1

Introduction

Data mining has been defined as the search for useful and previously unknown patterns in large datasets. Yet when faced with the task of mining a large dataset, it is not always obvious where to start and how to proceed. The purpose of this book is to introduce a methodology for data mining and to guide you in the application of that methodology using software specifically designed to support the methodology. In this chapter, we provide an overview of the methodology. The chapters that follow add detail to that methodology and contain a sequence of exercises that guide you in its application. The exercises use VisMiner, a powerful visual data mining tool which was designed around the methodology.

Data Mining Objectives

Normally in data mining a mathematical model is constructed for the purpose of **prediction** or **description**. A model can be thought of as a virtual box that accepts a set of inputs, then uses that input to generate output.

Prediction modeling algorithms use selected input attributes and a single selected output attribute from your dataset to build a model. The model, once built, is used to predict an output value based on input attribute values. The dataset used to build the model is assumed to contain historical data from past events in which the values of both the input and output attributes are known. The data mining methodology uses those values to construct a model that best fits the data. The process of model construction is sometimes referred to as **training**. The primary objective of model construction is to use the model for predictions in the future using known input attribute values when the value

Visual Data Mining: The VisMiner Approach, First Edition. Russell K. Anderson.

of the output attribute is not yet known. Prediction models that have a categorical output are known as **classification** models. For example, an insurance company may want to build a classification model to predict if an insurance claim is likely to be fraudulent or legitimate.

Prediction models that have numeric output are called **regression** models. For example, a retailer may use a regression model to predict sales for a proposed new store based on the demographics of the store. The model would be built using data from previously opened stores.

One special type of regression modeling is **forecasting**. Forecasting models use time series data to predict future values. They look at trends and cycles in previous periods in making the predictions for future time periods.

Description models built by data mining algorithms include: **cluster**, **association**, and **sequence** analyses.

Cluster analysis forms groupings of similar observations. The clusterings generated are not normally an end process in data mining. They are frequently used to extract subsets from the dataset to which other data mining methodologies may be applied. Because the behavioral characteristics of sub-populations within a dataset may be so different, it is frequently the case that models built using the subsets are more accurate than those built using the entire dataset. For example, the attitude toward, and use of, mass transit by the urban population is quite different from that of the rural population.

Association analysis looks for sets of items that occur together. Association analysis is also known as market basket analysis due to its application in studies of what consumers buy together. For example, a grocery retailer may find that bread, milk, and eggs are frequently purchased together. Note, however, that this would not be considered a real data mining discovery, since data mining is more concerned with finding the unexpected patterns rather than the expected.

Sequence analysis is similar to association analysis, except that it looks for groupings over time. For example, a women's clothing retailer may find that within two weeks of purchasing a pair of shoes, the customer may return to purchase a handbag. In bioinformatics, DNA studies frequently make use of sequence analysis.

Introduction to VisMiner

VisMiner is a software tool designed to visually support the entire data mining process. It is intended to be used in a course setting both for individual student use and classroom lectures when the processes of data mining are presented. During lectures, students using VisMiner installed on desktop, laptop, tablet computers, and smart phones are able to actively participate with the instructor as datasets are analyzed and the methodology is examined.

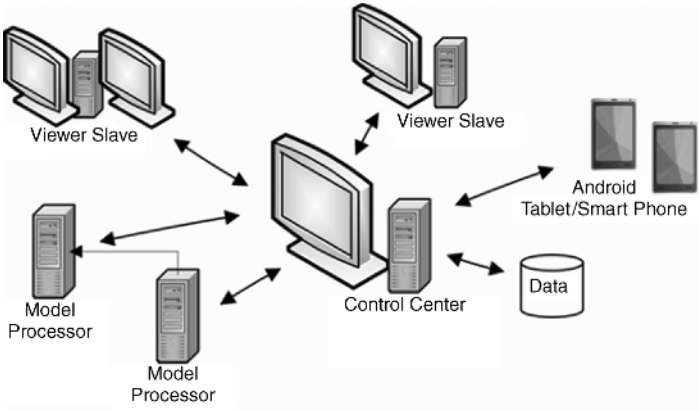

Figure 1.1 VisMiner Architecture

The architecture of VisMiner is represented in Figure 1.1. It consists of four main components:

- the **Control Center**, which manages the datasets, starts and stops the modelers and viewers, and coordinates synchronization between viewers
- **VisSlave** and **ModelSlave** which establish the connections between a slave computer and the Control Center
- the **modelers** that execute the sophisticated data mining algorithms
- the **viewers** that present interactive visualizations of the datasets and the models generated using the datasets.

As evidenced by Figure 1.1, VisMiner may run on one or more computers. The primary computer runs the Control Center. Computers that will present visualizations should run VisSlave; computers that will be used for back-end processing should run ModelSlave. In the full configuration of VisMiner, there should be just one instance of the Control Center executing, and as many instances of VisSlave and ModelSlave as there are computers available for their respective purposes. If there is only one computer, use it to run all three applications.

The Data Mining Process

Successful data mining requires a potentially time-consuming and methodical process. That's why they call it "mining". Gold prospectors don't buy their gear, head out and discover gold on the first day. For them it takes months or even

years of search. The same is true with data mining. It takes work, but hopefully not months or years.

In this book, we present a methodology. VisMiner is designed to support and streamline the methodology. The methodology consists of four steps:

- **Initial data exploration** – conduct an initial exploration of the data to gain an overall understanding of its size and characteristics, looking for clues that should be explored in more depth.
- **Dataset preparation** – prepare the data for analysis.
- **Algorithm application** – select and apply data mining algorithms to the dataset.
- **Results evaluation** – evaluate the results of the algorithm applications, assessing the "goodness of fit" of the data to the algorithm results and assessing the nature and strengths of inputs to the algorithm outputs.

These steps are not necessarily sequential in nature, but should be considered as an iterative process progressing towards the end result – a complete and thorough analysis. Some of the steps may even be completed in parallel. This is true for "Initial data exploration" and "dataset preparation". In VisMiner for example, interactive visualizations designed primarily for the initial data exploration also support some of the dataset preparation tasks.

In the sections that follow, we elaborate on the tasks to be completed in each of the steps. In later chapters, problems and exercises are presented that guide you through completion of these tasks using VisMiner. Throughout the book, reference is made back to the task descriptions introduced here. It is suggested that as you work through the problems and exercises, you refer back to this list. Use it as a reminder of what has and has not been completed.

Initial data exploration

The primary objective of initial data exploration is to help the analyst gain an overall understanding of the dataset. This includes:

- **Dataset size and format** – Determine the number of observations in the dataset. How much space does it occupy? In what format is it stored? Possible formats include tab or comma delimited text files, fixed field text files, tables in a relational database, and pages in a spreadsheet. Since most datasets stored in a relational database are encoded in the proprietary format of the database management system used to store the data, check that you have access to software that can retrieve and manipulate the content. Look also at the number of tables containing data of interest. If found in multiple tables, determine how they are linked and how they might be joined.

- **Attribute enumeration** – Begin by browsing the list of attributes contained in the dataset and the corresponding types of each attribute. Understand what each attribute represents or measures and the units in which it is encoded. Look for identifier or key attributes – those that uniquely identify observations in the dataset.
- **Attribute distributions** – For numeric types, determine the range of values in the dataset, then look at the shape and symmetry or skew of the distribution. Does it appear to approximate a normal distribution or some other distribution? For nominal (categorical) data, look at the number of unique values (categories) and the proportion of observations belonging to each category. For example, suppose that you have an attribute called *CustomerType*. The first thing that you want to determine is the number of different *CustomerTypes* in the dataset and the proportions of each.
- **Identification of sub-populations** – Look for attribute distributions that are **multimodal** – that is distributions that have multiple peaks. When you see such distributions, it indicates that the observations in the dataset are drawn from multiple sub-populations with potentially different distributions. It is possible that these sub-populations could generate very different models when submitted in isolation to the data mining algorithms as compared to the model generated when submitting the entire dataset. For example, in some situations the purchasing behavior of risk-taking individuals may be quite different from those that are risk averse.
- **Pattern search** – Look for potentially interesting and significant relationships (or patterns) between attributes. If your data mining objective is the generation of a prediction model, focus on relationships between your selected output attribute and attributes that may be considered for input. Note the type of the relationship – linear or non-linear, direct or inverse. Ask the question, "Does this relationship seem reasonable?" Also look at relationships between potential input attributes. If they are highly correlated, then you probably want to eliminate all but one as you conduct in-depth analyses.

Dataset preparation

The objective of dataset preparation is to change or morph the dataset into a form that allows the dataset to be submitted to a data mining algorithm for analysis. Tasks include:

- **Observation reduction** – Frequently there is no need to analyze the full dataset when a subset is sufficient. There are three reasons to reduce the observation count in a dataset.
    - The amount of time required to process the full dataset may be too computationally intensive. An organization's actual production database

may have millions of observations (transactions). Mining of the entire dataset may be too time-consuming for processing using some of the available algorithms.

- The dataset may contain sub-populations which are better mined independently. At times, patterns emerge in sub-populations that don't exist in the dataset as a whole.
- The level of detail (**granularity**) of the data may be more than is necessary for the planned analysis. For example, a sales dataset may have information on each individual sale made by an enterprise. However, for mining purposes, sales information summarized at the customer level or other geographic level, such as zip code, may be all that is necessary.

Observation reduction can be accomplished in three ways:

- extraction of sub-populations
- sampling
- observation aggregation.

- **Dimension reduction** – As dictated by the "**curse of dimensionality**", data becomes more **sparse** or spread out as the number of dimensions in a dataset increases. This leads to a need for larger and larger sample sizes to adequately fill the data space as the number of dimensions (attributes) increases. In general, when applying a dataset to a data mining algorithm, the fewer the dimensions the more likely the results are to be statistically valid. However, it is not advisable to eliminate attributes that may contribute to good model predictions or explanations. There is a trade-off that must be balanced.

  To reduce the dimensionality of a dataset, you may selectively remove attributes or arithmetically combine attributes.

  Attributes should be removed if they are not likely to be relevant to an intended analysis or if they are redundant. An example of an irrelevant attribute would be an observation identifier or key field. One would not expect a customer number, for example, to contribute anything to the understanding of a customer's purchase behavior. An example of a redundant attribute would be a measure that is recorded in multiple units. For example, a person's weight may be recorded in pounds and kilograms – both are not needed.

  You may also arithmetically combine attributes with a formula. For example, in a "homes for sale" dataset containing *price* and *area* (square feet) attributes, you might derive a new attribute "price per square foot" by dividing *price* by *area*, then eliminating the *price* and *area* attributes.

  A related methodology for combining attributes to reduce the number of dimensions is **principal component analysis**. It is a mathematical

procedure in which a set of correlated attributes are transformed into a potentially smaller and uncorrelated set.

- **Outlier detection** – Outliers are individual observations whose values are very different from the other observations in the dataset. Normally, outliers are erroneous data resulting from problems during data capture, data entry, or data encoding and should be removed from the dataset as they will distort results. In some cases, they may be valid data. In these cases, after verifying the validity of the data, you may want to investigate further – looking for factors contributing to their uniqueness.
- **Dataset restructuring** – Many of the data mining algorithms require a single tabular input dataset. A common source of mining data is transactional data recorded in a relational database, with data of interest spread across multiple tables. Before processing using the mining algorithms, the data must be joined in a single table. In other instances, the data may come from multiple sources such as marketing research studies and government datasets. Again, before processing the data will need to be merged into a single set of tabular data.
- **Balancing of attribute values** – Frequently a classification problem attempts to identify factors leading to a targeted anomalous result. Yet, precisely because the result is anomalous, there will be few observations in the dataset containing that result if the observations are drawn from the general population. Consequently, the classification modelers used will fail to focus on factors indicating the anomalous result, because there just are not enough in the dataset to derive the factors. To get around this problem, the ratio of anomalous results to other results in the dataset needs to be increased. A simple way to accomplish this is to first select all observations in the dataset with the targeted result, then combine those observations with an equal number of randomly selected observations, thus yielding a 50/50 ratio.
- **Separation into training and validation datasets** – A common problem in data mining is that the output model of a data mining algorithm is **overfit** with respect to the **training data** – the data used to build the model. When this happens, the model appears to perform well when applied to the training data, but performs poorly when applied to a different set of data. When this happens we say that the model does not **generalize** well. To detect and assess the level of overfit or lack of generalizability, before a data mining algorithm is applied to a dataset, the data is randomly split into training data and **validation data**. The training data is used to build the model and the validation data is then applied to the newly built model to determine if the model generalizes to data not seen at the time of model construction.

- **Missing values** – Frequently, datasets are missing values for one or more attributes in an observation. The values may be missing because at the time the data was captured they were unknown or, for a given observation, the values do not exist.

  Since many data mining algorithms do not work well, if at all, when there are missing values in the dataset, it is important that they be handled before presentation to the algorithm. There are three generally deployed ways to deal with missing values:

  - Eliminate all observations from the dataset containing missing values.
  - Provide a default value for any attributes in which there may be missing values. The default value for example, may be the most frequently occurring value in an attribute of discrete types, or the average value for a numeric attribute.
  - Estimate using other attribute values of the observation.

Algorithm selection and application

Once the dataset has been properly prepared and an initial exploration has been completed, you are ready to apply a data mining algorithm to the dataset. The choice of which algorithm to apply depends on the objective of your data mining task and the types of data available. If the objective is classification, then you will want to choose one or more of the available classification modelers. If you are predicting numeric output, then you will choose from an available regression modeler.

Among modelers of a given type, you may not have a prior expectation as to which modeler will generate the best model. In that case, you may want to apply the data to multiple modelers, evaluate, then choose the model that performs best for the dataset.

At the time of model building you will need to have decided which attributes to use as input attributes and which, if building a prediction model, is the output attribute. (Cluster, association, and sequence analyses do not have an output attribute.) The choice of input attributes should be guided by relationships uncovered during the initial exploration.

Once you have selected your modelers and attributes, and taken all necessary steps to prepare the dataset, then apply that dataset to the modelers – let them do their number crunching.

Model evaluation

After the modeler has finished its work and a model has been generated, evaluate that model. There are two tasks to be accomplished during this phase.

- **Model performance** – Evaluate how well the model performs. If it is a prediction model, how well does it predict? You can answer that question by either comparing the model's performance to the performance of a random guess, or by building multiple models and comparing the performance of each.
- **Model understanding** – Gain an understanding of how the model works. Again, if it is a prediction model, you should ask questions such as: "What input attributes contribute most to the prediction?" and "What is the nature of that contribution?" For some attributes you may find a direct relationship, while in others you may see an inverse relationship. Some of the relationships may be linear, while others are non-linear. In addition, the contributions of one input may vary depending on the level of a second input. This is referred to as variable interaction and is important to detect and understand.

Summary

In this chapter an overview of a methodology for conducting a data mining analysis was presented. The methodology consists of four steps: initial data exploration, dataset preparation, data mining modeler application, and model evaluation. In the chapters that follow, readers will be guided through application of the methodology using a visual tool for data mining – VisMiner. Chapter 2 uses the visualizations and features of VisMiner to conduct the initial exploration and do some dataset preparation. Chapter 3 introduces additional features of VisMiner for dataset preparation not covered in Chapter 2. Chapters 4 through 7 introduce the data mining methodologies available in VisMiner, with tutorials covering their application and evaluation using VisMiner visualizations.