Data Mining: Process

Monte F. Hancock, Jr.
Chief Scientist, Celestech, Inc., Melbourne, Florida, U.S.A.

Abstract

The purpose of this entry is to provide the reader with a deeper understanding of the fundamental principles of data mining. It presents an overview of data mining as a process of discovery and exploitation that is conducted in spirals, each consisting of multiple steps. A Rapid Application Development (RAD) data mining methodology is presented that accommodates disruptive discovery and changing requirements.

GOALS

After you have read this entry, you will be able to explain the more complex principles of data mining as a discipline. You will be familiar with the major components of the data mining process, and will know how these are implemented in a spiral methodology. Most importantly, you will understand the relative strengths and weaknesses of conventional and RAD development methodologies as they relate to data mining projects.

INTRODUCTION

Successful data mining requires the cultivation of an appropriate mindset. There are many ways that data mining efforts can go astray; even seemingly small oversights can cause significant delays or even project failure. Just as pilots must maintain situational awareness for safe performance, data miners must remember where they are in their analysis, and where they are going. All of this demands a principled approach implemented as a disciplined process.

The alternative to using a disciplined process is often expensive failure. “Data mining boys love their analytic toys”; directionless analysts can spend infinite time unsystematically pounding on data sets using powerful data mining tools. Someone who understands the data mining process must establish a plan: there needs to be a “Moses.”

There also needs to be a “Promised Land.” Someone familiar with the needs of the enterprise must establish general goals for the data mining activity. Because data mining is a dynamic, iterative discovery process, establishing goals and formulating a good plan can be difficult. Having a data mining expert review the problem, set up a reasonable sequence of experiments, and establish time budgets for each step of analysis will minimize profitless wandering through some high-dimensional wilderness.

There is still some disagreement among practitioners about the scope of the term data mining: Does data mining include building classifiers and other kinds of models, or only pattern discovery? How does conventional statistics fit in? There is also disagreement about the proper context for data mining: Is a data warehouse necessary? Is it essential to have an integrated set of tools? However, there is general agreement among practitioners that data mining is a process that begins with data in some form and ends with knowledge in some form.

As we have seen, data mining is a scientific activity requiring systematic thinking, careful planning, and informed discipline. We now lay out the steps of a principled data mining process at a high level, being careful not to get lost in the particulars of specific techniques or tools.

In computer science, development methodologies that repeat a standardized sequence of steps to incrementally produce successively more mature prototypes of a solution are referred to as spiral methodologies. Each cycle through the sequence of steps is one spiral.

An enterprise is any entity that is a data owner (DO) having an operational process. This includes businesses, government entities, and the World Wide Web. The following is an overview of a data mining project as a process of directed discovery and exploitation that occurs within an enterprise.

DISCOVERY AND EXPLOITATION

As a process, data mining has two components: discovery and exploitation.
Discovery is an analytic process, for example, determining the few factors that most influence customer churn. Exploitation is a modeling process, for example, building a classifier that identifies the customers most likely to churn based upon their orders last quarter. We can characterize these functionally by noting that, during discovery, meaningful patterns are detected in data, and characterized formally, resulting in descriptive models. During exploitation, detected patterns are used to build useful models (e.g., classifiers).

- Discovery
  - Detect actionable patterns in data
  - Characterize actionable patterns in data
- Exploitation
  - Create models
  - Interact with the enterprise

Depending on the complexity of the domain, the discovery process will have some or all of the components shown in Fig. 1.

Once the discovery process has provided the necessary insight into how the data represent the domain, exploitation begins. It will have some or all of the major components in Fig. 2, and might use some of the techniques suggested there.

This inclusive view of data mining is a bit broader than that presently held by some, who reserve the term data mining for what is here called the discovery component. They would refer to our exploitation component as predictive modeling. The broader view is taken here for three reasons:

1. The broader conception of data mining appears to be the direction things are headed in business intelligence circles, driven in part by tool vendors who continually increase the scope of their integrated data mining environments.
2. Some of the same tools and techniques are used for both discovery and exploitation, making discrimination between them somewhat subjective anyway (Are we exploiting yet?).
3. More and more, analysts want to engage in both discovery and exploitation using data mining tools and methods, going back and forth between the two during a project. Distinctions between discovery and exploitation are blurred in such situations.

Though an inclusive view of data mining is taken here, it should not be inferred that the distinction between discovery and exploitation is unimportant. For the purpose of managing a data mining project, selecting the right techniques, and keeping track of what we’re doing now is essential to proper project management.

The data mining process will now be described as a sequence of steps, each having a specified purpose. The purpose, order, and content of each step are expressed in terms general enough to encompass those outlined in the well-known competing process standards.

Data mining projects are undertaken to solve enterprise problems. Some of these problems can be considered solved when insight is gained (e.g., What are the indicators of impending default?); others are solved only when this insight is made actionable by some application (e.g., Automatically predict default!). It is the difference between a question mark and an exclamation point: the descriptive models developed during the

**Data selection (defining the sampling frame)**
- Common data problems (outliers, gaps, time, consistency, collisions, imbalance)
- Identifying/handling outliers (statistical methods)
- Handling missing fields (weak fill values, degapping, gap masking)
- Handling temporal problems (aging, cycling, trends, non-stationarity, etc.)
- Checking consistency (semantic filtering)
- Handling class collisions
- Handling class imbalance (replication, decimation)
- Other

**Data cleansing**
- Data syntax (precision, scale, format)
- Nominal vs. numeric trades
- Coding (nominal to numeric data)
- Quantization (numeric to nominal data)

**Feature extraction and transformation**
- Data registration
- Data normalization
- Feature synthesis

**Feature enhancement**
- Feature salience (discriminating power of a feature)
- Feature independence (information contribution of a feature)
- Information-theoretic transforms (making information more accessible)

**Data division**
- Sampling
- Stratifying
- Segmenting (creating training, calibration, validation, and holdback sets)

**Configuration management**
- Nomenclature and naming conventions
- Documenting the workflow for reproducibility

**Components of the Discovery Process**

![Fig. 1 Components of the discovery phase.](image-url)
discovery phase address the “?” and the predictive models produced during the exploitation phase bring about the “!”.

It is the enterprise goal that determines whether both discovery and exploitation are pursued for a particular data mining project. Typically, analysts and researchers want to discover, while managers and practitioners want to exploit.

Discovery is a prerequisite to exploitation. Sometimes though, there is sufficient knowledge of the domain to begin exploitation without undertaking an extensive discovery effort. This is the approach taken, for example, by expert system developers who build intelligent applications using the knowledge already possessed by domain experts (DEs). Either way, every well-designed data mining effort includes an inquiry into what is already known about the domain.

**ELEVEN KEY PRINCIPLES OF INFORMATION-DRIVEN DATA MINING**

Included here to round out this introduction to data mining as a process are some foundational principles of the data mining process. Overlooking any one of them can lead to costly data mining project failure; ignore them at your peril. These are so important that we list them together for ready review before moving on to detailed treatments:

1. In order of importance: choose the right people, methods, and tools
2. Make no prior assumptions about the problem (begin as a domain agnostic)
3. Begin with general techniques that let the data determine the direction of the analysis (funnel method)
4. Don’t jump to conclusions; perform process audits as needed
5. Don’t be a one widget wonder; integrate multiple paradigms so the strengths of one compensate for the weaknesses of the other
6. Break the problem into the right pieces; divide and conquer
7. Work the data, not the tools, but automate when possible
8. Be systematic, consistent, and thorough; don’t lose the forest for the trees.
9. Document the work so it is reproducible; create scripts when possible
10. Collaborate with team members, experts, and users to avoid surprises
11. Focus on the goal: maximum value to the user within cost and schedule

**KEY PRINCIPLES EXPANDED**

These principles are the fruit of some of the author’s own painful and expensive lessons learned the hard way. Working with these principles in mind will reduce the likelihood that you will make a costly and avoidable data mining error. Learn them; live them.

**Key Principle Number One**

Choose the right people, methods, and tools for your data mining effort

We place this principle first because it is the most important to the success of a data mining effort. Further, the three choices listed are in order of importance. If you have to compromise on something, it should not be the skill sets of the people involved in the mining effort.
It is the aggregate skill set possessed by the team as a whole that will generate your success.

Once the team has been put together, choosing the proper mining methods is the next most important. Even people who really know what they’re doing are less likely to be successful if the methods that they employ are not appropriate to the problem. By methods, we mean the mining goals, data sources, general methodology, and work plans.

Assuming that the right people and the right methods have been selected, the next most important item is the choice of appropriate tools. Fortunately, people well-versed in the procedures for conducting a data mining effort can often be successful even when they don’t have the best toolsets. This is because many data mining activities rely on the analytic skills of the miner more than they rely on the efficiency provided by good tools.

**Key Principle Number Two**

Make no prior assumptions about the problem

This involves taking an agnostic approach to the problem. Assumptions are actually restrictions on what constitutes a reasonable hypothesis. Since you will not be a DE, you shouldn’t have any of these. Often things that haven’t been discovered about the problem space are unknown precisely because prior assumptions prevented people from perceiving them. If you make certain assumptions about what are viable solutions to a problem, you insure that certain approaches will not be carefully explored. If one of these excluded solutions is actually the solution for which you are mining, you will not discover it. Taking an agnostic approach to a data mining activity will initially leave the door open to undiscovered facts about the domain. If you don’t have a solution at hand, it makes sense to at least consider approaches that might initially appear unpromising.

One of the implications of principle number two is that DEs are often not the best people to carry out a data mining effort. The reason for this is that they already have a large collection of assumptions, many held unconsciously, about what’s going to work and not work in the problem space. These assumptions will prevent them from investigating avenues of analysis that they “know won’t work.” A data mining effort led by a DE can be expected to discover all the things they already know about the problem space, and not much else. Someone not an expert in the problem will investigate areas an expert would let pass unexamined. It is generally much better to have someone who knows more about data mining than the problem space to be the technical lead on a data mining activity. The DE certainly is a necessary part of the team, but not as the leader.

**Key Principle Number Three**

Begin with general techniques that let the data determine the direction of the analysis

For problem areas that are not well understood, begin the project by pursuing several lines of general investigation. For example, initially employ two or more different analytic methods, toolsets, etc. Rather than putting all the eggs into one analytic basket, this holds open multiple lines of investigation long enough to determine which is likely to produce the best result. With this known, subsequent effort can be focused on that one best choice.

I refer to this as the *funnel method*, because work begins with a broad analytic attack, progressively narrowing as the problem is more fully understood. This method also provides a good way to make use of multiple team members, each using a different method and working in parallel until problem understanding is achieved. When that occurs, all of your working capital can shift over to the method found most likely to bear fruit in your problem domain.

**Key Principle Number Four**

Don’t jump to conclusions; perform process audits as needed

This refers to a phenomenon that occurs often in data mining efforts, and is usually (though not always!) manifested by an inexperienced data miner. It is not unusual for an amazing analytic miracle to occur, a silver bullet that solves the problem. It is often the case, however, that huge breakthroughs on really hard problems are the deceptive fruit of some analytic mistake: some data have been mislabeled, ground truth has accidentally been added to the data set, or some data conditioning error caused an experiment to produce erroneous results.

The damage is usually limited to some embarrassment on the part of the claimant, but the real danger is that word of a breakthrough will find its way to a decision maker who will take it at face value. This can derail a data mining effort and at the very least, you’ll have some explaining to do.

When results from a significant experiment differ radically from expectations, the proper approach is to perform a *process audit*: a thorough examination of the experiment to determine the facts. The most common errors are related to data preparation and use as follows:

1. Inappropriate data have been added to a training set (e.g., ground truth).
2. Validation has been done on the same data that was used to create the model (training data).
3. The data set has somehow been poisoned by additions and/or deletions in such a way that it misrepresents the problem.

Key Principle Number Five

Don’t be a one widget wonder; integrate multiple paradigms so the strengths of one compensate for the weaknesses of another.

There is no perfect data mining tool or application; every tool and application is going to have certain strengths and weaknesses. There will be aspects of the problem a particular tool handles well, and others it does not handle well. This means that if you rely entirely on one or two tools to attack a complex data mining problem you’ll probably not do the best job on all aspects of the problem.

An industry phrase for an analyst who relies entirely on one application is “a fool with a tool” (FWAT). The final product of an FWAT will consist of all the things their preferred tool does well, and none of the things it doesn’t.

Key Principle Number Six

Break the problem into the right pieces; divide and conquer.

A divide and conquer approach is often the key to success in difficult data mining problems. One of the factors that make some data mining problems difficult is the presence of several subproblems, not all of which can be addressed by a single approach. For example, if the data contain disparate items from multiple sources that are at different levels of precision/accuracy, or are in very different forms (such as a mixture of nominal and numeric data), using multiple tools might be the only viable approach.

When a problem has these local disparities, it makes sense to consider breaking it into pieces and deploying different methods against the pieces. This allows the use of the best tool on each component of the problem. Just as you wouldn’t fight a battle against a tank brigade using only infantry, you don’t want to use a single approach against a problem that has multiple modes or aspects that require different approaches. Instead, divide and conquer: break the problem into appropriate pieces and attack them individually with methods suited to the various data terrains (Fig. 3).

An additional benefit of problem segmentation is the ability to parallelize work by having different problem segments worked separately by different analysts. This can sometimes shorten the overall project schedule, and can keep staff usefully engaged, should there be slack periods in the project schedule.

Finally, it is often possible to automate the process of segmenting a difficult problem into natural chunks using unsupervised learning methods such as clustering. It is often the case that some chunks will be the hard part of the problem, and others the easy parts. Separating these allows elementary methods to handle the easy problem instances, with more computationally complex methods being used only when necessary.

Automation can make a big difference in data mining activities, but its limitations must be recognized. Automation is good for handling rote tasks that humans don’t do well, such as building histograms, computing correlation measures, grinding out bulk computations, doing clustering and matching, sifting through large quantities of data to perform repetitive statistical tests, and so on. But when it gets down to actually trying to squeeze the last bit of information out of a complex problem space, some reworking of the data is going to be required. By reworking the data we mean applying

![Multi-Paradigm Architecture](image-url)
some mathematical transform, alternate normalization, or synthesis of altogether new features by combining some of the old. All of these provide fresh views of the problem space that can make information not yet extracted more accessible to your tools.

Key Principle Number Seven

Work the data, not the tools, but automate when possible

If you’ve run a data mining problem through several spirals and have not made sufficient progress, at some point you will begin thinking about new things to try. Questions that pop up in this situation are: Do I need to modify my tools? Is there some perfect collection of parameter settings for my modeling software that will produce large improvements? (There are times when this is exactly what is called for, but it is usually not when you are running low on tricks. This leads to planless tinkering, which is addressed in key principle number eight.)

When faced with the question of whether to work the tools or work the data, the correct answer is almost always work the data. This is because the information you are looking for is not in the tools; it is in the data. Experience has shown over and over again that in mining of all sorts, data mining and otherwise, you are more likely to get increased value from better dirt than you are from a better shovel. Resorting to data mining by twiddling dials on a tool is a lot like doing your taxes by rolling dice: it’s just not the best approach.

Data mining is subject to the law of diminishing returns. Obvious patterns and principles are discovered early, but as information is mined from a data set, the rate of discovery will slow. The natural temptation when you’re in the hunt for information with automated tools is to speed things up by tweaking the tools to get them to act more effectively against the data (changing settings and parameters, recalibrating them, adjusting sensitivity levels, configurations, vigilance factors, etc.). This will work for a while, but at some point it will stop bearing fruit. It is important to realize when this stage has been reached, and the experienced data miner will switch from working the tool to working the data.

If you are quite sure that you’ve done everything with the data that makes sense, tweaking tools is worth a try. But to give it a passable chance of success, you should automate. The number of random experiments you can try manually with a tool is relatively small, and you are probably searching a large space of possible parameter settings. A better approach is to select a small data set, and then create some kind of a script that rapidly tries ranges of settings. Recognize, though, that solutions obtained in this manner often generalize poorly. This is to be expected, given that the solution was obtained by gaming the sensitivity of your tools to find just the right arcane configuration for this particular training set.

Key Principle Number Eight

Be systematic, consistent, and thorough; don’t lose the forest for the trees

It is very easy when you get into the heat of the information chase to begin trying long sequences of ad hoc experiments, hoping that something good will happen. When experimentation is run in an ad hoc manner, the risk is that the experimentation will not be systematic, won’t adequately cover the problem space, and will miss discoveries by leaving parts of the problem unexplored. Documentation and other methods designed to reduce errors also tend to suffer. When operating in an ad hoc manner, it’s easy to begin focusing on minutiae in the data, and lose sight of the fact that principled analysis is much more likely to be effective against a difficult problem than is luck. You don’t choose a career by flipping coins, and you can’t conduct good data science that way either.

Key Principle Number Nine

Document the work so it is reproducible

Just as in software engineering and hardware engineering, configuration management and version control is absolutely essential in data mining projects. In some ways it’s even more important to apply good documentation standards and audit control to data mining efforts, because these are by nature experimental: procedures for reproducing particular processes don’t yet exist. I’ve had the experience of making valuable discoveries in complex data sets, and then becoming so excited that I go into ad hoc mode before carefully documenting how I initially got them. Unless new processes are immediately documented, there is the very real possibility that you will be unable to reproduce the original breakthrough. This is a very frustrating position to be in: you know that the gold nugget is there; you’ve seen it and touched it but you can no longer find it.

To avoid this, when I do data mining work I keep a little notepad window open on my desktop so I can document the details of experiments as I perform them: what input files were used, what tool settings were made, what data conditioning sequence was applied, etc. This helps me reproduce any results obtained.

However, the best way to document work for reliable auditing and replication is to conduct the steps from script files that are executed automatically. In this way, not only is the process completely and unambiguously
documented at the moment it is conducted, but it’s easy to rerun the experiment later as well if necessary.

**Key Principle Number Ten**

Collaborate with team members, experts, and users to avoid surprises

We are not talking about good surprises here. Most surprises in science and engineering are not good. Data miners who are not DEs will discover things that DEs already know, and this is OK. The response to amazing discoveries must include review by a DE who can assess the practical value of pursuing them, because they might be well-known facts about the domain. If no mechanism for DE feedback is in place, much time will probably be spent rediscovering and documenting things that the expert already knows.

Another situation that can arise is the discovery of patterns in complex data that are real, but have no practical value in the solution of the users’ problem. I remember an activity during which I discovered a number of interesting patterns I was fairly sure had not been seen before. After devoting a lot of time and effort to the extraction and characterization of these patterns, I brought them to a DE for review. The expert pointed out that the patterns I discovered were real patterns that did exist in the data, and they were not well-known. They also had no value because they were noise artifacts randomly created during the collection of this particular data set, and would never occur again.

You don’t want to surprise the user too much. Even good surprises can have bad effects, because they disrupt plans and introduce technical uncertainty. It is important to keep both DEs and users engaged as mining work proceeds, so that as progress is made through the prototyping cycles, everyone is carried along in their understanding at essentially the same level. In this way, when the final delivery is made, everyone knows what to expect because they have seen the results unfold, and have a stake in it because they were involved. The most certain way to produce satisfied customers is to meet their expectations. Those expectations can change during the course of the project, but there won’t be a surprise if everyone is engaged.

**Key Principle Number Eleven**

Focus on the goal: maximum value to the user within cost and schedule

Data mining activities are undertaken to satisfy user needs. Data miners who love their analytics, their tools, and their methods can become distracted when interesting patterns are found or unexpected discoveries are made. These can initiate rabbit trails that cause the effort to go off track in pursuit of things that are interesting to the miner but not helpful to the user. It’s important to remember that no matter what happens during the mining effort, the goal is to provide value to the user, and to do so within the constraints of cost and schedule.

**TYPE OF MODELS: DESCRIPTIVE, PREDICTIVE, FORENSIC**

Like every well-designed process, data mining proceeds toward a goal. For almost all data mining efforts, this goal can be thought of as a model of some aspect of the problem domain. These models are of three kinds: descriptive, predictive, and forensic. Understanding the differences brings us to the notion of a domain ontology.

**Domain Ontologies as Models**

For philosophers, an ontology is a theory of being. It is an attempt to conceptualize answers to certain fundamental questions, such as: What is reality? Do things have meaning? What can be known?

In data mining, ontology has a similar meaning. Formally speaking, an ontology for a data set is a representational scheme that provides a consistent, coherent, unifying description of the domain data in context. Loosely speaking, an ontology for a data set is an interpretation of the data that reveals its meaning (s) by explaining its characteristics (sources, history, relationships, connotations). These representations are called models. Models represent ontologies in various ways: as consistent sets of equations describing data relationships and patterns (mathematical models), as coherent collections of empirical laws or principles (scientific models), etc.

When models merely describe patterns in data, they are called descriptive models. When models process patterns in data, they are called predictive models. When models discover or explain patterns in data, they are called forensic models. In practice, models often have elements of all three; the distinctions among them are not always clear.

The process by which scientists devise physical theories (the so-called scientific method) is a good example of a systematic development strategy for an ontology. The scientist comes to the experimental data with a minimum of a priori assumptions, intending to formulate and test a model that will explain the data. The scientific method is, in fact, one particular incarnation of the general data mining process.

Looking at how data has been used by scientists through history, we see an evolutionary development
through three stages: descriptive; predictive; and explanatory use of data.

Early scientific models were long on description, short on prediction, and shorter on coherent understanding. In many ways, the maturity of a scientific discipline can be assessed by where it lies along this progression. This is seen in its classic form in development of the cosmology of the solar system. The Pythagorean and Aristotelian models were mostly descriptive, with predictive and forensic components based mostly upon philosophical speculation. Ptolemy created a descriptive model that was useful for prediction, but it was ultimately made subservient to a philosophical model that hindered its development. It took a revolution in thought, begun by Copernicus and Galileo, and completed by Newton, to correct and extend Ptolemy’s patchwork cosmology to a coherent system derivable in a rational way from first principles (Fig. 4).

The same three modes (description, prediction, and explanation) that are operative in other instances of scientific reasoning are seen in data mining. They lie in the same natural hierarchy from simplest to most complex: descriptive, predictive, and forensic data mining.

**Descriptive Models**

Descriptive applications are the simplest; they use data to describe its source or context. This is simple because it leaves the work of interpretation entirely to the recipient: descriptive data is what it is, and says exactly (and only) what it says. Data mining operating in this mode produces meta-data: descriptive statistics such as averages, counts, rates, charts, and plots.

**Predictive Models**

Predictive applications of data are somewhat more complex, in that they add to the data some external assumptions about repeatability and fidelity. Prediction is data in action, the observed facts of past experience moving into the present and future. However, prediction is almost always based upon mere correlation: when these data are observed, certain conditions often co-occur. The point is, while this might look like intelligence, it can be done quite well without having any authentic understanding of the data at all. David Hume, the British philosopher, referred to this kind of reliable coincidence as the “constant conjunction of ideas”: it looks like implication or causation, but it is really an unexplained sequence for which exceptions have not yet been observed.

**Forensic Models**

Forensic applications are the most sophisticated data mining applications, because they interpret data in light of some context. This is an act of assigning meaning to symbolic patterns; in so doing, forensics generates semantic material from syntactic material. This is not just an enrichment process, but an elevation from one realm to another. This elevates data from the level of sense experience to understanding. This is data mining at its best.

**DATA MINING METHODOLOGIES**

Like any process, data mining can be carried out either haphazardly or systematically. While there is no universally accepted data mining process standard, there are several contenders for the position of de facto standard. Among these, two are preeminent, due largely to their association with widely used products: Sample, Explore, Modify, Model, Assess (SEMMA), created by the SAS Institute and supported by the tool SAS Enterprise Miner; and CRoss-Industry Standard Process for data mining, created by a consortium consisting of NCR, Daimler-Chrysler, SPSS, and OHRA, and supported by the tool SPSS Clementine.

A review of the various data mining processes that have been proposed as industry standards leaves one with the impression that underneath it all, roughly the same process is being described in different ways. This conclusion is supported by the fact that proponents of competing process standards rarely debate the merits of their favored process. Instead, they focus on how well their favored process works when applied in conjunction with their favored data mining tool. This makes sense
Decision

Healthcare–Data Mining:
given that the creators of the most popular processes are closely associated with tool developers.

The principal implication of all this is that, in actual practice, most data mining activities conducted by experienced analysts proceed in about the same general way: data are gathered, conditioned, and analyzed, giving descriptive models; then if desired, the results of the analysis are used to construct, validate, and field models. Each of these activities has multiple steps, requires the application of particular techniques, and has its own best practice. It is these steps, techniques, and practices that data mining process standards seek to specify.

Rather than sacrifice generality by describing the ad hoc details of just one of the competing standards, or blur the fundamentals by trying to survey them all, the following data mining process discussion will be conducted at a higher level of generality. With this done, each of the competing process standards will be seen as customizations for a particular problem type or tool set.

Conventional System Development: Waterfall Process

The standard development methodology used in many types of engineering is referred to as the Waterfall Process. In this process, projects proceed in order through a sequence of planning and development activities that culminates in the delivery of a capability specified by the user at the beginning of the effort.

This approach has some nice characteristics, and some shortcomings. On the plus side, it is inherently linear (in time), and has the look and feel of an orderly process that can be planned and managed. At each step along the way, you have some idea of how you are doing against the original plan, and how much work is left to do. It facilitates manpower and resource planning, and supports the prediction of project events. Because it has natural temporal and effort-level components, it supports cost estimation, and replanning if necessary.

On the minus side, it requires important decisions to be made at a time in the project when the least information is available. Users must fully express their requirements before work begins, and the resources required to cope with unanticipated problems and risks must be estimated in advance. Often, assumptions must be made about critical issues before they are fully understood.

In particular, since data mining has investigative activities woven throughout, the Waterfall Process is not at all natural for data mining projects. How do you schedule discovery? What skill mix do you need to implement an algorithm that doesn’t exist to solve a problem you don’t know about? If a discovery halfway through the project proves that a completely new approach with new goals is needed, what do you do?

A development process that can accommodate the impact of changes in problem understanding and project goals is needed for data mining: a rapid prototyping process. This and related methods are referred to as examples of Rapid Application Development (RAD) methodologies.

Data Mining as Rapid Prototyping

In practice, data mining is almost always conducted in a Rapid Prototyping fashion. Data miners using this methodology perform project work in a sequence of time-limited, goal-focused cycles (called spirals). Before jumping into the details of each step of this methodology, we begin with a summary overview of the steps that constitute a generic spiral:

- Step 1: Problem Definition
- Step 2: Data Evaluation
- Step 3: Feature Extraction and Enhancement
- Step 4a: Prototyping Plan
- Step 4b: Prototyping/Model Development
- Step 5: Model Evaluation
- Step 6: Implementation

The order and content of these steps will vary from spiral to spiral to accommodate project events. This is very different from the Waterfall Process, and addresses most of the issues that make the Waterfall Process ill-suited to data mining projects. Most importantly, rapid prototyping enables data mining researchers and developers to accommodate and benefit from incremental discovery, and holds at abeyance final decisions on some requirements until they can be settled in an informed and principled way.

The flexibility inherent in an RAD methodology does not mean that data mining efforts are undisciplined. On the contrary, using an RAD methodology in an disciplined manner usually results in the waste of resources, and perhaps project failure. This puts the onus on the program manager and technical lead to make sure that everything done during a spiral is done for a reason, that the outcomes are carefully evaluated in light of those reasons and that the team does not lose sight of the long-term goal: satisfying user needs within cost and schedule.

A GENERIC DATA MINING PROCESS

More formal characterizations of the data mining process are discussed elsewhere in this encyclopedia. However, we will complete our informal characterization of a single spiral by reciting the process as a two-paragraph narrative:
We begin by developing a clear understanding of what is to be accomplished in collaboration with the user. We then interview subject-matter experts using a (more or less formal) knowledge acquisition protocol. Next, we prepare the available data (evidence and hypotheses) for analysis. This involves inferring descriptive information in the form of meta-data (schemas), demographics (counts, ranges, distributions, visualizations), and data problems (conformation problems, outliers, gaps, class collisions, and population imbalance, all of which are covered by other entries).

Based upon this descriptive modeling work, data are repaired if necessary, and transformed for analysis. Appropriate pattern processing methods and applications are used to extract and enhance features for model construction. Appropriate modeling paradigm(s) is/are selected (e.g., rule-based system, decision tree, and support vector machine) and then integrated. The transformed data are then segmented, and models are built, evaluated, optimized, and applied. The results are interpreted in light of the goals for this spiral, and hypotheses and plans are adjusted. All work is documented and discussed with DEs.

Each of the steps in a spiral advance us around the spiral found in Fig. 5.

Each of the steps in a RAD data mining methodology will be addressed in detail in the following sections. Keep in mind that RAD is intended to be a flexible process that enables researchers and developers to accommodate both new information and changing user requirements. It is not intended, nor is it recommended, that the work pattern described here be imposed in a rigid manner because to do so would eliminate its principal advantages.

**RAD SKILL SET DESIGNATORS**

In later discussions of the RAD methodology, we will have occasion to discuss specific skill sets as they are tasked during the steps of an RAD effort. For our purposes in understanding the RAD methodology, it is assumed these skills will be held by specialists designated by the following titles and their abbreviations:

- CE: Cognitive Engineer (the data mining analyst)
- DE: DE (usually a customer/end-user having in-depth operational knowledge of the domain)
- DO: DOs (usually MIS personnel who collect/manage the user data)
- PM: Program Manager (manager who can make cost/schedule decisions for the effort)
- SE: Software/Database Engineer (application developer)

**SUMMARY**

Having read this entry, you now have a deeper understanding of the fundamental principles of data mining. You understand data mining as a process of discovery and exploitation that is conducted in multiple spirals consisting of multiple steps. You have been introduced to the steps that constitute an RAD cycle. Most importantly, you can explain the relative strengths and weaknesses of conventional and RAD development methodologies as they relate to data mining projects. Now that you have a summary understanding of data mining as a structured process, you are ready to begin your study of the individual steps that comprise it.