

Chapter 1: Data Exploration as a Process

Overview

Data exploration starts with data, right? Wrong! That is about as true as saying that making sales starts with products.

Making sales starts with identifying a need in the marketplace that you know how to meet profitably. The product must fit the need. If the product fits the need, is affordable to the end consumer, and the consumer is informed of your product's availability (marketing), then, and only then, can sales be made. When making sales, meeting the needs of the marketplace is paramount.

Data exploration also starts with identifying a need in its "marketplace" that can be met profitably. Its marketplace is corporate decision making. If a company cannot make correct and appropriate decisions about marketing strategies, resource deployment, product distribution, and every other area of corporate behavior, it is ultimately doomed. Making correct, appropriate, and informed business decisions is the paramount business need. Data exploration can provide some of the basic source material for decision making—information. It is information alone that allows informed decision making.

So if the marketplace for data exploration is corporate decision making, what about profit? How can providing any information not be profitable to the company? To a degree, any information is profitable, but not all information is equally useful. It is more valuable to provide accurate, timely, and useful information addressing corporate strategic problems than about a small problem the company doesn't care about and won't deploy resources to fix anyway. So the value of the information is always proportional to the scale of the problem it addresses. And it always costs to discover information. Always. It takes time, money, personnel, effort, skills, and insight to discover appropriate information. If the cost of discovery is greater than the value gained, the effort is not profitable.

What, then, of marketing the discovered information? Surely it doesn't need marketing. Corporate decision makers know what they need to know and will ask for it—won't they? The short answer is no! Just as you wouldn't even go to look for stereo equipment unless you knew it existed, and what it was good for, so decision makers won't seek information unless they know it can be had and what it is good for. Consumer audio has a great depth of detail that needs to be known in order to select appropriate equipment. Whatever your level of expertise, there is always more to be known that is important—once you know about it. Speakers, cables, connectors, amplifiers, tuners, digital sound recovery, distortion, surround sound, home theater, frequency response. On and on goes the list, and detailed books have been written about the subject. In selecting audio equipment (or anything else for that matter), an educated consumer makes the best choice. It is exactly

the same with information discovered using data exploration.

The consumers are decision makers at all levels, and in all parts of any company. They need to know that information is available, as well as the sort of information, its range of applicability, limits to use, duration of applicability, likely return, cost to acquire, and a host of other important details. As with anything else, an educated consumer makes the best use of the resource available. But unlike home audio equipment, each problem in data exploration for business is unique and has needs different from other problems. It has not yet become common that the decision maker directly explores broadly based corporate data to discover information. At the present stage of data exploration technology, it is usual to have the actual exploration done by someone familiar with the tools available—the *miner*. But how are the miner and the decision maker(s) to stay “in synch” during the process? How is the consumer, the decision maker, to become educated about reasonable expectations, reasonable return, and appropriate uses of the discovered information?

What is needed is a process. A process that works to ensure that all of the participants are engaged and educated, that sets appropriate expectations, and that ensures the most value is obtained for the effort put in. That process is the data exploration process, introduced in this chapter.

1.1 The Data Exploration Process

Data exploration is a practical multistage business process at which people work using a structured methodology to discover and evaluate appropriate problems, define solutions and implementation strategies, and produce measurable results. Each of the stages has a specific purpose and function. This discussion will give you a feel for the process: how to decide what to do at each stage and what needs to be done. This is a look at what goes in, what goes on, and what comes out of data exploration. While much of this discussion is at a conceptual level, it provides some practical “hands-on” advice and covers the major issues and interrelationships between the stages.

At the highest-level overview, the stages in the data exploration process are

1. Exploring the Problem Space
2. Exploring the Solution Space
3. Specifying the Implementation Method
4. Mining the Data (three parts)
 - a. Preparing the Data

b. Surveying the Data

c. Modeling the Data

This is the “map of the territory” that you should keep in mind as we visit each area and discuss issues. Figure 1.1 illustrates this map and shows how long each stage typically takes. It also shows the relative importance of each stage to the success of the project. Eighty percent of the importance to success comes from finding a suitable problem to address, defining what success looks like in the form of a solution, and, most critical of all, implementing the solution. If the final results are not implemented, it is impossible for any project to be successful. On the other hand, mining—preparation, surveying, and modeling—traditionally takes most of the time in any project. However, after the importance of actually implementing the result, the two most important contributors to success are solving an appropriate problem and preparing the data. While implementing the result is of the first importance to success, it is almost invariably outside the scope of the data exploration project itself. As such, implementation usually requires organizational or procedural changes inside an organization, which is well outside the scope of this discussion. Nonetheless, implementation is critical, since without implementing the results there can be no success.

Data exploration project		
	Time to complete (percent of total)	Importance to success (percent of total)
1. Exploring the problem	10	15
2. Exploring the solution	9	14
3. Implementation specification	1	51
4. Data mining		
a. Data preparation	60	15
b. Data surveying	15	3
c. Data modeling	5	2
	20	80
	80	20

Figure 1.1 Stages of a data exploration project showing importance and duration of each stage.

1.1.1 Stage 1: Exploring the Problem Space

This is a critical place to start. It is also the place that, without question, is the source of most of the misunderstandings and unrealistic expectations from data mining. Quite aside from the fact that the terms “data exploration” and “data mining” are (incorrectly) used interchangeably, data mining has been described as “a worm that crawls through your data and finds golden nuggets.” It has also been described as “a method of automatically

extracting unexpected hidden patterns from data.” It is hard to see any analogous connection between either data exploration or data mining and metaphorical worms. As for automatically extracting hidden and unexpected patterns, there is some analogous truth to that statement. The real problem is that it gives no flavor for what goes into finding those hidden patterns, why you would look for them, nor any idea of how to practically use them when they are found. As a statement, it makes data mining appear to exist in a world where such things happen by themselves. This leads to “the expectation of magic” from data mining: wave a magic wand over the data and produce answers to questions you didn’t even know you had!

Without question, effective data exploration provides a disciplined approach to identifying business problems and gaining an understanding of data to help solve them. Absolutely no magic used, guaranteed.

Identifying Problems

The data exploration process starts by *identifying the right problems to solve*. This is not as easy as it seems. In one instance, a major telecommunications company insisted that they had already identified their problem. They were quite certain that the problem was *churn*. They listened patiently to the explanation of the data exploration methodology, and then, deciding it was irrelevant in this case (since they were sure they already understood the problem), requested a model to predict churn. The requested churn model was duly built, and most effective it was too. The company’s previous methods yielded about a 50% accurate prediction model. The new model raised the accuracy of the churn predictions to more than 80%. Based on this result, they developed a major marketing campaign to reduce churn in their customer base. The company spent vast amounts of money targeting at-risk customers with very little impact on churn and a disastrous impact on profitability. (Predicting churn and stopping it are different things entirely. For instance, the amazing discovery was made that unemployed people over 80 years old had a most regrettable tendency to churn. They died, and no incentive program has much impact on death!)

Fortunately they were persuaded by the apparent success, at least of the predictive model, to continue with the project. After going through the full data exploration process, they ultimately determined that the problem that should have been addressed was improving return from underperforming market segments. When appropriate models were built, the company was able to create highly successful programs to improve the value that their customer base yielded to them, instead of fighting the apparent dragon of churn.

The value of finding and solving the appropriate problem was worth literally millions of dollars, and the difference between profit and loss, to this company.

Precise Problem Definition

So how is an appropriate problem discovered? There is a methodology for doing just this.

Start by defining problems in a precise way. Consider, for a moment, how people generally identify problems. Usually they meet, individually or in groups, and discuss what they feel to be precise descriptions of problems; on close examination, however, they are really general statements. These general statements need to be analyzed into smaller components that can, in principle at least, be answered by examining data. In one such discussion with a manufacturer who was concerned with productivity on the assembly line, the problem was expressed as, "I really need a model of the Monday and Friday failure rates so we can put a stop to them!" The owner of this problem genuinely thought this was a precise problem description.

Eventually, this general statement was broken down into quite a large number of applicable problems and, in this particular case, led to some fairly sophisticated models reflecting which employees best fit which assembly line profiles, and for which shifts, and so on. While exploring the problem, it was necessary to define additional issues, such as what constituted a failure; how failure was detected or measured; why the Monday and Friday failure rates were significant; why these failure rates were seen as a problem; was this in fact a quality problem or a problem with fluctuation of error rates; what problem components needed to be looked at (equipment, personnel, environmental); and much more. By the end of the problem space exploration, many more components and dimensions of the problem were explored and revealed than the company had originally perceived.

It has been said that a clear statement of a problem is half the battle. It is, and it points directly to the solution needed. That is what exploring the problem space in a rigorous manner achieves. Usually (and this was the case with the manufacturer), the exploration itself yields insights without the application of any automated techniques.

Cognitive Maps

Sometimes the problem space is hard to understand. If it seems difficult to gain insight into the structure of the problem, or there seem to be many conflicting details, it may be helpful to structure the problem in some convenient way. One method of structuring a problem space is by using a tool known as a *cognitive map* (Figures 1.2(a) and 1.2(b)). A useful tool for exploring complex problem spaces, a cognitive map is a physical picture of what are perceived as the objects that make up the problem space, together with the interconnections and interactions of the variables of the objects. It will very often show where there are conflicting views of the structure of the problem.



Figure 1.2 Cognitive maps: simple (a) and complex (b).

Figure 1.2(a) shows a simple cognitive map expressing the perceived relationships among the amount of sunshine, the ocean temperature, and the level of cloud cover. Figure 1.2(b) shows a somewhat more complex cognitive map. Cloud cover and global albedo are significant in this view because they have a high number of connections, and both introduce negative feedback relationships. Greenhouse gases don't seem to be closely coupled. A more sophisticated cognitive map may introduce numerical weightings to indicate the strength of connections. Understanding the implications of the more complex relationships in larger cognitive maps benefits greatly from computer simulation.

Note that what is important is not to resolve or remove these conflicting views, but to understand that they are there and exactly in which parts of the problem they occur. They may in fact represent valid interpretations of different views of a situation held by different problem owners.

Ambiguity Resolution

While the problems are being uncovered, discovered, and clarified, it is important to use techniques of *ambiguity resolution*. While ambiguity resolution covers a wide range of areas and techniques, its fundamental purpose is to assure that the mental image of the problem in the problem owner's mind—a mental image replete with many associated assumptions—is clearly communicated to, and understood by, the problem solver—most specifically that the associated assumptions are brought out and made clear. Ambiguity resolution serves to ensure that where there are alternative interpretations, any assumptions are explicated. For a detailed treatment of ambiguity resolution, see the excellent *Exploring Requirements: Quality Before Design* by Grause and Weinberg. (See Further Reading.)

Pairwise Ranking and Building the Problem Matrix

Exploring the problem space, depending on the scope of the project, yields anything from tens to hundreds of possible problems. Something must be done to deal with these as there may be too many to solve, given the resources available. We need some way of deciding which problems are the most useful to tackle, and which promise the highest yields for the time and resources invested.

Drawing on work done in the fields of decision theory and econometrics, it is possible to use a rationale that does in fact give consistent and reliable answers as to the most appropriate and effective problems to solve: the *pairwise ranking*. Figure 1.3 illustrates the concept. Generating pairwise rankings is an extremely powerful technique for reducing comparative selections. Surprisingly, pairwise rankings will probably give different results than an intuitive ranking of a list. Here is a simple technique that you can use to experiment.

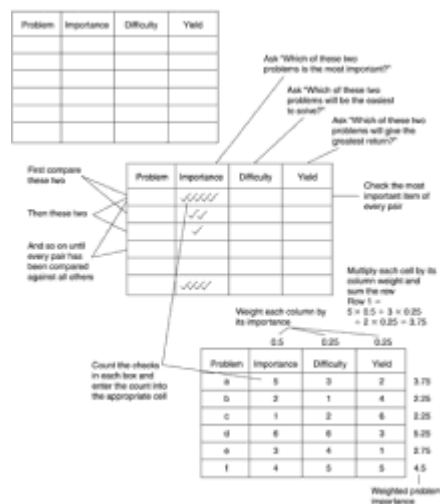


Figure 1.3 Pairwise ranking method. This method is illustrative only. In practice, using a spreadsheet or a decision support software package would ease the comparison.

Create a four-column matrix. In column 1, list 10–20 books, films, operas, sports teams, or whatever subject is of interest to you. Start at the top of the list and pick your best, favorite, or highest choice, putting a “1” against it in column 2. Then choose your second favorite and enter “2” in column 2 and so on until there is a number against each choice in that column. This is an *intuitive ranking*.

Now start again at the top of the list in column 1. This time, choose which is the preferable pick between items 1 and 2, then 1 and 3, then 1 and 4, and so on to the last item. Then make your preferable picks between those labeled 2 and 3, 2 and 4, and so on. For each pair, put a check mark in column 3 against the top pick. When you have finished this, add up the check marks for each preferred pick and put the total in column 4. When you have

finished, column 4 cells will contain 1, 2, 3, 4, and so on, check marks. If there is a tie in any of your choices, simply make a head-to-head comparison of the tied items. In column 4, enter a “1” for the row with the most check marks, a “2” for the second-highest number, and so on. This fourth column represents your pairwise ranking.

There are many, well-founded psychological studies that show, among other things, that a human can make judgments about 7 (plus or minus 2) items at the same time. Thus an intuitive ranking with more than 10 items will tend to be inconsistent. However, by making a comparison of each pair, you will generate a consistent ranking that gives a highly reliable indicator of where each item ranks. Look at the results. Are your listings different? Which is the most persuasive listing of your actual preferences—the intuitive ranking or the pairwise ranking?

Using the principle of the comparison technique described above with identified problems forms the problem space matrix (PSM). An actual PSM uses more than a single column of judgment rankings—“Problem,” “Importance,” “Difficulty,” “Yield,” and “Final Rank,” for example. Remember that the underlying ranking for each column is always based on the pairwise comparison method described above.

Where there are many problem owners, that is, a number of people involved in describing and evaluating the problem, the PSM uses a consensus ranking made from the individual rankings for “Importance,” “Difficulty,” and “Yield.” For the column “Importance,” a ranking is made to answer the question “Which of these two problems do you think is the most important?” The column “Difficulty” ranks the question “Given the availability of data, resources, and time, which of these two problems will be the easier to solve?” Similarly for “Yield,” the question is “If you had a solution for each of these two problems, which is likely to yield the most value to the company?” If there are special considerations in a particular application, an additional column or columns might be used to rank those considerations. For instance, you may have other columns that rank internal political considerations, regulatory issues, and so on.

The “Final Rank” is a weighted scoring from the columns “Importance,” “Difficulty,” and “Yield,” made by assigning a weight to each of these factors. The total of the weights must add up to 1. If there are no additional columns, good preliminary weightings are

Importance	0.5
Difficulty	0.25
Yield	0.25

This is because “Importance” is a subjective weighting that includes both “Difficulty” and “Yield.” The three are included for balance. However, discussion with the problem owners may indicate that they feel “Yield,” for example, is more important since benefit to the

company outweighs the difficulty of solving the problem. Or it may be that time is a critical factor in providing results and needs to be included as a weighted factor. (Such a column might hold the ranks for the question, “Which of these two will be the quickest to solve?”)

The final ranking is made in two stages. First, multiplying the value in each column by the weighting for that column creates a score. For this reason it is critical to construct the questions for each column so that the “best” answer is always the highest or the lowest number in all columns. Whichever method you chose, this ranks the scores from highest to lowest (or lowest to highest as appropriate).

If completed as described, this matrix represents the best selection and optimum ranking of the problems to solve that can be made. Note that this may not be the absolute best selection and ranking—just the best that can be made with the resources and judgments available to you.

Generating real-world matrixes can become fairly complex, especially if there are many problems and several problem owners. Making a full pairwise comparison of a real-world matrix having many problems is usually not possible due to the number of comparisons involved. For sizeable problems there are a number of ways of dealing with this complexity. A good primer on problem exploration techniques is *The Thinker’s Toolkit* by Morgan D. Jones (see Further Reading). This mainly focuses on decision making, but several techniques are directly applicable to problem exploration.

Automated help with the problem ranking process is fairly easy to find. Any modern computer spreadsheet program can help with the rankings, and several decision support software packages also offer help. However, new decision support programs are constantly appearing, and existing ones are being improved and modified, so that any list given here is likely to quickly become out of date. As with most other areas of computer software, this area is constantly changing. There are several commercial products in this area, although many suitable programs are available as shareware. A search of the Internet using the key words “decision support” reveals a tremendous selection. It is probably more important that you find a product and method that you feel comfortable with, and will actually use, than it is to focus on the particular technical merits of individual approaches and products.

1.1.2 Stage 2: Exploring the Solution Space

After discovering the best mix of precisely defined problems to solve, and ranking them appropriately, does the miner now set out to solve them? Not quite. Before trying to find a solution, it helps to know what one looks like!

Typical outputs from simple data exploration projects include a selection from some or all of the following: reports, charts, graphs, program code, listings of records, and algebraic formulae, among others. What is needed is to specify as clearly and completely as

possible what output is desired (Figure 1.4). Usually, many of the problems share a common solution.

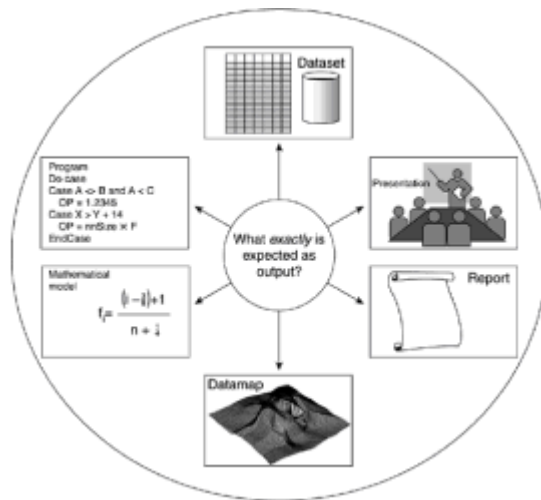


Figure 1.4 Exactly how does the output fit into the solution space?

For example, if there are a range of problems concerning fraudulent activity in branch offices, the questions to ask may include: What are the driving factors? Where is the easiest point in the system to detect it? What are the most cost-effective measures to stop it? Which patterns of activity are most indicative of fraud? And so on. In this case, the solution (in data exploration terms) will be in the form of a written report, which would include a listing of each problem, proposed solutions, and their associated rankings.

If, on the other hand, we were trying to detect fraudulent transactions of some sort, then a solution might be stated as “a computer model capable of running on a server and measuring 700,000 transactions per minute, scoring each with a probability level that this is fraudulent activity and another score for confidence in the prediction, routing any transactions above a specific threshold to an operator for manual intervention.”

It cannot be emphasized enough that in the Solution Space Exploration stage, the specified solution must be precise and complete enough that it actually specifies a real-world, implementable solution to solve the problem. Keep in mind that this specification is needed for the data exploration process, not data mining. Data mining produces a more limited result, but still one that has to fit into the overall need.

A company involved in asset management of loan portfolios thought that they had made a precise solution statement by explaining that they wanted a ranking for each portfolio such that a rational judgment could be made as to the predicted performance. This sounds like a specific objective; however, a specific objective is *not* a solution specification.

The kind of statement that was needed was something more like “a computer program to run on a Windows NT workstation terminal that can be used by trained operators and that scores portfolios and presents the score as a bar graph . . .” and so on. The point here is that the output of the data exploration process needed to be made specific enough so that the solution could be practically implemented. Without such a specific target to aim at, it is impossible to mine data for the needed model that fits with the business solution. (In reality, the target must be expected to move as a project continues, of course. But the target is still needed. If you don’t know what you’re aiming at, it’s hard to know if you’ve hit it!)

Another company wanted a model to improve the response to their mailed catalogs. Discovering what they really needed was harder than creating the model. Was a list of names and addresses needed? Simply a list of account numbers? Mailing labels perhaps? How many? How was response to be measured? How was the result to be used? It may seem unlikely, but the company had no clear definition of a deliverable from the whole process. They wanted things to improve in general, but would not be pinned down to specific objectives. It was even hard to determine if they wanted to maximize the number of responses for a given mailing, or to maximize the value per response. (In fact, it turned out—after the project was over—that what they really wanted to do was to optimize the value per page of the catalog. Much more effective models could have been produced if that had been known in advance! As it was, no clear objective was defined, so the models that were built addressed another problem they didn’t really care about.)

The problems and difficulties are compounded enormously by not specifying what success looks like in practice.

For both the problem and the solution exploration it is important to apply ambiguity resolution. This is the technique that is used to test that what was conceived as a problem is what was actually addressed. It also tests that what is presented as a solution is what was really wanted by the problem owners. Ambiguity resolution techniques seek to pinpoint any misunderstandings in communication, reveal underlying assumptions, and ensure that key points and issues are understood by everyone involved. Removing ambiguity is a crucial element in providing real-world data exploration.

1.1.3 Stage 3: Specifying the Implementation Method

At this point, problems are generated and ranked, solutions specified, expectations and specifications matched, and hidden assumptions revealed.

However, no data exploration project is conducted just to discover new insights. The point is to apply the results in a way that increases profitability, improves performance, improves quality, increases customer satisfaction, reduces waste, decreases fraud, or meets some other specified business goal. This involves what is often the hardest part of any successful data exploration project—modifying the behavior of an organization.

In order to be successful, it is not enough to simply specify the results. Very successful and potentially valuable projects have died because they were never seriously implemented. Unless everyone relevant is involved in supporting the project, it may not be easy to gain maximum benefit from the work, time, and resources involved.

Implementation specification is the final step in detailing how the various solutions to chosen problems are actually going to be applied in practice. This details the final form of the deliverables for the project. The specification needs to be a complete practical definition of the solution (what problem it addresses, what form it takes, what value it delivers, who is expected to use it, how it is produced, limitations and expectations, how long it is expected to last) and to specify five of the “six w’s”: who, how, what, when, and where (why is already covered in the problem specification).

It is critical at this point to get the “buy-in” of both “problem owners” and “problem holders.” The problem owners are those who experience the actual problem. The problem holders are those who control the resources that allow the solution to be implemented. The resources may be in one or more of various forms: money, personnel, time, or corporate policy, to name only a few. To be effective, the defined solution *must* be perceived to be cost-effective and appropriate by the problem holder. Without the necessary commitment there is little point in moving further with the project.

1.1.4 Stage 4: Mining the Data

Geological mining (coal, gold, etc.) is not carried out by simply applying mining equipment to a lump of geology. Enormous preparation is made first. Large searches are made for terrain that is geologically likely to hold whatever is to be mined. When a likely area is discovered, detailed surveys are made to pinpoint the most likely location of the desired ore. Test mines are dug before the full project is undertaken; ore is assayed to determine its fineness. Only when all of the preparation is complete, and the outcome of the effort is a foregone conclusion, is the full-scale mining operation undertaken.

So it should be with mining data. Actually mining the data is a multistep process. The first step, preparation, is a two-way street in which both the miner is prepared and the data is prepared. It is not, and cannot be, a fully autonomous process since the objective is to prepare the miner just as much as it is to prepare the data. Much of the actual data preparation part of this first and very important step can be automated, but miner interaction with the data remains essential. Following preparation, the survey. For effective mining this too is most important. It is during the survey that the miner determines if the data is adequate—a small statement with large ramifications, and more fully explored in [Chapter 11](#).

When the preparation and survey are complete, actually modeling the data becomes a relatively small part of the overall mining effort. The discovery and insight part of mining

comes during preparation and surveying. Models are made only to capture the insights and discoveries, not to make them. The models are built only when the outcome is a foregone conclusion.

Preparing the Data for Modeling

Why prepare data? Why not just take it as it comes? The answer is that preparing data also prepares the miner so that when using prepared data, the miner produces better models, faster.

Activities that today come under the umbrella of the phrase “data mining” actually have been used for many years. During that time a lot of effort has been put forth to apply a wide variety of techniques to data sets of many different types, building both predictive and inferential models. Many new techniques for modeling have been developed over that time, such as evolution programming. In that same time other modeling tools, such as neural networks, have changed and improved out of all recognition in their capabilities. However, what has not changed at all, and what is almost a law of nature, is GIGO—garbage in, garbage out. Keeping that now-popular aphorism firmly in mind leads logically to the observation that good data is a prerequisite for producing effective models of any type.

Unfortunately, there is no such thing as a universal garbage detector! There are, however, a number of different types of problems that constantly recur when attempting to use data sets for building the types of models useful in solving business problems. The source, range, and type of these problems, the “GI” in GIGO, are explored in detail starting in [Chapter 4](#). Fortunately, there are a number of these problems that are more or less easily remedied. Some remedies can be applied automatically, while others require some choices to be made by the miner, but the actual remedial action for a wide range of problems is fairly well established. Some of the corrective techniques are based on theoretical considerations, while others are rules of thumb based on experience. The difficulty is in application.

While methodologies and practices that are appropriate for making models using various algorithms have become established, there are no similar methodologies or practices for using data preparation techniques. Yet good data preparation is essential to practical modeling in the real world.

The data preparation tools on the accompanying CD-ROM started as a collection of practical tools and techniques developed from experience while trying to “fix” data to build decent models. As they were developed, some of them were used over and over on a wide variety of modeling projects. Their whole purpose was to help the miner produce better models, faster than can be done with unprepared data, and thus assure that the final user received cost-effective value. This set of practical tools, in the form of a computer program, and a technique of applying the program, must be used together to

get their maximum benefit, and both are equally important. The accompanying demonstration software actually carries out the data manipulations necessary for data preparation. The technique is described as the book progresses. Using this technique results in the miner understanding the data in ways that modeling alone cannot reveal. Data preparation is about more than just readying the data for application of modeling tools; it is also about gaining the necessary insights to build the best possible models to solve business problems with the data at hand.

One objective of data preparation is to end with a prepared data set that is of maximum use for modeling, in which the natural order of the data is least disturbed, yet that is best enhanced for the particular purposes of the miner. As will become apparent, this is an almost totally different sort of data preparation activity than is used, say, in preparing data for data warehousing. The objective, techniques, and results used to prepare data when mining are wholly different.

The Prepared Information Environment (PIE)

A second objective of data preparation is to produce the Prepared Information Environment (PIE). The PIE is an active computer program that “envelops” the modeling tools to protect them from damaged and distorted data. The purpose and use of this very important tool in modeling is more fully described in Chapter 3. Its main purposes are to protect the modeling tool from damaged data and to maximally expose the data set’s information content to the modeling tool. One component, the Prepared Information Environment Input module (PIE-I) does this by acting as an intelligent buffer between the incoming data, manipulating the training, testing, and execution data sets before the modeling tool sees the data. Since even the output prediction variables are prepared by the PIE-I, any model predictions are predictions of the prepared values. The predictions of prepared values need to be converted back into their unmodified form, which is done by the Prepared Information Environment Output module (PIE-O).



A clear distinction has to be made between the training and testing data set, and the execution data set. On some occasions the training, testing, and execution data sets may all be drawn from the same “pool” of data that has been assembled prior to modeling. On other occasions the execution data may be impossible to obtain at the time of modeling. In the case of industrial modeling, for instance, it may be required to build a model that predicts likely time to failure for a manufactured component based on the manufacturing information collected as it is manufactured. The model, when built, validated, and verified, will be placed in service to monitor future production. However, at the time the model is being built, using already collected data, the data on next month’s or next year’s production is impossible to acquire. The same is true for stock market data, or insurance claims data, for instance, where the model is built on data already collected, but applied to future stock movements or insurance claims.



In the continuously learning model described in the [Supplemental Material](#) section at the

end of this chapter, the actual data to be used for mailing was not available until it was acquired specifically for the mailing. The model built to predict likely responders to the mailing solicitation was built before the mailing data was available. The initial mailing response model was built on information resulting from previous mailings. It was known that the characteristics of the variables (described in [Chapter 2](#)) for the training data that was available were similar to those in the actual mailing data set—even though the precise data set for the mailing had not been selected.

In general, preparation of the data for modeling requires various adjustments to be made to the data prior to modeling. The model produced, therefore, is built using adjusted, prepared data. Some mechanism is needed to ensure that any new data, especially data to which the model is to be applied, is also adjusted similarly to the training data set. If this is not done, the model will be of no value as it won't work with raw data, only with data similarly prepared to that used for training.



It is the PIE that accomplishes this transformation. It may perform many other useful tasks as well, such as novelty detection, which measures how similar the current data is to that which was used for training. The various tasks and measures are discussed in detail in various parts of the book. However, a principal purpose of the PIE is to transform previously unencountered data into the form that was initially used for modeling. (This is done by the PIE-I.)

Notable too is that a predictive model's output variable(s), the one(s) that the model is trying to predict or explain, will also have been in its adjusted format, since the model was trying to predict or explain it in a prepared data set. The PIE also will transform the prepared and normalized model output into the experiential range encountered in the data before preparation—in other words, it undoes the transformations for the predicted values to get back the original range and type of values for the predicted output. (This is accomplished by the PIE-O.)

While the PIE adds great value in many other areas, its main function is allowing models trained on prepared data to be used on other data sets.

For one-shot modeling, where all of the data to be modeled and explained is present, the PIE's role is more limited. It is simply to produce a file of prepared data that is used to build the model. Since the whole of the data is present, the role of the PIE is limited to translating the output variables from the predicted adjusted value to their predicted actual expected value.



Thus, the expected output from the data preparation process is threefold: first, a prepared miner, second, a prepared data set, and third, the PIE, which will allow the trained model to be applied to other data sets and also performs many valuable ancillary functions. The PIE provides an envelope around the model, both at training and execution time, to insulate the model from the raw data problems that data preparation corrects.

Surveying the Data

Surveying the prepared data is a very important aspect of mining. It focuses on answering three questions: What's in the data set? Can I get my questions answered? Where are the danger areas? These questions may seem similar to those posed by modeling, but there is a significant difference.



Using the survey to look at the data set is different in nature from the way modeling approaches the data. Modeling optimizes the answer for some specific and particular problem. Finding the problem or problems that are most appropriate is what the first stage of data exploration is all about. Providing those answers is the role of the modeling stage of data mining. The survey, however, looks at the general structure of the data and reports whether or not there is a useful amount of information enfolded in the data set about various areas. The survey is not really concerned with exactly what that information might be—that is the province of modeling. A most particular purpose of the survey is to find out if the answer to the problem that is to be modeled is actually in the data prior to investing much time, money, and resource in building the model.



The survey looks at all areas of the data set equally to make its estimate of what information is enfolded in the data. This affects data preparation in that such a survey may allow the data to be restructured in some way prior to modeling, so that it better addresses the problem to be modeled.

In a rich data set the survey will yield a vast amount of insight into general relationships and patterns that are in the data. It does not try to explicate them or evaluate them, but it does show the structure of the data. **Modeling explores the fine structure; survey reveals the broad structure.**

Given the latter fact, the search for danger areas is easier. An example of a danger area is where some bias is detectable in the data, or where there is particular sparsity of data and yet variables are rapidly changing in value. **In these areas where the relationship is changing rapidly, and the data do not describe the area well, any model's performance should be suspect.** Perhaps the survey will reveal that the range in which the model predictions will be important is not well covered.



All of these areas are explored in much more detail in **Chapter 11**, although the perspective there is mainly on how the information provided by the survey can be used for better preparing the data. However, the essence of the data survey is to build an overall map of the territory before committing to a detailed exploration. Metaphorically speaking, it is of immense use to know where the major mountain ranges, rivers, lakes, and deserts are before setting off on a hiking expedition. It is still necessary to make the detailed exploration to find out what is present, but the map is the guide to the territory. Vacationers, paleontologists, and archeologists all use the same basic topographic map

to find their way to sites that interest them. Their detailed explorations are very different and may lead them to each make changes to the local, or fine, structure of their individual maps. However, without the general map it would be impossible for them to find their way to likely places for a good vacation site, dinosaur dig, or an ancient city. The general map—the data survey—shows the way.

Modeling the Data

When considering data mining, even some of the largest companies in the U.S. have asked questions whose underlying meaning was, “What sort of problems can I solve with a neural net (or other specific technique)?” This is exactly analogous to going to an architect and asking, “What sort of buildings can I build with this power saw (or other tool of your choice)?” The first question is not always immediately seen as irrelevant, whereas the second is.

Some companies seem to have the impression that in order to produce effective models, knowledge of the data and the problem are not really required, but that the tools will do all the work. Where this myth came from is hard to imagine. It is so far from the truth that it would be funny if it were not for the fact that major projects have failed entirely due to ignorance on the part of the miner. Not that the miner was always at fault. If ordered to “find out what is in this data,” an employee has little option but to do *something*. No one who expected to achieve anything useful would approach a lump of unknown substance, put on a blindfold, and whack at it with whatever tool happened to be at hand. Why this is thought possible with data mining tools is difficult to say!

Unfortunately, focusing on the data mining modeling tools as the primary approach to a problem often leads to the problem being formulated in inappropriate ways. Significantly, there may be times when data mining tools are not the right ones for the job. It is worth commenting on the types of questions that are particularly well addressed with a data-mined model. These are the questions of the “How do I . . . ?” and “Why is it that . . . ?” sort.

For instance, if your questions are those that will result in summaries, such as “What were sales in the Boston branch in June?” or “What was the breakdown by shift and product of testing failures for the last six weeks?” then these are questions that are well addressed by on-line analytical processing (OLAP) tools and probably do not need data mining. If however, the questions are more hypothesis driven, such as “What are the factors driving fraudulent usage in the Eastern sector?” or “What should be my target markets and what is the best feature mix in the marketing campaign to capture the most new customers?” then data mining, used in the context of a data exploration process, is the best tool for the job.

1.1.5 Exploration: Mining and Modeling

This brief look at the process of data exploration emphasizes that none of the pieces stands alone. Problems need to be identified, which leads to identifying potential solutions, which leads to finding and preparing suitable data that is then surveyed and finally modeled. Each part has an inextricable relationship to the other parts. Modeling, the types of tools and the types of models made, also has a very close relationship with how data is best prepared, and before leaving this introduction, a first look at modeling is helpful to set the frame of reference for what follows.








1.2 Data Mining, Modeling, and Modeling Tools




One major purpose for preparing data is so that mining can discover models. But what *is* modeling? In actual fact, what is being attempted is very simple. The ways of doing it may not be so simple, but the actual intent is quite straightforward.

It is assumed that a data set, either one immediately available or one that is obtainable, might contain information that would be of interest if we could only understand what was in it. Therein lies the rub. Since we don't understand the information that is in the data just by looking at it, some tool is needed that will turn the information enfolded in the data set into a form that is understandable. That's all. That's the modeling part of data mining—a process for transforming information enfolded in data into a form amenable to human cognition.

1.2.1 Ten Golden Rules

As discussed earlier in this chapter, the data exploration process helps build a framework for data mining so that appropriate tools are applied to appropriate data that is appropriately prepared to solve key business problems and deliver required solutions. This framework, or one similar to it, is critical to helping miners get the best results and return from their data mining projects. In addition to this framework, it may be helpful to keep in mind the 10 Golden Rules for Building Models:

1. Select clearly defined problems that will yield tangible benefits. 
2. Specify the required solution. 
3. Define how the solution delivered is going to be used. 
4. Understand as much as possible about the problem and the data set (the domain). 
5. Let the problem drive the modeling (i.e., tool selection, data preparation, etc.). 
6. Stipulate assumptions. 
7. Refine the model iteratively. 

8. Make the model as simple as possible—but no simpler. 
9. Define instability in the model (critical areas where change in output is drastically different for a small change in inputs). 
10. Define uncertainty in the model (critical areas and ranges in the data set where the model produces low confidence predictions/insights). 

In other words, rules 1–3 recapitulate the first three stages of the data exploration process. Rule 4 captures the insight that if you know what you’re doing, success is more likely. Rule 5 advises to find the best tool for the job, not just a job you can do with the tool. Rule 6 says don’t *just* assume, tell someone. Rule 7 says to keep trying different things until the model seems as good as it’s going to get. Rule 8 means KISS (Keep It Sufficiently Simple). Rules 9 and 10 mean state what works, what doesn’t, and where you’re not sure.

To make a model of data is to express the relationships that change in one variable, or set of variables, has on another variable or set of variables. Another way of looking at it is that regardless of the type of model, the aim is to express, in symbolic terms, the shape of how one variable, or set of variables, changes when another variable or set of variables changes, and to obtain some information about the reliability of this relationship. The final expression of the relationship(s) can take a number of forms, but the most common are charts and graphs, mathematical equations, and computer programs. Also, different things can be done with each of these models depending on the need. **Passive models** usually express relationships or associations found in data sets. These may take the form of the charts, graphs, and mathematical models previously mentioned. **Active models** take sample inputs and give back predictions of the expected outputs.

Although models can be built to accomplish many different things, the usual objective in data mining is to produce either predictive or explanatory (also known as inferential) models.

1.2.2 Introducing Modeling Tools

There are a considerable variety of data mining modeling tools available. A brief review of some currently popular techniques is included in **Chapter 12**, although the main focus of that chapter is the effect of using prepared data with different modeling techniques. Modeling tools extend analysis into producing models of several different types, some mentioned above and others examined in more detail below.

Data mining modeling tools are almost uniformly regarded as software programs to be run on a computer and that perform various translations and manipulations on data sets. These are indeed the tools themselves, but it does rather leave out the expertise and

domain knowledge needed to successfully use them. In any case, there are a variety of support tools that are also required in addition to the so-called data mining tools, such as databases and data warehouses, to name only two obvious examples. Quite often the results of mining are used within a complex and sophisticated decision support system. Close scrutiny often makes problematic a sharp demarcation between the actual data mining tools themselves and other supporting tools. For instance, is presenting the results in, say, an OLAP-type tool part of data mining, or is it some other activity?

In any case, since data mining is the discovery of patterns useful in a business situation, the venerable tools of statistical analysis may be of great use and value. The demarcation between statistical analysis and data mining is becoming somewhat difficult to discern from any but a philosophical perspective. There are, however, some clear pointers that allow determination of which activity is under way, although the exact tool being used may not be indicative. (This topic is also revisited in [Chapter 12](#).)

Philosophically and historically, statistical analysis has been oriented toward verifying and validating hypotheses. These inquiries, at least recently, have been scientifically oriented. Some hypothesis is proposed, evidence gathered, and the question is put to the evidence whether the hypothesis can reasonably be accepted or not. Statistical reasoning is concerned with logical justification, and, like any formal system, not with the importance or impact of the result. This means that, in an extreme case, it is quite possible to create a result that is statistically significant—and utterly meaningless.

It is fascinating to realize that, originally, the roots of statistical analysis and data mining lie in the gaming halls of Europe. In some ways, data mining follows this heritage more closely than statistical analysis. Instead of an experimenter devising some hypothesis and testing it against evidence, data mining turns the operation around. Within the parameters of the data exploration process, data mining approaches a collection of data and asks, “What are all the hypotheses that this data supports?” There is a large conceptual difference here. Many of the hypotheses produced by data mining will not be very meaningful, and some will be almost totally disconnected from any use or value. Most, however, will be more or less useful. This means that with data mining, the inquirer has a fairly comprehensive set of ideas, connections, influences, and so on. The job then is to make sense of, and find use for, them. Statistical analysis required the inquirer first to devise the ideas, connections, and influences to test.

There is an area of statistical analysis called “exploratory data analysis” that approaches the previous distinction, so another signpost for demarcation is useful. Statistical analysis has largely used tools that enable the human mind to visualize and quantify the relationships existing within data in order to use its formidable pattern-seeking capabilities. This has worked well in the past. Today, the sheer volume of data, in numbers of data sets, let alone quantity of data, is beyond the ability of humans to sift for meaning. So, automated solutions have been called into play. These automated solutions draw largely on techniques developed in a discipline known as “machine learning.” In

essence, these are various techniques by which computerized algorithms can, to a greater or lesser degree, learn which patterns actually do exist in data sets. They are not by any means as capable as a trained human mind, educated in the knowledge domain, would be. They are, however, formidably fast (compared to humans), tireless, consistent, and error-free for a particular class of errors. They are error-free in the sense that, once validated that they are indeed performing accurately, the output is consistent. Judgments about what the outputs mean remain firmly in the human domain. That is to say, while decisions as to particular actions to be taken under given circumstances can be programmed algorithmically, humans had to either explicitly program such switch points or permit the program to train and learn them. No amount of artificial intelligence reaches the level of sophistication represented by even human stupidity! In fact, appearances to the contrary, computer programs still cannot make self-motivated, intentional decisions.

Regardless of their source and how they are used (or misused), the function and purpose of modeling tools is actually very straightforward. It is to transform any of the required knowledge enfolded in a particular data set into a form useful to, or comprehensible by, humans. It may be both useful and comprehensible, but this is not necessarily so.

In marketing applications, for instance, models often have to be created where comprehensibility is not an issue. The marketing manager simply wants a model that delivers more, or more valuable, leads, customers, or orders. Why such a model works is not an issue, at least not until someone asks, "Why does that market segment produce better results?" A specific instance of this occurred with a company concerned with providing college students with funding to attend college. It had long been their practice to mail solicitations to people they felt would be appropriate candidates somewhat before the end of the school year, assuming that was the time when people were considering which college to attend and applying for financial aid. In order to investigate this further, marketing response models were made with a variety of their assumptions altered for a small subset of the mailing. Analysis of the results indicated strongly that mailing *immediately following the end of the school year* showed a stronger response. This seemed so counterintuitive to the marketers that they found it hard to accept and immediately asked why this was so. At this point a variety of different models drawing on different data sets had to be built to explore the question. (It turned out that, for the population segment for which this response was valid, colleges were explored first and the earlier solicitation had been thrown away as unwanted "junk mail" by the time financial aid for school was being considered. Early mailing meant that they weren't in the running for that segment of the population.)

This leads to consideration of the types of models that are used.

1.2.3 Types of Models

After conducting a data exploration project and stipulating the problem set, solution set, and implementation strategies, preparing the data, surveying it, then selecting algorithms

for the purpose, there still remains the process of building models and delivering the results.

First, a brief observation about modeling in general. A misconception of inexperienced modelers is that modeling is a linear process. This imagined linear process can be shown as

1. State the problem.
2. Choose the tool.
3. Get some data.
4. Make a model.
5. Apply the model.
6. Evaluate results.

On the contrary, *building any model should be a continuous process incorporating several feedback loops and considerable interaction among the components*. Figure 1.5 gives a conceptual overview of such a process. At each stage there are various checks to ensure that the model is in fact meeting the required objectives. It is a dynamic process in which various iterations converge toward the best solution. There is naturally a fair amount of human interaction and involvement in guiding the search for an optimum solution.

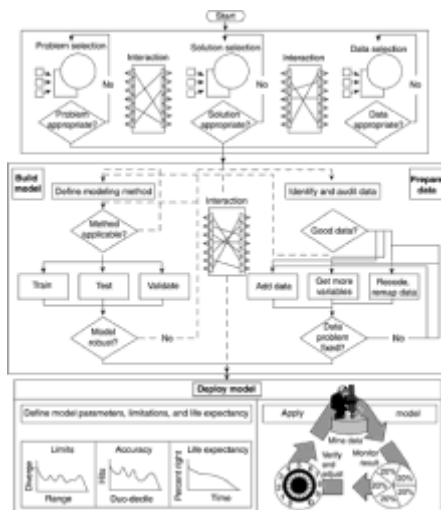


Figure 1.5 Model building outline.

The various types of models were briefly touched on previously, but discussing them

together helps clarify their similarities and differences.

1.2.4 Active and Passive Models

Basically, active models actually respond in some way, whereas passive models are nonreactive.

Passive models generally answer questions and show relationships using charts, graphs, words, mathematical formulae, and so on. The example above describing why some college applicants respond better to late mailings was a passive model. It explicated in an understandable way the “why” of the relationship. It was “actionable information” in that, as a result, better marketing plans can be made and characteristics of the targeted population can be identified. A model is passive in that it does not take inputs, give outputs, change, react, or modify anything as it is used. It is simply a fixed expression, such as a statement on a piece of paper.

On the other hand, an *active model* performs one or more activities. An active model built for the college loan application, for instance, might take a specific input file and score or categorize it as to the type of response to be expected for each instance (record) in the file.

The differentiation between active and passive models may be critical to the modeler and to the application. It will have a considerable effect on which data is selected for modeling. However, when preparing the selected data set, the difference between active and passive model requirements has little if any impact on how the data is further prepared for modeling.

1.2.5 Explanatory and Predictive Models

Here, of course, the one type of model is created to explain some facet of the data, while the other is designed to predict, classify, or otherwise interpret data. These are not synonymous with active and passive.

On occasion, particularly in the arena of industrial automation, the required output from the modeling process is a passive, predictive model. For instance, in a paper mill, where paper is made, the key parts of the process were captured in the shift foreman’s experience. At shift change, the new foreman, who had enormous experience, would make various adjustments based on such measures as the taste of the process (actually tasting the slurry as a means of measuring what was happening in the mixture) at a particular stage. Each foreman knew how to tune the process to produce fine paper. Each foreman knew what was going wrong when indeed things were going wrong, and how to fix them. Yet each foreman’s recipe was different!

The business problem here was that automating such a process seemed impossible. The

“rules” for making paper were embedded in the shift foremen’s heads, and extracting a useful set of rules by questioning them, although tried, proved impossible. Instead of studying the experts (shift foremen), the modeling approach was to instrument the paper mill, collect data about the process, analyze the collected data, and model the process. This type of approach is called *automated expertise capture*. This process involves watching and modeling what an expert actually does rather than questioning the expert to create a model.

It took considerable effort, but eventually successful passive predictive models were produced in the form of mathematical statements. These mathematical statements described how the process behaved, and how its behavior changed as conditions changed. To automate the paper-making process, these mathematical statements were turned into a particular sort of programming language called “ladder logic,” which is widely used in programmable logic controllers (PLCs). The passive, explanatory model was used to create the program for the PLCs. It essentially captured the expertise of the foremen and encapsulated it in succinct expressions. These, in turn, were used in machine and process automation.

Without giving detailed examples of each model type (which would properly belong in a book on modeling rather than data preparation), it can be easily seen that it is quite possible to have active-explanatory, passive-explanatory, active-predictive, or passive-predictive models.

Passive-predictive models can be exemplified in the “score cards” used to score certain credit applicants. These are really worksheets that loan officers can use. Modeling techniques have been used to improve the performance of such devices. The output is a fixed, passive worksheet printed on a form. It is, nonetheless, used as a predictive and classification tool by the user. However, note that the *output* of the modeling technique used is passive predictive.

1.2.6 Static and Continuously Learning Models

This is an interesting and important division of modeling that deserves a closer look, particularly the continuously learning models. These hold enormous promise for the application of the sophisticated techniques outlined here.

Static (One-Shot) Models

Static modeling is used to discover relationships or answer questions that are drawn from historical data. In point of fact, all data is historical. (If you have future data about, say, the stock market, please let me know!) However, in this context “historical” data can be taken to mean that the data set from which the model is built is not going to be updated with more current data. Questions leading to the building of static models might be similar to “What factors drive the failure modes in disk drive manufacture?” Once the failure modes

in manufacturing are analyzed, corrective action will be applied to fix problems, and that's that. Naturally, the process will be monitored to find out how well the "fix" worked, but the data previously collected is no longer representative of production since changes were made based on the failure mode's driving factors. If any further investigation into the problem is wanted, the historical data cannot address the new issues as systemic changes were made. New data representative of the modified system's performance would have to be collected.

Although pursuing answers to problems requiring static models can be a fairly complex undertaking and draw on the full resources of the tools available, as well as heavily relying on the experience of the modeler, producing the static models themselves is fairly straightforward. An answer in a fixed form, one that does not interact with data to modify itself, is the final solution.

Inexperienced modelers frequently see the static model, or a series of static models, as how modeling should take place. While static models are certainly an appropriate solution for many problems, they are very prevalent even where more extensive techniques are more appropriate. We will now examine one alternative.

Continuously Learning Models

These types of models represent a relatively hands-off, controlling, or discovering process working in dynamic conditions. Constructing a robust continuously learning model draws on resources from outside the domain of data exploration. The core, or enabling, technology, however, is data mining directed by the data exploration process.

Continuous learning is a system using an autonomous model containing a number of internal set points. One natural example of such a system is a human being. We contain many set points that control our behavior, one of which is internal temperature. The internal temperature of a healthy human being is estimated at about 98.6° Fahrenheit. That temperature may be regarded as a set point. Our bodies seek to maintain a constant internal temperature in spite of external assaults. We may be motivated to make a number of internal and external environmental adjustments to, say, keep warm when the external temperature is falling. These include turning up the thermostat, putting on more clothes, shivering, having a hot drink, and possibly a whole host of other activities. All the time we're actively manipulating the environment, both internal and external, to maintain the specific set point for internal temperature. It is exactly this type of behavior that is used in an artificially constructed, but still self-motivated model.

In artificial continuously learning systems, the primary set points are always externally specified; natural continuously learning systems may evolve suitable set points. The system evaluates incoming data and modifies its behavior in such a way as to modify those parameters of its environment that are more or less under its control so that the system maintains the set points. It is a self-adaptive system adjusting in real time to a

dynamic environment. It is continuously changing its internal structure to reflect its past experiences, and using those past experiences to modify its environment. If a continuously learning predictive model was given an identical input at different times, it may well produce totally different predictions—depending on what it had experienced, and the changes in its environment, in the interim. This is very different from a sequentially updated series of static models. The key is a continuous interaction between components.

As far as data preparation is concerned, the preparatory activities carried out when making static models tend to be manual. When continuously learning systems are deployed, however, the PIE that permits continuous, automated data preparation becomes a vital part of the whole process.

The easiest way to see what is involved in a continuously learning model is to examine a simplified actual application, and the Supplemental Material section at the end of this chapter briefly outlines a simplified application using a continuously learning model.

1.3 Summary

When discussing data mining, it is easy to think of the process only in terms of what various tools can do. This is exactly analogous to focusing on types of nails and what to do with them simply because a hammer collection is available. For sure, we will do different things with a 6-ounce ball-peen hammer, a 12-ounce claw hammer, and a 14-pound sledgehammer. However, the object of the exercise may be to knock down a wall, construct a house, repair a car door, or drive a railroad spike. It is the nature of the job to be done that determines which tool to use—not the other way around. So it is too with data mining. To obtain effective results when mining, focusing on the tools is not enough. This chapter has looked from the “100,000-foot level” at the whole process of data exploration, giving a perspective of where data mining fits within the process, and how data preparation, modeling, and the other components of mining interact. Because modeling is so closely connected with data preparation, the chapter introduced various types of models commonly produced by miners.

The key point is that data mining does not exist independently of the business problems that it needs to solve. Data mining exists to serve needs, in general the needs of a business user. The first thing to focus on is the business problem—what is the real problem, what does success look like? When that is established, then and only then is it time to select data and tools appropriate for the job.

Supplemental Material

A Continuously Learning Model Application

A major credit card issuer in the United States wanted to try innovative and more effective

approaches for a solicitation program aimed at acquiring new customers. Several routes to market, including telemarketing and “take-one” programs, were used as well as direct mail. Additional marketing promotions, solicitations, and offer structures were included in the overall program, including balance transfer, affinity group marketing, and a variety of rates and payment terms. Most of this detail will not be included in this description. Although continuously learning models were used in all aspects of the solicitation, for clarity we will focus only on the direct mail portion. Before describing how a continuously learning model was built for this customer, it will be helpful to have a brief introduction to the significant considerations in credit card solicitations.

Typically, a marketing solicitation program without continuously learning techniques involves selecting the mailing list, producing the mail piece, making a bulk mailing, receiving the responses, entering the responses, and approving or declining each responding applicant. In essence, the bulk of the mailing goes out all together as far as that is possible. In practice the mailing is usually of such quantity that it is often spread over a number of days.

Since the mailing goes out, in effect, all at once, the response quickly builds to a maximum and then gradually tails off to a trickle. The sudden influx of responses necessitates hiring temporary staff, and renting office space and equipment for the staff, to cope with the sudden data entry workload. Federal requirements put great pressure on credit card issuers to “decision” an application—that is, to approve or decline the applicant—within a very short time period or face heavy financial penalties. The “decisioning” process involves studying credit references on all applicants, with the reference information almost invariably obtained from outside vendors. However, credit reference information can only legally be obtained for people who are actually offered credit.

Furthermore, the national average response rate for an unsolicited mailing program is well under 3%. The approve/decline rate is difficult to generalize since usually a variety of groups are targeted and the approve/decline rate varies enormously depending on the group actually targeted. It was estimated that at the time of this program it cost about \$140 to acquire a new credit card customer by direct mail.

How the Continuously Learning Model Worked

The initial reaction of the company when approached with a discussion of the possibilities of data exploration was to say that they knew all about data mining, as they had bought a neural network package and one of their business analysts had built a model, but it didn’t work in their market. Fortunately, they were persuaded to consider the power of data exploration, not simply to mistake it for a PC-based neural network tool.

For the purposes of this explanation it is not necessary to examine the problem and solution explorations. Suffice it to say that the credit card issuer was interested in reducing

the overall cost of the program, lowering the cost per acquisition, and improving the quality of the applicants and users in ways that are discussed during the course of this example.

An advantage of the traditional system of making a massive bulk mailing is that maximum return is felt shortly after the start date of the program. The bulk of new users who are going to respond typically have their credit cards activated within 60 days of program commencement. As discussed, this requires a considerable investment in temporary staff and facilities. The credit card issuer was willing to forgo this quick return for the proposed system that was designed to produce a steady stream of applicants at a preselected rate. The constancy in application level removed the need for, and cost of, temporary staff and facilities, allowing existing staff to cope with the flow rate. It also removed the sudden pressure on the "decisioning" process and additionally permitted the mailing to be routinized. Over the length of the program these changes alone saved a considerable amount of money for the card issuer.

Although it was conceived as a whole, we will consider the simplified pieces of the continuously learning system as they become relevant. The system as a whole is larger than the model itself since the whole system also includes the environment in which the model operates. The system starts with what was labeled the "slush pile."

As was mentioned above, credit reference information can only be obtained for those people to whom credit is actually offered. In order that certain information can be obtained about people that the credit card company may wish to offer credit to, but has not yet done so, a method of "reservation numbers" was devised. Using this system, most of the information pertaining to a particular candidate (instance) is made available, but not the information about who it is. Thus you might know details such as education level, credit balance, number of children, marital status, and possibly well over 100 or more other demographic and sociographic measures. What you cannot know until an offer is made is name and address information, which makes it impossible to attach a record to an individual. In place of the missing information a unique key was supplied; called the "reservation number," it allowed the credit information vendor to supply the relevant information when an offer of credit was to be made. The important point here is that it was possible to know a great deal about the population of potential new customers, but not to know specifically who they were until an offer was made. The information about the population was truly anonymous.

The slush pile consisted of a large number of records (instances) of credit information identified by reservation number. The pool was maintained with a minimum of about 1,000,000 records. As the records (instances) were "used," that is, they had been selected and a solicitation made to them, the appropriate record was removed from the slush pile and replaced with a new, unsolicited credit record.

The continuously learning solicitation system began with the mailing process. A lot of

preparatory work was needed to build a PIE. A result of the data preparation process, the PIE is a model of the data that allows new data drawn from the same population that was used to create the model to be transformed in “real time” into a form appropriate to be modeled. Its purpose is to convert raw data into the selected form after manual involvement in data preparation has been completed. The PIE for this application was built from credit information similar to, but not used in, this application.

Since there was no history of performance, the initial action was to make a random selection from the slush pile for the initial mailing. However, once responses from the initial mailing were received, it immediately became possible to begin building a model of who was likely to respond to the solicitation.

Information about who had actually responded to the mailing was entered into a table. Using the information about who had been solicited and who had responded, a fully automatic modeling process built models segmenting those attributes most indicative of a response to the mailing. The key features of the continuously learning system were that modeling was fully automatic—*with no human operator involvement*—and that the response was automatically optimized based on feedback information. One of the set points was externally fixed within the system—that is, the number of responses required. Another set point was variable and selected for optimization—that is, response rate as a percentage of solicitations. The environmental parameters under the “control” of this piece of the system included the selection criteria from the slush pile—that is, the characteristics of the person to whom the offer was made. Also included in the environment was which offer to make from the variety available. Thus, optimal cross-sell of additional products was automatically built in. The system, on receiving feedback information about response, used the information to update its model of the current driving factors—that is, what was working best at that instant, incorporating changes produced by competing offers, market dynamics, or social changes in the real world.

Fully automatic selection of the next batch to be mailed was made by the system based on the response model generated from previous mailings. The system automatically adjusted the number of solicitations to be mailed, based upon response levels, so that the right number of responses came back to meet the target selected in the project objectives.

The next stage in the process was that the applications were decisioned. This approve/decline information was entered into the system. The system now had additional factors in its environment to control—targeting not only people who would respond, but also those most likely to be approved. Once again, this model was automatically maintained, without human intervention, by the continuously learning system.

Following this, additional automatic environmental controls were added. The first was added when pattern of use information became available. Many credit card issuers feel a strong preference toward customers who are not “convenience users,” those who pay the balance in full when requested and, thus, never generate revenue for the issuer in the

form of interest payments. Another increase in the quality of the target potential customers resulted—those who would not only respond and be approved, but also would be profitable for the card issuer. Eventually, default and fraud were modeled and added into the selection process.

This is a highly simplified description indeed. However, the system as described consists of four “sensors” feeding into a model continuously learning to recognize particular features in the environment—responsiveness, approval likeliness, convenience user tendency, and proclivity to default/fraud. The “environmental” parameters under the system’s control were the selection criteria for the 160 or so variables of the prospects in the slush pile, plus what products to offer each candidate.

Some particularly notable features of this system were that, for the duration of the program, the internal structures of the various model elements changed—dramatically in some cases. That is to say, the key indicating factors of, for example, who was likely to respond to the solicitation were dramatically different at different times. (A competing offer from another company targeted much of the original population. Any static model would have been defeated. The continuously learning model simply moved its sights and kept right on producing. Some time later the company’s marketers discovered what was going on.)

Clearly, any static model would have lost predictive power very quickly. The “half-life” of a static model, especially as market and economic conditions were changing rapidly, seemed to be about six weeks. While no full analysis of many of the underlying reasons for this shift was made, various economic, political, and social changes were happening during the solicitation period, from such things as the competing offer already mentioned, to a presidential election. cursory examination of the parameter drift in the models indicated that these changes had an impact. In fact, competition from other credit card companies’ solicitation programs, targeting similar demographic and affinity groups, made for the most dramatic changes in the model.

In addition to actively reacting to changing conditions in order to optimize return, various pieces of business intelligence were generated. There were, in fact, a variety of different offers made, such as gold cards, preapproved and non-preapproved cards, interest rates and terms, home equity loans, lines of credit, and so on. Although not specifically requested in the specification of the system, a response surface model built of the response pattern based on the actual offer revealed what it was about different offers that different groups found attractive. This allowed the company’s marketing organization to make adjustments to terms and conditions offered to increase the appeal of the solicitation.

Although this is a very brief summary description of what a continuously learning model looks like in practice, it shows that it has a key place in a data miner’s toolkit. This particular model produced spectacular results. This system was able to achieve, among other things,

response rates peaking over 10% (compared to an industry standard of well under 3%) and a greatly reduced acquisition cost (varying from time to time, of course, but under \$75 at times compared to the client's previous \$140). Additional benefits gained might be described (from the credit card issuer's viewpoint) as higher-quality customers—less likely to be convenience users or to default.