Undergraduate Topics in Computer Science

Max Bramer

Principles of Data Mining

Fourth Edition

UTiCS

Springer

Undergraduate Topics in Computer Science

‘Undergraduate Topics in Computer Science’ (UTiCS) delivers high-quality instructional content for undergraduates studying in all areas of computing and information science. From core foundational and theoretical material to final-year topics and applications, UTiCS books take a fresh, concise, and modern approach and are ideal for self-study or for a one- or two-semester course. The texts are all authored by established experts in their fields, reviewed by an international advisory board, and contain numerous examples and problems, many of which include fully worked solutions.

More information about this series at http://www.springer.com/series/7592

Max Bramer

Principles of Data Mining

Fourth Edition

Springer

Prof. Max Bramer
School of Computing
University of Portsmouth
Portsmouth, Hampshire, UK

ISSN 1863-7310 ISSN 2197-1781 (electronic)
Undergraduate Topics in Computer Science
ISBN 978-1-4471-7492-9 ISBN 978-1-4471-7493-6 (eBook)
DOI 10.1007/978-1-4471-7493-6

Printed on acid-free paper

This Springer imprint is published by the registered company Springer-Verlag London Ltd. part of Springer Nature.
The registered company address is: The Campus, 4 Crinan Street, London, N1 9XW, United Kingdom

About This Book

This book is designed to be suitable for an introductory course at either undergraduate or masters level. It can be used as a textbook for a taught unit in a degree programme on potentially any of a wide range of subjects including Computer Science, Business Studies, Marketing, Artificial Intelligence, Bioinformatics and Forensic Science. It is also suitable for use as a self-study book for those in technical or management positions who wish to gain an understanding of the subject that goes beyond the superficial. It goes well beyond the generalities of many introductory books on Data Mining but — unlike many other books — you will not need a degree and/or considerable fluency in Mathematics to understand it.

Mathematics is a language in which it is possible to express very complex and sophisticated ideas. Unfortunately it is a language in which 99% of the human race is not fluent, although many people have some basic knowledge of it from early experiences (not always pleasant ones) at school. The author is a former Mathematician who now prefers to communicate in plain English wherever possible and believes that a good example is worth a hundred mathematical symbols.

One of the author's aims in writing this book has been to eliminate mathematical formalism in the interests of clarity wherever possible. Unfortunately it has not been possible to bury mathematical notation entirely. A 'refresher' of everything you need to know to begin studying the book is given in Appendix A. It should be quite familiar to anyone who has studied Mathematics at school level. Everything else will be explained as we come to it. If you have difficulty following the notation in some places, you can usually safely ignore it, just concentrating on the results and the detailed examples given. For those who would like to pursue the mathematical underpinnings of Data Mining in greater depth, a number of additional texts are listed in Appendix C.

No introductory book on Data Mining can take you to research level in the subject — the days for that have long passed. This book will give you a good grounding in the principal techniques without attempting to show you this year's latest fashions, which in most cases will have been superseded by the time the book gets into your hands. Once you know the basic methods, there are many sources you can use to find the latest developments in the field. Some of these are listed in Appendix C. The other appendices include information about the main datasets used in the examples in the book, many of which are of interest in their own right and are readily available for use in your own projects if you wish, and a glossary of the technical terms used in the book.

Self-assessment Exercises are included for each chapter to enable you to check your understanding. Specimen solutions are given in Appendix E.

Note on the Fourth Edition

Since the first edition there has been a vast and ever-accelerating increase in the volume of data available for data mining. According to IBM (in 2016) 2.5 billion billion bytes of data is produced every day from sensors, mobile devices, online transactions and social networks, with 90 percent of the data in the world having been created in the last two years alone. Today the amount of healthcare data available in the world is estimated as over 2 trillion gigabytes. To reflect the growing popularity of 'deep learning' a new final chapter has been added which gives a detailed introduction to one of the most important types of neural net and shows how it can be applied to classification tasks.

Acknowledgements

I would like to thank my daughter Bryony for drawing many of the more complex diagrams and for general advice on design, Dr. Frederic Stahl for advice on Chapters 21 and 22 and my wife Dawn for her very valuable comments on draft chapters and for preparing the index. The responsibility for any errors that may have crept into the final version remains with me.

Max Bramer
Emeritus Professor of Information Technology
University of Portsmouth, UK
May 2020

Contents

1
Introduction to Data Mining

1.1 The Data Explosion

Modern computer systems are accumulating data at an almost unimaginable rate and from a very wide variety of sources: from point-of-sale machines in the high street to machines logging every bank cash withdrawal and credit card transaction, to Earth observation satellites in space, and with an ever-growing volume of information available from social media and the Internet.

Some examples will serve to give an indication of the volumes of data involved (by the time you read this, some of the numbers will have increased considerably):

- The current NASA Earth observation satellites generate a terabyte (i.e. 10^{12} bytes) of data *every day*. This is more than the total amount of data ever transmitted by all previous observation satellites.
- Biologists are generating around 15 million gigabytes of gene sequence data every year.
- Many companies maintain large Data Warehouses of customer transactions. A fairly small data warehouse might contain more than a hundred million transactions.
- There are vast amounts of data recorded every day on automatic recording devices, such as credit card transaction files and web logs, as well as non-symbolic data such as CCTV recordings.
- There are estimated to be over 1.5 billion websites, some extremely large.
- There are over 2.4 billion active users of Facebook, with an estimated 350 million photographs uploaded every day.

M. Bramer, *Principles of Data Mining*, Undergraduate Topics in Computer Science, DOI 10.1007/978-1-4471-7493-6_1

Alongside advances in storage technology, which increasingly make it possible to store such vast amounts of data at relatively low cost whether in commercial data warehouses, scientific research laboratories or elsewhere, has come a growing realisation that such data contains buried within it knowledge that can be critical to a company's growth or decline, knowledge that could lead to important discoveries in science, knowledge that could enable us accurately to predict the weather and natural disasters, knowledge that could enable us to identify the causes of and possible cures for lethal illnesses, knowledge that could literally mean the difference between life and death. Yet the huge volumes involved mean that most of this data is merely stored — never to be examined in more than the most superficial way, if at all. It has rightly been said that the world is becoming 'data rich but knowledge poor'.

As well as all the stored data, data streams of over a million records a day, potentially continuing forever, are now commonplace.

Machine learning technology, some of it very long established, has the potential to solve the problem of the tidal wave of data that is flooding around organisations, governments and individuals.

1.2 Knowledge Discovery

Knowledge Discovery has been defined as the 'non-trivial extraction of implicit, previously unknown and potentially useful information from data'. It is a process of which data mining forms just one part, albeit a central one.

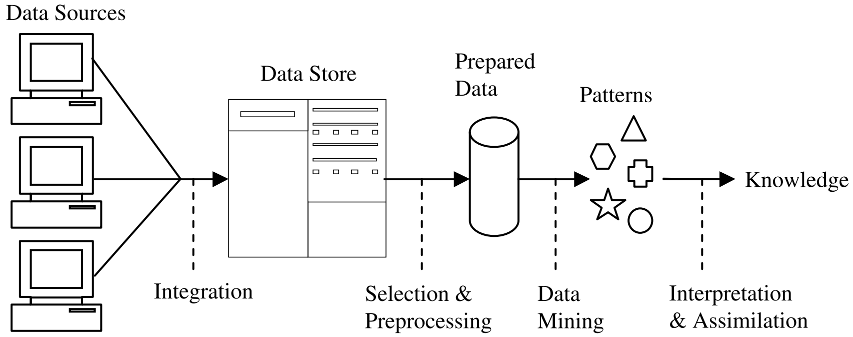

Figure 1.1 The Knowledge Discovery Process

Figure 1.1 shows a slightly idealised version of the complete knowledge discovery process.

Data comes in, possibly from many sources. It is integrated and placed in some common data store. Part of it is then taken and pre-processed into a standard format. This ‘prepared data’ is then passed to a data mining algorithm which produces an output in the form of rules or some other kind of ‘patterns’. These are then interpreted to give — and this is the Holy Grail for knowledge discovery — new and potentially useful knowledge.

This brief description makes it clear that although the data mining algorithms, which are the principal subject of this book, are central to knowledge discovery they are not the whole story. The pre-processing of the data and the interpretation (as opposed to the blind use) of the results are both of great importance. They are skilled tasks that are far more of an art (or a skill learnt from experience) than an exact science. Although they will both be touched on in this book, the algorithms of the data mining stage of knowledge discovery will be its prime concern.

1.3 Applications of Data Mining

There is a rapidly growing body of successful applications in a wide range of areas as diverse as:

- analysing satellite imagery
- analysis of organic compounds
- automatic abstracting
- bioinformatics
- credit card fraud detection
- criminal investigation
- customer relationship management
- electric load prediction
- financial forecasting
- fraud detection
- healthcare
- market basket analysis
- medical diagnosis
- predicting share of television audiences
- product design
- real estate valuation
- targeted marketing
- text summarisation
- thermal power plant optimisation
- toxic hazard analysis
- weather forecasting

and many more.

Some examples of applications (potential or actual) are:

- a supermarket chain mines its customer transactions data to optimise targeting of high value customers
- a credit card company can use its data warehouse of customer transactions for fraud detection
- a major hotel chain can use survey databases to identify attributes of a 'high-value' prospect
- predicting the probability of default for consumer loan applications by improving the ability to predict bad loans
- reducing fabrication flaws in VLSI chips
- data mining systems can sift through vast quantities of data collected during the semiconductor fabrication process to identify conditions that are causing yield problems
- predicting audience share for television programmes, allowing television executives to arrange show schedules to maximise market share and increase advertising revenues
- predicting the probability that a cancer patient will respond to chemotherapy, thus reducing health-care costs without affecting quality of care
- analysing motion-capture data for elderly people
- trend mining and visualisation in social networks
- analysing data from a face recognition system to locate a suspected criminal in a crowd
- analysing information about a range of drugs and natural compounds to identify significant candidates for new antibiotics
- analysing MRI images to identify possible brain tumours.

Applications can be divided into four main types: classification, numerical prediction, association and clustering. Each of these is explained briefly below. However first we need to distinguish between two types of data.

1.4 Labelled and Unlabelled Data

In general we have a dataset of examples (called *instances*), each of which comprises the values of a number of variables, which in data mining are often called *attributes*. There are two types of data, which are treated in radically different ways.

For the first type there is a specially designated attribute and the aim is to use the data given to predict the value of that attribute for instances that have not yet been seen. Data of this kind is called *labelled*. Data mining using labelled data is known as *supervised learning*. If the designated attribute is *categorical*, i.e. it must take one of a number of distinct values such as 'very good', 'good' or 'poor', or (in an object recognition application) 'car', 'bicycle', 'person', 'bus' or 'taxi' the task is called *classification*. If the designated attribute is numerical, e.g. the expected sale price of a house or the opening price of a share on tomorrow's stock market, the task is called *regression*.

Data that does not have any specially designated attribute is called *unlabelled*. Data mining of unlabelled data is known as *unsupervised learning*. Here the aim is simply to extract the most information we can from the data available.

1.5 Supervised Learning: Classification

Classification is one of the most common applications for data mining. It corresponds to a task that occurs frequently in everyday life. For example, a hospital may want to classify medical patients into those who are at high, medium or low risk of acquiring a certain illness, an opinion polling company may wish to classify people interviewed into those who are likely to vote for each of a number of political parties or are undecided, or we may wish to classify a student project as distinction, merit, pass or fail.

This example shows a typical situation (Figure 1.2). We have a dataset in the form of a table containing students' grades on five subjects (the values of attributes SoftEng, ARIN, HCI, CSA and Project) and their overall degree classifications. The row of dots indicates that a number of rows have been omitted in the interests of simplicity. We want to find some way of predicting the classification for other students given only their grade 'profiles'.

There are several ways we can do this, including the following.

Nearest Neighbour Matching. This method relies on identifying (say) the five examples that are 'closest' in some sense to an unclassified one. If the five 'nearest neighbours' have grades Second, First, Second, Second and Second we might reasonably conclude that the new instance should be classified as 'Second'.

Classification Rules. We look for rules that we can use to predict the classification of an unseen instance, for example:

SoftEng	**ARIN**	**HCI**	**CSA**	**Project**	**Class**
A	B	A	B	B	Second
A	B	B	B	B	Second
B	A	A	B	A	Second
A	A	A	A	B	First
A	A	B	B	A	First
B	A	A	B	B	Second
....	....	....	....	....	....
A	A	B	A	B	First

Figure 1.2 Degree Classification Data

IF SoftEng = A AND Project = A THEN Class = First
IF SoftEng = A AND Project = B AND ARIN = B THEN Class = Second
IF SoftEng = B THEN Class = Second

Classification Tree. One way of generating classification rules is via an intermediate tree-like structure called a *classification tree* or a *decision tree.*

Figure 1.3 shows a possible decision tree corresponding to the degree classification data.

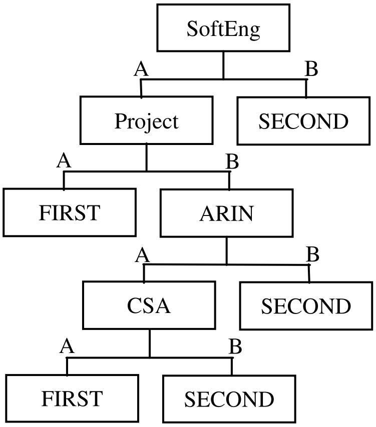

Figure 1.3 Decision Tree for Degree Classification Data

1.6 Supervised Learning: Numerical Prediction

Classification is one form of prediction, where the value to be predicted is a label. Numerical prediction (often called *regression*) is another. In this case we wish to predict a numerical value, such as a company's profits or a share price.

A very popular way of doing this is to use a *Neural Network* as shown in Figure 1.4 (often called by the simplified name *Neural Net*).

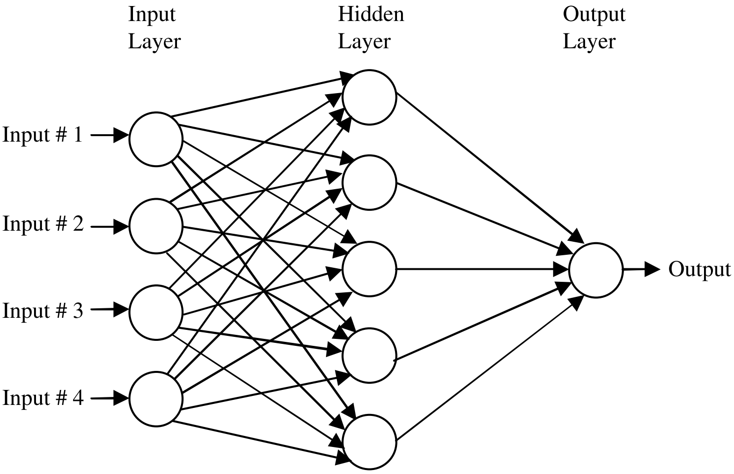

Figure 1.4 A Neural Network

This is a complex modelling technique based on a model of a human neuron. A neural net is given a set of inputs and is used to predict one or more outputs.

One of the most widely used types of neural network is discussed in Chapter 23. However the focus is primarily on classification rather than numerical prediction.

1.7 Unsupervised Learning: Association Rules

Sometimes we wish to use a training set to find any relationship that exists amongst the values of variables, generally in the form of rules known as *association rules*. There are many possible association rules derivable from any given dataset, most of them of little or no value, so it is usual for association rules to be stated with some additional information indicating how reliable they are, for example:

IF variable_1 > 85 and switch_6 = open
THEN variable_23 < 47.5 and switch_8 = closed (probability = 0.8)

A common form of this type of application is called 'market basket analysis'. If we know the purchases made by all the customers at a store for say a week, we may be able to find relationships that will help the store market its products more effectively in the future. For example, the rule

IF cheese AND milk THEN bread (probability = 0.7)

indicates that 70% of the customers who buy cheese and milk also buy bread, so it would be sensible to move the bread closer to the cheese and milk counter, if customer convenience were the prime concern, or to separate them to encourage impulse buying of other products if profit were more important.

1.8 Unsupervised Learning: Clustering

Clustering algorithms examine data to find groups of items that are similar. For example, an insurance company might group customers according to income, age, types of policy purchased or prior claims experience. In a fault diagnosis application, electrical faults might be grouped according to the values of certain key variables (Figure 1.5).

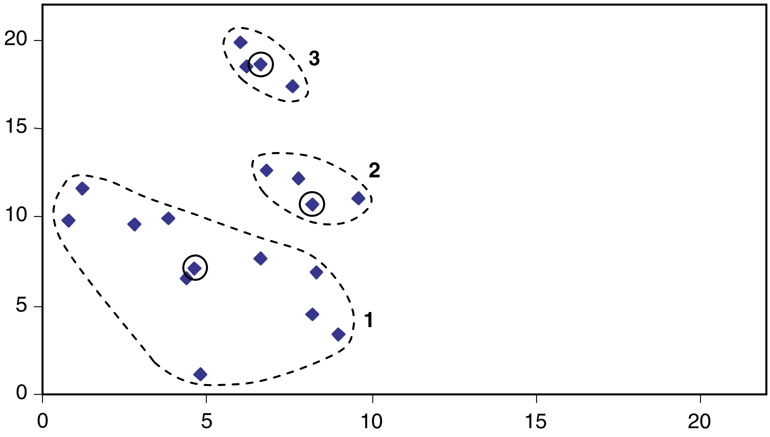

Figure 1.5 Clustering of Data

2
Data for Data Mining

Data for data mining comes in many forms: from computer files typed in by human operators, business information in SQL or some other standard database format, information recorded automatically by equipment such as fault logging devices, to streams of binary data transmitted from satellites. For purposes of data mining (and for the remainder of this book) we will assume that the data takes a particular standard form which is described in the next section. We will look at some of the practical problems of data preparation in Section 2.3.

2.1 Standard Formulation

We will assume that for any data mining application we have a *universe of objects* that are of interest. This rather grandiose term often refers to a collection of people, perhaps all human beings alive or dead, or possibly all the patients at a hospital, but may also be applied to, say, all dogs in England, or to inanimate objects such as all train journeys from London to Birmingham, all the rocks on the moon or all the pages stored in the World Wide Web.

The universe of objects is normally very large and we have only a small part of it. Usually we want to extract information from the data available to us that we hope is applicable to the large volume of data that we have not yet seen.

Each object is described by a number of *variables* that correspond to its properties. In data mining variables are often called *attributes*. We will use both terms in this book.

M. Bramer, *Principles of Data Mining*, Undergraduate Topics in Computer Science, DOI 10.1007/978-1-4471-7493-6_2

The set of variable values corresponding to each of the objects is called a *record* or (more commonly) an *instance*. The complete set of data available to us for an application is called a *dataset*. A dataset is often depicted as a table, with each row representing an instance. Each column contains the value of one of the variables (attributes) for each of the instances. A typical example of a dataset is the 'degrees' data given in the Introduction (Figure 2.1).

SoftEng	**ARIN**	**HCI**	**CSA**	**Project**	**Class**
A	B	A	B	B	Second
A	B	B	B	B	Second
B	A	A	B	A	Second
A	A	A	A	B	First
A	A	B	B	A	First
B	A	A	B	B	Second
....	....	....	....	....	....
A	A	B	A	B	First

Figure 2.1 The Degrees Dataset

This dataset is an example of *labelled* data, where one attribute is given special significance and the aim is to predict its value. In this book we will give this attribute the standard name 'class'. When there is no such significant attribute we call the data *unlabelled*.

2.2 Types of Variable

In general there are many types of variable that can be used to measure the properties of an object. A lack of understanding of the differences between the various types can lead to problems with any form of data analysis. At least six main types of variable can be distinguished.

Nominal Variables

A variable used to put objects into categories, e.g. the name or colour of an object. A nominal variable may be numerical in form, but the numerical values have no mathematical interpretation. For example we might label 10 people as numbers $1, 2, 3, \ldots, 10$, but any arithmetic with such values, e.g. $1 + 2 = 3$

would be meaningless. They are simply labels. A **classification** can be viewed as a nominal variable which has been designated as of particular importance.

Binary Variables

A binary variable is a special case of a nominal variable that takes only two possible values: true or false, 1 or 0 etc.

Ordinal Variables

Ordinal variables are similar to nominal variables, except that an ordinal variable has values that can be arranged in a meaningful order, e.g. small, medium, large.

Integer Variables

Integer variables are ones that take values that are genuine integers, for example 'number of children'. Unlike nominal variables that are numerical in form, arithmetic with integer variables is meaningful (1 child + 2 children = 3 children etc.).

Interval-scaled Variables

Interval-scaled variables are variables that take numerical values which are measured at equal intervals from a zero point or origin. However the origin does not imply a true absence of the measured characteristic. Two well-known examples of interval-scaled variables are the Fahrenheit and Celsius temperature scales. To say that one temperature measured in degrees Celsius is greater than another or greater than a constant value such as 25 is clearly meaningful, but to say that one temperature measured in degrees Celsius is twice another is meaningless. It is true that a temperature of 20 degrees is twice as far from the zero value as 10 degrees, but the zero value has been selected arbitrarily and does not imply 'absence of temperature'. If the temperatures are converted to an equivalent scale, say degrees Fahrenheit, the 'twice' relationship will no longer apply.

Ratio-scaled Variables

Ratio-scaled variables are similar to interval-scaled variables except that the zero point does reflect the absence of the measured characteristic, for example Kelvin temperature and molecular weight. In the former case the zero value corresponds to the lowest possible temperature ‘absolute zero’, so a temperature of 20 degrees Kelvin is twice one of 10 degrees Kelvin. A weight of 10 kg is twice one of 5 kg, a price of 100 dollars is twice a price of 50 dollars etc.

2.2.1 Categorical and Continuous Attributes

Although the distinction between different categories of variable can be important in some cases, many practical data mining systems divide attributes into just two types:

- **categorical** corresponding to nominal, binary and ordinal variables
- **continuous** corresponding to integer, interval-scaled and ratio-scaled variables.

This convention will be followed in this book. For many applications it is helpful to have a third category of attribute, the ‘ignore’ attribute, corresponding to variables that are of no significance for the application, for example the name of a patient in a hospital or the serial number of an instance, but which we do not wish to (or are unable to) delete from the dataset.

It is important to choose methods that are appropriate to the types of variable stored for a particular application. The methods described in this book are applicable to categorical and continuous attributes as defined above. There are other types of variable to which they would not be applicable without modification, for example any variable that is measured on a logarithmic scale. Two examples of logarithmic scales are the Richter scale for measuring earthquakes (an earthquake of magnitude 6 is 10 times more severe than one of magnitude 5, 100 times more severe than one of magnitude 4 etc.) and the Stellar Magnitude Scale for measuring the brightness of stars viewed by an observer on Earth.

2.3 Data Preparation

Although this book is about data mining not data preparation, some general comments about the latter may be helpful.

For many applications the data can simply be extracted from a database in the form described in Section 2.1, perhaps using a standard access method such as ODBC. However, for some applications the hardest task may be to get the data into a standard form in which it can be analysed. For example data values may have to be extracted from textual output generated by a fault logging system or (in a crime analysis application) extracted from transcripts of interviews with witnesses. The amount of effort required to do this may be considerable.

2.3.1 Data Cleaning

Even when the data is in the standard form it cannot be assumed that it is error free. In real-world datasets erroneous values can be recorded for a variety of reasons, including measurement errors, subjective judgements and malfunctioning or misuse of automatic recording equipment.

Erroneous values can be divided into those which are possible values of the attribute and those which are not. Although usage of the term *noise* varies, in this book we will take a *noisy* value to mean one that is valid for the dataset, but is incorrectly recorded. For example the number 69.72 may accidentally be entered as 6.972, or a categorical attribute value such as *brown* may accidentally be recorded as another of the possible values, such as *blue*. Noise of this kind is a perpetual problem with real-world data.

A far smaller problem arises with noisy values that are invalid for the dataset, such as 69.7X for 6.972 or *bbrown* for *brown*. We will consider these to be *invalid values*, not noise. An invalid value can easily be detected and either corrected or rejected.

It is hard to see even very 'obvious' errors in the values of a variable when they are 'buried' amongst say 100,000 other values. In attempting to 'clean up' data it is helpful to have a range of software tools available, especially to give an overall visual impression of the data, when some anomalous values or unexpected concentrations of values may stand out. However, in the absence of special software, even some very basic analysis of the values of variables may be helpful. Simply sorting the values into ascending order (which for fairly small datasets can be accomplished using just a standard spreadsheet) may reveal unexpected results. For example:

- A numerical variable may only take six different values, all widely separated. It would probably be best to treat this as a categorical variable rather than a continuous one.

- All the values of a variable may be identical. The variable should be treated as an 'ignore' attribute.