



Ian H. Witten & Eibe Frank

DATA MINING

Practical Machine Learning Tools and Techniques

SECOND EDITION



Trend Micro, Inc.
EX1014-1

Publisher: Diane Cerra
 Publishing Services Manager: Simon Crump
 Project Manager: Brandy Lilly
 Editorial Assistant: Asma Stephan
 Cover Design: Yvo Riezebos Design
 Cover Image: Getty Images
 Composition: SNP Best-set Typesetter Ltd., Hong Kong
 Technical Illustration: Dartmouth Publishing, Inc.
 Copyeditor: Graphic World Inc.
 Proofreader: Graphic World Inc.
 Indexer: Graphic World Inc.
 Interior printer: The Maple-Vail Book Manufacturing Group
 Cover printer: Phoenix Color Corp

Morgan Kaufmann Publishers is an imprint of Elsevier.
 500 Sansome Street, Suite 400, San Francisco, CA 94111

This book is printed on acid-free paper.

© 2005 by Elsevier Inc. All rights reserved.

Designations used by companies to distinguish their products are often claimed as trademarks or registered trademarks. In all instances in which Morgan Kaufmann Publishers is aware of a claim, the product names appear in initial capital or all capital letters. Readers, however, should contact the appropriate companies for more complete information regarding trademarks and registration.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopying, scanning, or otherwise—without prior written permission of the publisher.

Permissions may be sought directly from Elsevier's Science & Technology Rights Department in Oxford, UK: phone: (+44) 1865 843830, fax: (+44) 1865 853333, e-mail: permissions@elsevier.com.uk. You may also complete your request on-line via the Elsevier homepage (<http://elsevier.com>) by selecting "Customer Support" and then "Obtaining Permissions."

Library of Congress Cataloging-in-Publication Data

Witten, I. H. (Ian H.)

Data mining : practical machine learning tools and techniques / Ian H. Witten, Eibe Frank. – 2nd ed.

p. cm. – (Morgan Kaufmann series in data management systems)

Includes bibliographical references and index.

ISBN-13: 978-0-12-088407-0 ISBN-10: 0-12-088407-0

1. Data mining. I. Frank, Eibe. II. Title. III. Series.

QA76.9.D343W58 2005
 006.3—dc22

2005043385

ISBN-13: 978-0-12-088407-0
 ISBN-10: 0-12-088407-0

For information on all Morgan Kaufmann publications,
 visit our Web site at www.mkp.com or www.books.elsevier.com

Printed in the United States of America
 06 07 08 09 5 4 3

Working together to grow
 libraries in developing countries

www.elsevier.com | www.bookaid.org | www.sabre.org

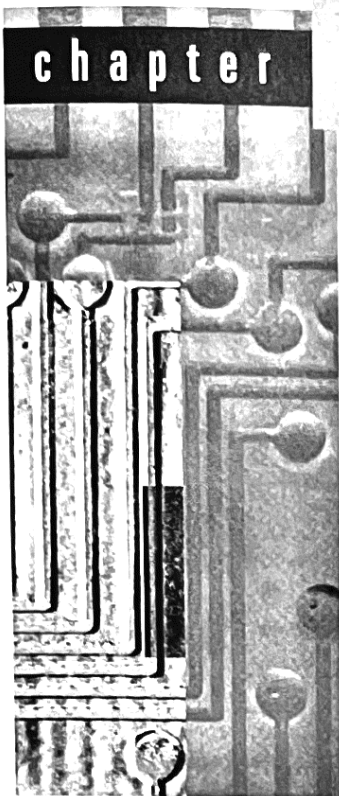
ELSEVIER

BOOK AID
 International

Sabre Foundation

Trend Micro, Inc.
 EX1014-2

What's It All About?



Human *in vitro* fertilization involves collecting several eggs from a woman's ovaries, which, after fertilization with partner or donor sperm, produce several embryos. Some of these are selected and transferred to the woman's uterus. The problem is to select the "best" embryos to use—the ones that are most likely to survive. Selection is based on around 60 recorded features of the embryos—characterizing their morphology, oocyte, follicle, and the sperm sample. The number of features is sufficiently large that it is difficult for an embryologist to assess them all simultaneously and correlate historical data with the crucial outcome of whether that embryo did or did not result in a live child. In a research project in England, machine learning is being investigated as a technique for making the selection, using as training data historical records of embryos and their outcome.

Every year, dairy farmers in New Zealand have to make a tough business decision: which cows to retain in their herd and which to sell off to an abattoir. Typically, one-fifth of the cows in a dairy herd are culled each year near the end of the milking season as feed reserves dwindle. Each cow's breeding and milk pro-

duction history influences this decision. Other factors include age (a cow is nearing the end of its productive life at 8 years), health problems, history of difficult calving, undesirable temperament traits (kicking or jumping fences), and not being in calf for the following season. About 700 attributes for each of several million cows have been recorded over the years. Machine learning is being investigated as a way of ascertaining what factors are taken into account by successful farmers—not to automate the decision but to propagate their skills and experience to others.

Life and death. From Europe to the antipodes. Family and business. Machine learning is a burgeoning new technology for mining knowledge from data, a technology that a lot of people are starting to take seriously.

1.1 Data mining and machine learning

We are overwhelmed with data. The amount of data in the world, in our lives, seems to go on and on increasing—and there's no end in sight. Omnipresent personal computers make it too easy to save things that previously we would have trashed. Inexpensive multigigabyte disks make it too easy to postpone decisions about what to do with all this stuff—we simply buy another disk and keep it all. Ubiquitous electronics record our decisions, our choices in the supermarket, our financial habits, our comings and goings. We swipe our way through the world, every swipe a record in a database. The World Wide Web overwhelms us with information; meanwhile, every choice we make is recorded. And all these are just personal choices: they have countless counterparts in the world of commerce and industry. We would all testify to the growing gap between the *generation* of data and our *understanding* of it. As the volume of data increases, inexorably, the proportion of it that people understand decreases, alarmingly. Lying hidden in all this data is information, potentially useful information, that is rarely made explicit or taken advantage of.

This book is about looking for patterns in data. There is nothing new about this. People have been seeking patterns in data since human life began. Hunters seek patterns in animal migration behavior, farmers seek patterns in crop growth, politicians seek patterns in voter opinion, and lovers seek patterns in their partners' responses. A scientist's job (like a baby's) is to make sense of data, to discover the patterns that govern how the physical world works and encapsulate them in theories that can be used for predicting what will happen in new situations. The entrepreneur's job is to identify opportunities, that is, patterns in behavior that can be turned into a profitable business, and exploit them.

In *data mining*, the data is stored electronically and the search is automated—or at least augmented—by computer. Even this is not particularly new. Economists, statisticians, forecasters, and communication engineers have long worked

with the idea that patterns in data can be sought automatically, identified, validated, and used for prediction. What is new is the staggering increase in opportunities for finding patterns in data. The unbridled growth of databases in recent years, databases on such everyday activities as customer choices, brings data mining to the forefront of new business technologies. It has been estimated that the amount of data stored in the world's databases doubles every 20 months, and although it would surely be difficult to justify this figure in any quantitative sense, we can all relate to the pace of growth qualitatively. As the flood of data swells and machines that can undertake the searching become commonplace, the opportunities for data mining increase. As the world grows in complexity, overwhelming us with the data it generates, data mining becomes our only hope for elucidating the patterns that underlie it. Intelligently analyzed data is a valuable resource. It can lead to new insights and, in commercial settings, to competitive advantages.

Data mining is about solving problems by analyzing data already present in databases. Suppose, to take a well-worn example, the problem is fickle customer loyalty in a highly competitive marketplace. A database of customer choices, along with customer profiles, holds the key to this problem. Patterns of behavior of former customers can be analyzed to identify distinguishing characteristics of those likely to switch products and those likely to remain loyal. Once such characteristics are found, they can be put to work to identify present customers who are likely to jump ship. This group can be targeted for special treatment, treatment too costly to apply to the customer base as a whole. More positively, the same techniques can be used to identify customers who might be attracted to another service the enterprise provides, one they are not presently enjoying, to target them for special offers that promote this service. In today's highly competitive, customer-centered, service-oriented economy, data is the raw material that fuels business growth—if only it can be mined.

Data mining is defined as the process of discovering patterns in data. The process must be automatic or (more usually) semiautomatic. The patterns discovered must be meaningful in that they lead to some advantage, usually an economic advantage. The data is invariably present in substantial quantities.

How are the patterns expressed? Useful patterns allow us to make nontrivial predictions on new data. There are two extremes for the expression of a pattern: as a black box whose innards are effectively incomprehensible and as a transparent box whose construction reveals the structure of the pattern. Both, we are assuming, make good predictions. The difference is whether or not the patterns that are mined are represented in terms of a structure that can be examined, reasoned about, and used to inform future decisions. Such patterns we call *structural* because they capture the decision structure in an explicit way. In other words, they help to explain something about the data.

Now, finally, we can say what this book is about. It is about techniques for finding and describing structural patterns in data. Most of the techniques that we cover have developed within a field known as *machine learning*. But first let us look at what structural patterns are.

Describing structural patterns

What is meant by *structural patterns*? How do you describe them? And what form does the input take? We will answer these questions by way of illustration rather than by attempting formal, and ultimately sterile, definitions. There will be plenty of examples later in this chapter, but let's examine one right now to get a feeling for what we're talking about.

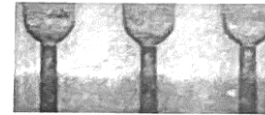
Look at the contact lens data in Table 1.1. This gives the conditions under which an optician might want to prescribe soft contact lenses, hard contact lenses, or no contact lenses at all; we will say more about what the individual

Table 1.1 **The contact lens data.**

Age	Spectacle prescription	Astigmatism	Tear production rate	Recommended lenses
young	myope	no	reduced	none
young	myope	no	normal	soft
young	myope	yes	reduced	none
young	myope	yes	normal	hard
young	hypermetrope	no	reduced	none
young	hypermetrope	no	normal	soft
young	hypermetrope	yes	reduced	none
young	hypermetrope	yes	normal	hard
pre-presbyopic	myope	no	reduced	none
pre-presbyopic	myope	no	normal	soft
pre-presbyopic	myope	yes	reduced	none
pre-presbyopic	myope	yes	normal	hard
pre-presbyopic	hypermetrope	no	reduced	none
pre-presbyopic	hypermetrope	no	normal	soft
pre-presbyopic	hypermetrope	yes	reduced	none
pre-presbyopic	hypermetrope	yes	normal	none
presbyopic	myope	no	reduced	none
presbyopic	myope	no	normal	none
presbyopic	myope	yes	reduced	none
presbyopic	myope	yes	normal	hard
presbyopic	hypermetrope	no	reduced	none
presbyopic	hypermetrope	no	normal	soft
presbyopic	hypermetrope	yes	reduced	none
presbyopic	hypermetrope	yes	normal	none

Credibility:

Evaluating What's Been Learned



Evaluation is the key to making real progress in data mining. There are lots of ways of inferring structure from data: we have encountered many already and will see further refinements, and new methods, in the next chapter. But to determine which ones to use on a particular problem we need systematic ways to evaluate how different methods work and to compare one with another. Evaluation is not as simple as it might appear at first sight.

What's the problem? We have the training set; surely we can just look at how well different methods do on that. Well, no: as we will see very shortly, performance on the training set is definitely not a good indicator of performance on an independent test set. We need ways of predicting performance bounds in practice, based on experiments with whatever data can be obtained.

When a vast supply of data is available, this is no problem: just make a model based on a large training set, and try it out on another large test set. But although data mining sometimes involves “big data”—particularly in marketing, sales, and customer support applications—it is often the case that data, quality data, is scarce. The oil slicks mentioned in Chapter 1 (pages 23–24) had to be detected

and marked manually—a skilled and labor-intensive process—before being used as training data. Even in the credit card application (pages 22–23), there turned out to be only 1000 training examples of the appropriate type. The electricity supply data (pages 24–25) went back 15 years, 5000 days—but only 15 Christmas Days and Thanksgivings, and just 4 February 29s and presidential elections. The electromechanical diagnosis application (pages 25–26) was able to capitalize on 20 years of recorded experience, but this yielded only 300 usable examples of faults. Marketing and sales applications (pages 26–28) certainly involve big data, but many others do not: training data frequently relies on specialist human expertise—and that is always in short supply.

The question of predicting performance based on limited data is an interesting, and still controversial, one. We will encounter many different techniques, of which one—repeated cross-validation—is gaining ascendance and is probably the evaluation method of choice in most practical limited-data situations. Comparing the performance of different machine learning methods on a given problem is another matter that is not so easy as it sounds: to be sure that apparent differences are not caused by chance effects, statistical tests are needed. So far we have tacitly assumed that what is being predicted is the ability to classify test instances accurately; however, some situations involve predicting the class probabilities rather than the classes themselves, and others involve predicting numeric rather than nominal values. Different methods are needed in each case. Then we look at the question of cost. In most practical data mining situations the cost of a misclassification error depends on the type of error it is—whether, for example, a positive example was erroneously classified as negative or vice versa. When doing data mining, and evaluating its performance, it is often essential to take these costs into account. Fortunately, there are simple techniques to make most learning schemes cost sensitive without grappling with the internals of the algorithm. Finally, the whole notion of evaluation has fascinating philosophical connections. For 2000 years philosophers have debated the question of how to evaluate scientific theories, and the issues are brought into sharp focus by data mining because what is extracted is essentially a “theory” of the data.

5.1 Training and testing

For classification problems, it is natural to measure a classifier’s performance in terms of the *error rate*. The classifier predicts the class of each instance: if it is correct, that is counted as a *success*; if not, it is an *error*. The error rate is just the proportion of errors made over a whole set of instances, and it measures the overall performance of the classifier.

Of course, what we are interested in is the likely future performance on new data, not the past performance on old data. We already know the classifications

of each instance in the training set, which after all is why we can use it for training. We are not generally interested in learning about those classifications—although we might be if our purpose is data cleansing rather than prediction. So the question is, is the error rate on old data likely to be a good indicator of the error rate on new data? The answer is a resounding no—not if the old data was used during the learning process to train the classifier.

This is a surprising fact, and a very important one. Error rate on the training set is *not* likely to be a good indicator of future performance. Why? Because the classifier has been learned from the very same training data, any estimate of performance based on that data will be optimistic, and may be hopelessly optimistic.

We have already seen an example of this in the labor relations dataset. Figure 1.3(b) was generated directly from the training data, and Figure 1.3(a) was obtained from it by a process of pruning. The former is likely to be more accurate on the data that was used to train the classifier but will probably perform less well on independent test data because it is overfitted to the training data. The first tree will look good according to the error rate on the training data, better than the second tree. But this does not reflect how they will perform on independent test data.

The error rate on the training data is called the *resubstitution error*, because it is calculated by resubstituting the training instances into a classifier that was constructed from them. Although it is not a reliable predictor of the true error rate on new data, it is nevertheless often useful to know.

To predict the performance of a classifier on new data, we need to assess its error rate on a dataset that played no part in the formation of the classifier. This independent dataset is called the *test set*. We assume that both the training data and the test data are representative samples of the underlying problem.

In some cases the test data might be distinct in nature from the training data. Consider, for example, the credit risk problem from Section 1.3. Suppose the bank had training data from branches in New York City and Florida and wanted to know how well a classifier trained on one of these datasets would perform in a new branch in Nebraska. It should probably use the Florida data as test data to evaluate the New York-trained classifier and the New York data to evaluate the Florida-trained classifier. If the datasets were amalgamated before training, performance on the test data would probably not be a good indicator of performance on future data in a completely different state.

It is important that the test data was not used *in any way* to create the classifier. For example, some learning methods involve two stages, one to come up with a basic structure and the second to optimize parameters involved in that structure, and separate sets of data may be needed in the two stages. Or you might try out several learning schemes on the training data and then evaluate them—on a fresh dataset, of course—to see which one works best. But none of

this data may be used to determine an estimate of the future error rate. In such situations people often talk about three datasets: the *training* data, the *validation* data, and the *test* data. The training data is used by one or more learning methods to come up with classifiers. The validation data is used to optimize parameters of those classifiers, or to select a particular one. Then the test data is used to calculate the error rate of the final, optimized, method. Each of the three sets must be chosen independently: the validation set must be different from the training set to obtain good performance in the optimization or selection stage, and the test set must be different from both to obtain a reliable estimate of the true error rate.

It may be that once the error rate has been determined, the test data is bundled back into the training data to produce a new classifier for actual use. There is nothing wrong with this: it is just a way of maximizing the amount of data used to generate the classifier that will actually be employed in practice. What is important is that error rates are not quoted based on any of this data. Also, once the validation data has been used—maybe to determine the best type of learning scheme to use—then it can be bundled back into the training data to retrain that learning scheme, maximizing the use of data.

If lots of data is available, there is no problem: we take a large sample and use it for training; then another, independent large sample of different data and use it for testing. Provided that both samples are representative, the error rate on the test set will give a true indication of future performance. Generally, the larger the training sample the better the classifier, although the returns begin to diminish once a certain volume of training data is exceeded. And the larger the test sample, the more accurate the error estimate. The accuracy of the error estimate can be quantified statistically, as we will see in the next section.

The real problem occurs when there is not a vast supply of data available. In many situations the training data must be classified manually—and so must the test data, of course, to obtain error estimates. This limits the amount of data that can be used for training, validation, and testing, and the problem becomes how to make the most of a limited dataset. From this dataset, a certain amount is held over for testing—this is called the *holdout* procedure—and the remainder is used for training (and, if necessary, part of that is set aside for validation). There's a dilemma here: to find a good classifier, we want to use as much of the data as possible for training; to obtain a good error estimate, we want to use as much of it as possible for testing. Sections 5.3 and 5.4 review widely used methods for dealing with this dilemma.

5.2 Predicting performance

Suppose we measure the error of a classifier on a test set and obtain a certain numeric error rate—say 25%. Actually, in this section we refer to success rate