studies–which are basically statistical experiments done with a computer program, and as such lend themselves very nicely to things such as pairwise comparison with historical observations from the true system. Dr. Robert Sargent is one of the pioneers in computer simulation, output analysis and verification, and validation methodologies–the canonical methods he described in his 2007 paper [15] provide valuable lessons not only for simulation modelers, but also for those doing testing, verification, and validation in other types of analytics and OR projects.

B.5. *Disseminate, Use, or Deploy.* Once the solution is ready to be used, it is rolled out (disseminated, deployed), and the work is still not done! Usually, at this stage, there needs to be training, advocacy, sometimes adjustment, and virtually always maintenance (fixing things that are wrong, or adding new features as the users and stakeholders hopefully become enthralled with the work and have new ideas for it). At this stage, it is usually useful to have baked in some monitoring–that is, if you can think ahead to put in metrics that automatically observe value that is being derived from using the solution, that's awesome foresight. In too many analytics and OR projects, deployment and dissemination merely means a final presentation and report. In some cases, those recommendations are good enough! In others, they might signal that the true solution is not really intended to be "used." Sometimes, this leads to an iterative process of refinement and redeployment, allowing practitioners to restart this entire step process. In other cases, you write the report, and perhaps an experience paper gets submitted to a peer-reviewed journal or is presented at an INFORMS

conference. Whatever the outcome, practitioners need to keep in mind that all projects are worthy learning experiences–even the ones that are not deployed in the manner in which we were hoping.

It is not surprising that the OR project method, being exploratory in nature, is somewhat of a derivative of the scientific method. As Hillier and Lieberman point out in the introductory material of Ref. [16], operations research has a fairly broad definition, but in fact gets its name from research on operations. The *study objects* of the research are "operations," or sometimes "systems." These operations and systems are often digital in their planning and execution, and so tons of data now exist to model, recreate them, and model/experiment with them. In other words, these observable digital histories mean they are rich in data (analytics) that can be used to model very quickly. Unfortunately, the ability to jump right into modeling, analysis, and conclusions often means skipping over early methodological steps, particularly in the area of problem definition.

5.2.3 The Cross-Industry Standard Process for Data Mining (CRISP-DM) Methodology

"The cross-industry standard process for data mining methodology," [17,18] known as CRISP or CRISP-DM, is credited to Colin Shear, who is considered to be a pioneeer in data mining and business analytics [19]. This methodology heavily influences the current practical use of SPSS (Statistical Package for the Social Sciences), a software package with its roots in the late 1960s that was acquired by IBM in 2009 and that is currently sold as IBM's main analytics "solution" [18].

As an aside, note that SAS and SPSS are commercial packages that were born in about the same era and that were designed to do roughly the same sort of thing–the computation of statistics. SAS evolved as the choice vehicle of the science and technical world, while SPSS got its start among social scientists. Both have evolved into the data-mining and analytics commercial packages that they are today, heavily influencing the field. As mentioned earlier, the "descriptive–predictive–prescriptive" paradigm appears to have its roots in SAS. As noted above, CRISP is heavily peddled as the methodology of choice for SPSS. However, we note that this methodology is a viable one for data-mining methods that use any package, including R and SAS.

The steps of the CRISP-DM macro-methodology, from Ref. [17], are the following:

C.1. *Business Understanding.* This step is, essentially, the *domain understanding plus problem definition* step. In the business analytics context, CRISP calls out specific activities in this step, such as stating background, defining the business objectives, defining data-mining goals, and defining the success criteria. Within this step, traditional project planning (cost/benefits, risk assessment, and project plan) are included. This step also involves assessment of tools and techniques. Note that this step aligns with **B.1** of the OR project methodology.

C.2. *Data Understanding.* This is a step used to judge what data is available, by specifically identifying and describing it (for example, with a data dictionary) and assessing its quality or utility for the project goals. In most cases, actual data is collected and explored/tested.

C.3. *Data Preparation.* This is the step where analysts decide which data to use and why. This step also includes "data cleansing" (roughly, the act of finding and fixing or removing strange or inaccurate data, and in some cases, adding, enhancing, or modifying data to fix incomplete forms), reformatting data, and creating derivative data (i.e., extracting implied or derived attributes from existing data, merging data, etc.). An example of reformatting data would be converting GIS latitude and longitude (i.e., latitude/longitude) data from degree/minute/second format, for example, 41° 13′ 1″ N, 73° 48′ 27″ W, to decimal degrees, that is, 41.217, − 73.808.An example of enhancing in data cleansing is finding and adding a postal code field to a street, city, state address or geocoding the address (i.e., finding the corresponding latitude/longitude). Data merging is a common activity in this step, and it generally is used to create extended views of data by adding attributes, via match up by some key. Note that a common "mistake" among inexperienced data scientists is to try to merge extremely large *unsorted* data sets. Packages such as SPSS, SAS, and R, and even scripting languages such as Python, allow for these common types of data movement, but without presorting lists, execution to accomplish merge operations can end up taking days instead of a few minutes when the list sizes are in the millions, which is not an unrealistic volume of data to be working with these days.

C.4. *Modeling.* This is the step where models are built and applied. In data mining and knowledge discovery, the models are generally built from the data (e.g., a regression model with a single independent variable is basically a model of a linear relationship where the data is used to derive the slope and y-intercept). Other modeling-related steps include articulating the assumptions, assessing the model, and fitting parameters. Note that this step, together with the previous two steps, aligns with *B.2* and *B.3* of the OR project methdology.

C.5. *Evaluation.* This step is equivalent to the OR project verification and validation step. See *B.4*. Note that Dwork et al. [20] give a well-recognized example of a validation method for data dependent methods.

C.6. *Deployment.* This step is equivalent to the OR project deployment step. See *B.5*.

The CRISP-DM macro-methodology is thought of as an iterative process. In fact, the scientific method and the OR project method can also be embedded in an iterative process. More details of the CRISP-DM macro-methodology can be found in Chapter 7.

5.2.4 Software Engineering-Related Solution Methodologies

Software engineering is relevant to analytics macro-solution methodology because of the frequent expectation of an outcome implemented in a software tool or system. The steps of the most standard software engineering methodology, the waterfall method, are the following:

D.1. *Requirements.* This step is a combination of understanding the business or technical environment in which a system will be used and identifying the behavior (function) and various other attributes (performance, security, usability, etc.) that are needed for a solution. Advisable prerequisites for identifying high quality requirement specifications are problem, business, and data understanding. Thus, this step aligns with *B.1*, *C.1*, and *C.2*.

D.2. *Design.* The design step in software engineering translates the requirements (usually documented in a "specification") into a technical plan that covers, at a higher level, the software components and how they fit together, and at a lower level, how the components are structured. This generally includes plans for databases, queries, data movement, algorithms, modules or objects to be coded, and so on.

D.3. *Implementation.* Implementation refers to the translation of the design into code that can be executed on a computer.

D.4. *Verification.* Similar to previous macro-methodologies, verification means testing. In software, this can be unit testing, system testing, performance testing, reliability testing, and so on. The step is similar to other macro-method verification steps in that it is intended to make sure that the code works as intended.

D.5. *Maintenance.* This is the phase, in software development, that assumes the programs have been deployed and when sometimes either bug fixes will need to be done or else new functions may be added.

A number of other software engineering methodologies exist. See, for example, Ref. [21] for descriptions of rapid application development (comprised of data modeling, process modeling, application generation, testing, and turnover), the incremental model (analysis, design, code, test, etc.; analysis, design, code, test, etc.; analysis, design, code, test, etc.), and the spiral model (customer communication, planning, risk analysis, engineering, construction and release, evaluation). When looking more deeply at these steps, one can see that they can also be mapped to the other macro-methodologies–note that Agile, a popular newer form of software development, is very much like the Incremental model in that it focuses on fast progress with iterative steps.

5.2.5 Summary of Macro-Methodologies

Figure 5.2 shows how the four macro-solution methodologies are comparatively related. It is not difficult to imagine any of these macro-methodologies

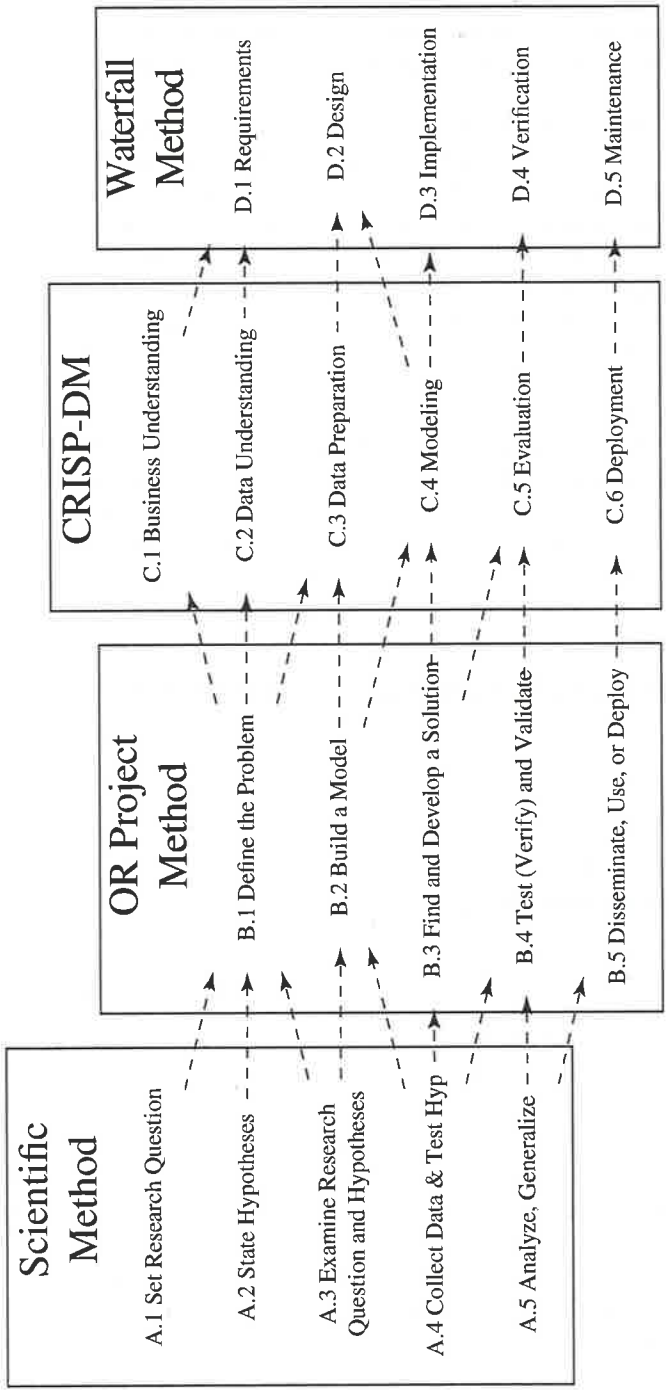

Figure 5.2 Relationship among the macro-methodologies.

embedded in an iterative process. One can also see, through their relationships, how it can be argued that each one, in some way, is derivative of the scientific method.

Every analytics project is unique and can benefit from following a macro-methodology. In fact, a macro-methodology can literally save a troubled project, can help to ensure credibility and repeatability, can provide a structure to an eventual experience paper or documentation, and so on. In fact, veteran practitioners may use a combination of steps from different macro-methodologies without being fully conscious of doing so. (All fine and good, but, in fact, you veterans could contribute to our field significantly if you documented your projects in the form of papers submitted for INFORMS publication consideration and if, in those papers, you described the methodology that you used.)

The take-home message about macro-methodologies is that it is not necessarily important exactly which one of them you use–its just important that you use one (or a hybrid) of them. It is recommended that, for all analytics projects, the steps of *problem definition* and *verification and validation* be inserted and strictly followed, whether the specific macro-methodology used calls them out directly or not.

5.3 Micro-Solution Methodologies for the Analytics Practitioner

In this section, we turn our attention to micro-methodology options available to the analytics practitioner.

5.3.1 Micro-Solution Methodology Preliminaries

In general, for any micro-methodology, two factors are most significant in how one proceeds to "solutioning":

i) The specific modeling approach
ii) The manner in which the data (analytics) are leveraged with respect to model building as well as analysis prior to modeling and using the model

Modeling approaches vary widely, even within the discipline of operations research. For example, data, numerical, mathematical, and logical models are distinguished by their form; stochastic and deterministic models are distinguished by whether they consider random variables or not; linear and nonlinear models are differentiated by assumptions related to the relationship between variables and the mathematical equations that use them, and so on. We note that

micro-solution methodology depends on the chosen modeling approach, which in turn depends on domain understanding and problem definition–that is, some of those macro-methodology steps covered in the previous section. Skipping over those foundational steps becomes easier to justify when the methods that are most closely affiliated with them (e.g., descriptive statistics and statistical inference) are side-lined in a rush to use "advanced (prescriptive) analytics."

Thus, we begin this micro-solution methodology section by re-stating the importance of following a macro-solution methodology, and by emphasizing that the selection of appropriate micro-solution methodologies–which could even constitute a collection of techniques–is best accomplished when practitioners integrate their selection considerations into a systematic framework that enforces some degree of precision in *problem definition* and *domain understanding*, that is, macro-method steps in the spirit of *A.1*, *A.2*, *A.3*, *B.1*, *B.2*, *C.1*, *C.2*, *C.3*, and *D.1* (see Figure 5.2).

All of this is not to diminish the importance of the form and purpose of the project analytics, that is, the data, in selection of micro-solution methodologies to be used. In fact,

- how data are created, collected, or acquired,
- how data are mined, transformed, and analyzed,
- how data are used to build and parameterize models, and
- whether general "solutions" to models are dependent or independent of the data

are all consequential in micro-solution methodology. However, it is the *model* that is our representation of the real world for purposes of analysis or decision-making, and as such it gives the *context* for the underlying problem and the understanding of the domain in which "solving the problem" is relevant. This is why consideration of (i) the specific modeling approach should always take precedence over (ii) the manner of leveraging the data. Thus, this section is organized around modeling approaches first, while taking their relationship to analytics into account as a close second.

5.3.2 Micro-Solution Methodology Description Framework

This section presents the micro-solution methodologies in these three general groups:

Group I. Micro-solution methodologies for exploration and discovery
Group II. Micro-solution methodologies using models where techniques to find solutions are independent of data
Group III. Micro-solution methodologies using models where techniques to find solutions are dependent on data