Preface

The convergence of computing and communication has produced a society that feeds on information. Yet most of the information is in its raw form: data. If *data* is characterized as recorded facts, then *information* is the set of patterns, or expectations, that underlie the data. There is a huge amount of information locked up in databases—information that is potentially important but has not yet been discovered or articulated. Our mission is to bring it forth.

Data mining is the extraction of implicit, previously unknown, and potentially useful information from data. The idea is to build computer programs that sift through databases automatically, seeking regularities or patterns. Strong patterns, if found, will likely generalize to make accurate predictions on future data. Of course, there will be problems. Many patterns will be banal and uninteresting. Others will be spurious, contingent on accidental coincidences in the particular dataset used. And real data is imperfect: Some parts will be garbled, some missing. Anything that is discovered will be inexact: There will be exceptions to every rule and cases not covered by any rule. Algorithms need to be robust enough to cope with imperfect data and to extract regularities that are inexact but useful.

Machine learning provides the technical basis of data mining. It is used to extract information from the raw data in databases—information that is expressed in a comprehensible form and can be used for a variety of purposes. The process is one of abstraction: taking the data, warts and all, and inferring whatever structure underlies it. This book is about the tools and techniques of machine learning that are used in practical data mining for finding, and describing, structural patterns in data.

As with any burgeoning new technology that enjoys intense commercial attention, the use of data mining is surrounded by a great deal of hype in the technical—and sometimes the popular—press. Exaggerated reports appear of the secrets that can be uncovered by setting learning algorithms loose on oceans of data. But there is no magic in machine learning, no hidden power, no alchemy. Instead, there is an identifiable body of simple and practical techniques that can often extract useful information from raw data. This book describes these techniques and shows how they work.

We interpret machine learning as the acquisition of structural descriptions from examples. The kind of descriptions that are found can be used for prediction, explanation, and understanding. Some data mining applications focus on prediction: They forecast what will happen in new situations from data that describe what happened in the past, often by guessing the classification of new examples. But we are equally—perhaps more—interested in applications where the result of "learning" is an actual description of a structure that can be used to classify examples. This structural description supports explanation and understanding as well as prediction. In our experience, insights gained by the user are of most interest in the majority of practical data mining applications; indeed, this is one of machine learning's major advantages over classical statistical modeling.

The book explains a wide variety of machine learning methods. Some are pedagogically motivated: simple schemes that are designed to explain clearly how the basic ideas work. Others are practical: real systems that are used in applications today. Many are contemporary and have been developed only in the last few years.

A comprehensive software resource has been created to illustrate the ideas in this book. Called the Waikato Environment for Knowledge Analysis, or Weka[1] for short, it is available as Java source code at *www.cs.waikato.ac.nz/ml/weka*. It is a full, industrial-strength implementation of essentially all the techniques that are covered in this book. It includes illustrative code and working implementations of machine learning methods. It offers clean, spare implementations of the simplest techniques, designed to aid understanding of the mechanisms involved. It also provides a workbench that includes full, working, state-of-the-art implementations of many popular learning schemes that can be used for practical data mining or for research. Finally, it contains a framework, in the form of a Java class library, that supports applications that use embedded machine learning and even the implementation of new learning schemes.

The objective of this book is to introduce the tools and techniques for machine learning that are used in data mining. After reading it, you will understand what these techniques are and appreciate their strengths and applicability. If you wish to experiment with your own data, you will be able to do this easily with the Weka software.

The book spans the gulf between the intensely practical approach taken by trade books that provide case studies on data mining and the more theoretical, principle-driven exposition found in current textbooks on machine learning. (A brief description of these books appears in the Further Reading section at the end of Chapter 1.) This gulf is rather wide. To apply machine learning techniques productively, you need to understand something about how they work; this is not a technology that you can apply blindly and expect to get good results. Different problems yield to different techniques, but it is rarely obvious which techniques are suitable for a given situation: You need to know something about the range of possible solutions. And we cover an extremely wide range of techniques. We can do this because, unlike many trade books, this volume does not promote any particular commercial software or approach. We include a large number of examples, but they use illustrative datasets that are small enough to allow you to follow what is going on. Real datasets are far too large to show this (and in any case are usually company confidential). Our datasets are chosen not to illustrate actual large-scale practical problems but to help you understand what the different techniques do, how they work, and what their range of application is.

The book is aimed at the technically aware general reader who is interested in the principles and ideas underlying the current practice of data mining. It will also

[1]Found only on the islands of New Zealand, the weka (pronounced to rhyme with "Mecca") is a flightless bird with an inquisitive nature.

be of interest to information professionals who need to become acquainted with this new technology, and to all those who wish to gain a detailed technical understanding of what machine learning involves. It is written for an eclectic audience of information systems practitioners, programmers, consultants, developers, information technology managers, specification writers, patent examiners, and curious lay people, as well as students and professors, who need an easy-to-read book with lots of illustrations that describes what the major machine learning techniques are, what they do, how they are used, and how they work. It is practically oriented, with a strong "how to" flavor, and includes algorithms, code, and implementations. All those involved in practical data mining will benefit directly from the techniques described. The book is aimed at people who want to cut through to the reality that underlies the hype about machine learning and who seek a practical, nonacademic, unpretentious approach. We have avoided requiring any specific theoretical or mathematical knowledge, except in some sections that are marked by a box around the text. These contain optional material, often for the more technically or theoretically inclined reader, and may be skipped without loss of continuity.

The book is organized in layers that make the ideas accessible to readers who are interested in grasping the basics, as well as accessible to those who would like more depth of treatment, along with full details on the techniques covered. We believe that consumers of machine learning need to have some idea of how the algorithms they use work. It is often observed that data models are only as good as the person who interprets them, and that person needs to know something about how the models are produced to appreciate the strengths, and limitations, of the technology. However, it is not necessary for all users to have a deep understanding of the finer details of the algorithms.

We address this situation by describing machine learning methods at successive levels of detail. The book is divided into three parts. Part I is an introduction to data mining. The reader will learn the basic ideas, the topmost level, by reading the first three chapters. Chapter 1 describes, through examples, what machine learning is and where it can be used; it also provides actual practical applications. Chapters 2 and 3 cover the different kinds of input and output, or *knowledge representation*, that are involved—different kinds of output dictate different styles of algorithm. Chapter 4 describes the basic methods of machine learning, simplified to make them easy to comprehend. Here, the principles involved are conveyed in a variety of algorithms without getting involved in intricate details or tricky implementation issues. To make progress in the application of machine learning techniques to particular data mining problems, it is essential to be able to measure how well you are doing. Chapter 5, which can be read out of sequence, equips the reader to evaluate the results that are obtained from machine learning, addressing the sometimes complex issues involved in performance evaluation.

Part II introduces advanced techniques of data mining. At the lowest and most detailed level, Chapter 6 exposes in naked detail the nitty-gritty issues of implementing a spectrum of machine learning algorithms, including the complexities that are necessary for them to work well in practice (but omitting the heavy mathematical

machinery that is required for a few of the algorithms). Although many readers may want to ignore such detailed information, it is at this level that the full, working, tested Java implementations of machine learning schemes are written. Chapter 7 describes practical topics involved with engineering the input and output to machine learning—for example, selecting and discretizing attributes—while Chapter 8 covers techniques of "ensemble learning," which combine the output from different learning techniques. Chapter 9 looks to the future.

The book describes most methods used in practical machine learning. However, it does not cover reinforcement learning because that is rarely applied in practical data mining; nor does it cover genetic algorithm approache, because these are really an optimization technique, or relational learning and inductive logic programming because they are not very commonly used in mainstream data mining applications.

Part III describes the Weka data mining workbench, which provides implementations of almost all of the ideas described in Parts I and II. We have done this in order to clearly separate conceptual material from the practical aspects of how to use Weka. At the end of each chapter in Parts I and II are pointers to related Weka algorithms in Part III. You can ignore these, or look at them as you go along, or skip directly to Part III if you are in a hurry to get on with analyzing your data and don't want to be bothered with the technical details of how the algorithms work.

Java has been chosen for the implementations of machine learning techniques that accompany this book because, as an object-oriented programming language, it allows a uniform interface to learning schemes and methods for pre- and postprocessing. We chose it over other object-oriented languages because programs written in Java can be run on almost any computer without having to be recompiled, having to go through complicated installation procedures, or—worst of all—having to change the code itself. A Java program is compiled into byte-code that can be executed on any computer equipped with an appropriate interpreter. This interpreter is called the *Java virtual machine*. Java virtual machines—and, for that matter, Java compilers—are freely available for all important platforms.

Of all programming languages that are widely supported, standardized, and extensively documented, Java seems to be the best choice for the purpose of this book. However, executing a Java program is slower than running a corresponding program written in languages like C or C++ because the virtual machine has to translate the byte-code into machine code before it can be executed. This penalty used to be quite severe, but Java implementations have improved enormously over the past two decades, and in our experience it is now less than a factor of two if the virtual machine uses a *just-in-time compiler*. Instead of translating each byte-code individually, a just-in-time compiler translates whole chunks of byte-code into machine code, thereby achieving significant speedup. However, if this is still too slow for your application, there are compilers that translate Java programs directly into machine code, bypassing the byte-code step. Of course, this code cannot be executed on other platforms, thereby sacrificing one of Java's most important advantages.

UPDATED AND REVISED CONTENT

We finished writing the first edition of this book in 1999, the second edition in early 2005, and now, in 2011, we are just polishing this third edition. How things have changed over the past decade! While the basic core of material remains the same, we have made the most opportunities to both update it and to add new material. As a result the book has close to doubled in size to reflect the changes that have taken place. Of course, there have also been errors to fix, errors that we had accumulated in our publicly available errata file (available through the book's home page at *http://www.cs.waikato.ac.nz/ml/weka/book.html*).

Second Edition

The major change in the second edition of the book was a separate part at the end that included all the material on the Weka machine learning workbench. This allowed the main part of the book to stand alone, independent of the workbench, which we have continued in this third edition. At that time, Weka, a widely used and popular feature of the first edition, had just acquired a radical new look in the form of an interactive graphical user interface—or, rather, three separate interactive interfaces—which made it far easier to use. The primary one is the Explorer interface, which gives access to all of Weka's facilities using menu selection and form filling. The others are the Knowledge Flow interface, which allows you to design configurations for streamed data processing, and the Experimenter interface, with which you set up automated experiments that run selected machine learning algorithms with different parameter settings on a corpus of datasets, collect performance statistics, and perform significance tests on the results. These interfaces lower the bar for becoming a practicing data miner, and the second edition included a full description of how to use them.

It also contained much new material that we briefly mention here. We extended the sections on rule learning and cost-sensitive evaluation. Bowing to popular demand, we added information on neural networks: the perceptron and the closely related Winnow algorithm, and the multilayer perceptron and the backpropagation algorithm. Logistic regression was also included. We described how to implement nonlinear decision boundaries using both the kernel perceptron and radial basis function networks, and also included support vector machines for regression. We incorporated a new section on Bayesian networks, again in response to readers' requests and Weka's new capabilities in this regard, with a description of how to learn classifiers based on these networks and how to implement them efficiently using AD-trees.

The previous five years (1999–2004) had seen great interest in data mining for text, and this was reflected in the introduction of string attributes in Weka, multinomial Bayes for document classification, and text transformations. We also described efficient data structures for searching the instance space: *k*D-trees and ball trees for finding nearest neighbors efficiently and for accelerating distance-based clustering. We described new attribute selection schemes, such as race search and the use of

support vector machines, and new methods for combining models such as additive regression, additive logistic regression, logistic model trees, and option trees. We also covered recent developments in using unlabeled data to improve classification, including the co-training and co-EM methods.

Third Edition

For this third edition, we thoroughly edited the second edition and brought it up to date, including a great many new methods and algorithms. Our basic philosophy has been to bring the book and the Weka software even closer together. Weka now includes implementations of almost all the ideas described in Parts I and II, and vice versa—pretty well everything currently in Weka is covered in this book. We have also included far more references to the literature: This third edition practically triples the number of references that were in the first edition.

As well as becoming far easier to use, Weka has grown beyond recognition over the last decade, and has matured enormously in its data mining capabilities. It now incorporates an unparalleled range of machine learning algorithms and related techniques. This growth has been partly stimulated by recent developments in the field and partly user-led and demand-driven. This puts us in a position where we know a lot about what actual users of data mining want, and we have capitalized on this experience when deciding what to include in this book.

As noted earlier, this new edition is split into three parts, which has involved a certain amount of reorganization. More important, a lot of new material has been added. Here are a few of the highlights.

Chapter 1 includes a section on web mining, and, under ethics, a discussion of how individuals can often be "reidentified" from supposedly anonymized data. A major addition describes techniques for multi-instance learning, in two new sections: basic methods in Section 4.9 and more advanced algorithms in Section 6.10. Chapter 5 contains new material on interactive cost–benefit analysis. There have been a great number of other additions to Chapter 6: cost-complexity pruning, advanced association-rule algorithms that use extended prefix trees to store a compressed version of the dataset in main memory, kernel ridge regression, stochastic gradient descent, and hierarchical clustering methods. The old chapter Engineering the Input and Output has been split into two: Chapter 7 on data transformations (which mostly concern the input) and Chapter 8 on ensemble learning (the output). To the former we have added information on partial least-squares regression, reservoir sampling, one-class learning, decomposing multiclass classification problems into ensembles of nested dichotomies, and calibrating class probabilities. To the latter we have added new material on randomization versus bagging and rotation forests. New sections on data stream learning and web mining have been added to the last chapter of Part II.

Part III, on the Weka data mining workbench, contains a lot of new information. Weka includes many new filters, machine learning algorithms, and attribute selection algorithms, and many new components such as converters for different file formats and parameter optimization algorithms. Indeed, within each of these categories Weka

contains around 50% more algorithms than in the version described in the second edition of this book. All these are documented here. In response to popular demand we have given substantially more detail about the output of the different classifiers and what it all means. One important change is the inclusion of a brand new Chapter 17 that gives several tutorial exercises for the Weka Explorer interface (some of them quite challenging), which we advise new users to work though to get an idea of what Weka can do.