Geoff Dougherty

Pattern Recognition and Classification

An Introduction

EXTRA
MATERIALS
extras.springer.com

Springer

Pattern Recognition and Classification

Geoff Dougherty

Pattern Recognition and Classification

An Introduction

Springer

Geoff Dougherty
Applied Physics and Medical Imaging
California State University, Channel Islands
Camarillo, CA, USA

Please note that additional material for this book can be downloaded from http://extras.springer.com

ISBN 978-1-4614-5322-2 ISBN 978-1-4614-5323-9 (eBook)
DOI 10.1007/978-1-4614-5323-9
Springer New York Heidelberg Dordrecht London

Library of Congress Control Number: 2012949108

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The use of pattern recognition and classification is fundamental to many of the automated electronic systems in use today. Its applications range from military defense to medical diagnosis, from biometrics to machine learning, from bioinformatics to home entertainment, and more. However, despite the existence of a number of notable books in the field, the subject remains very challenging, especially for the beginner.

We have found that the current textbooks are not completely satisfactory for our students, who are primarily computer science students but also include students from mathematics and physics backgrounds and those from industry. Their mathematical and computer backgrounds are considerably varied, but they all want to understand and absorb the core concepts with a minimal time investment to the point where they can use and adapt them to problems in their own fields. Texts with extensive mathematical or statistical prerequisites were daunting and unappealing to them. Our students complained of "not seeing the wood for the trees," which is rather ironic for textbooks in pattern recognition. It is crucial for newcomers to the field to be introduced to the key concepts at a basic level in an ordered, logical fashion, so that they appreciate the "big picture"; they can then handle progressively more detail, building on prior knowledge, without being overwhelmed. Too often our students have dipped into various textbooks to sample different approaches but have ended up confused by the different terminologies in use.

We have noticed that the majority of our students are very comfortable with and respond well to visual learning, building on their often limited entry knowledge, but focusing on key concepts illustrated by practical examples and exercises. We believe that a more visual presentation and the inclusion of worked examples promote a greater understanding and insight and appeal to a wider audience.

This book began as notes and lecture slides for a senior undergraduate course and a graduate course in Pattern Recognition at California State University Channel Islands (CSUCI). Over time it grew and approached its current form, which has been class tested over several years at CSUCI. It is suitable for a wide range of students at the advanced undergraduate or graduate level. It assumes only a modest background in statistics and mathematics, with the necessary additional material integrated into the text so that the book is essentially self-contained.

The book is suitable both for individual study and for classroom use for students in physics, computer science, computer engineering, electronic engineering, biomedical engineering, and applied mathematics taking senior undergraduate and graduate courses in pattern recognition and machine learning. It presents a comprehensive introduction to the core concepts that must be understood in order to make independent contributions to the field. It is designed to be accessible to newcomers from varied backgrounds, but it will also be useful to researchers and professionals in image and signal processing and analysis, and in computer vision. The goal is to present the fundamental concepts of supervised and unsupervised classification in an informal, rather than axiomatic, treatment so that the reader can quickly acquire the necessary background for applying the concepts to real problems. A final chapter indicates some useful and accessible projects which may be undertaken.

We use ImageJ (http://rsbweb.nih.gov/ij/) and the related distribution, Fiji (http://fiji.sc/wiki/index.php/Fiji) in the early stages of image exploration and analysis, because of its intuitive interface and ease of use. We then tend to move on to MATLAB for its extensive capabilities in manipulating matrices and its image processing and statistics toolboxes. We recommend using an attractive GUI called DipImage (from http://www.diplib.org/download) to avoid much of the command line typing when manipulating images. There are also classification toolboxes available for MATLAB, such as Classification Toolbox (http://www.wiley.com/WileyCDA/Section/id-105036.html) which requires a password obtainable from the associated computer manual) and PRTools (http://www.prtools.org/download.html). We use the Classification Toolbox in Chap. 8 and recommend it highly for its intuitive GUI. Some of our students have explored Weka, a collection of machine learning algorithms for solving data mining problems implemented in Java and open sourced (http://www.cs.waikato.ac.nz/ml/weka/index_downloading.html).

There are a number of additional resources, which can be downloaded from the companion Web site for this book at http://extras.springer.com/, including several useful Excel files and data files. Lecturers who adopt the book can also obtain access to the end-of-chapter exercises.

In spite of our best efforts at proofreading, it is still possible that some typos may have survived. Please notify me if you find any.

I have very much enjoyed writing this book; I hope you enjoy reading it!

Camarillo, CA Geoff Dougherty

Acknowledgments

I would like to thank my colleague Matthew Wiers for many useful conversations and for helping with several of the Excel files bundled with the book. And thanks to all my previous students for their feedback on the courses which eventually led to this book; especially to Brandon Ausmus, Elisabeth Perkins, Michelle Moeller, Charles Walden, Shawn Richardson, and Ray Alfano.

I am grateful to Chris Coughlin at Springer for his support and encouragement throughout the process of writing the book and to various anonymous reviewers who have critiqued the manuscript and trialed it with their classes. Special thanks go to my wife Hajijah and family (Daniel, Adeline, and Nadia) for their patience and support, and to my parents, Maud and Harry (who passed away in 2009), without whom this would never have happened.

Contents

Chapter 1
Introduction

1.1 Overview

Humans are good at recognizing objects (or *patterns*, to use the generic term). We are so good that we take this ability for granted, and find it difficult to analyze the steps in the process. It is generally easy to distinguish the sound of a human voice, from that of a violin; a handwritten numeral "3," from an "8"; and the aroma of a rose, from that of an onion. Every day, we recognize faces around us, but we do it unconsciously and because we cannot explain our expertise, we find it difficult to write a computer program to do the same. Each person's face is a pattern composed of a particular combination of structures (eyes, nose, mouth, ...) located in certain positions on the face. By analyzing sample images of faces, a program should be able to capture the pattern specific to a face and identify (or *recognize*) it as a face (as a member of a category or class we already know); this would be *pattern recognition*. There may be several categories (or classes) and we have to sort (or classify) a particular face into a certain category (or *class*); hence the term *classification*. Note that in *pattern recognition*, the term pattern is interpreted widely and does not necessarily imply a repetition; it is used to include all objects that we might want to classify, e.g., apples (or oranges), speech waveforms, and fingerprints.

A class is a collection of objects that are similar, but not necessarily identical, and which is distinguishable from other classes. Figure 1.1 illustrates the difference between classification where the classes are known beforehand and classification where classes are created after inspecting the objects.

Interest in pattern recognition and classification has grown due to emerging applications, which are not only challenging but also computationally demanding. These applications include:

- Data mining (sifting through a large volume of data to extract a small amount of relevant and useful information, e.g., fraud detection, financial forecasting, and credit scoring)

G. Dougherty, *Pattern Recognition and Classification: An Introduction*,
DOI 10.1007/978-1-4614-5323-9_1, © Springer Science+Business Media New York 2013

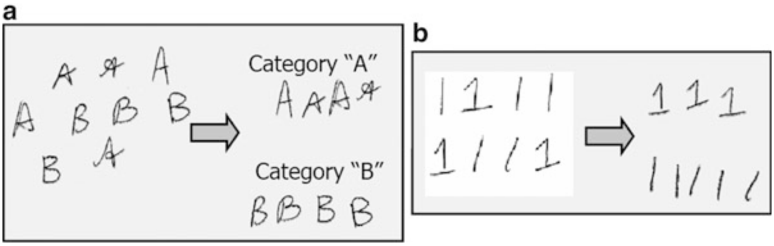

Fig. 1.1 Classification when the classes are (**a**) known and (**b**) unknown beforehand

- Biometrics (personal identification based on physical attributes of the face, iris, fingerprints, etc.)
- Machine vision (e.g., automated visual inspection in an assembly line)
- Character recognition [e.g., automatic mail sorting by zip code, automated check scanners at ATMs (automated teller machines)]
- Document recognition (e.g., recognize whether an e-mail is spam or not, based on the message header and content)
- Computer-aided diagnosis [e.g., helping doctors make diagnostic decisions based on interpreting medical data such as mammographic images, ultrasound images, electrocardiograms (ECGs), and electroencephalograms (EEGs)]
- Medical imaging [e.g., classifying cells as malignant or benign based on the results of magnetic resonance imaging (MRI) scans, or classify different emotional and cognitive states from the images of brain activity in functional MRI]
- Speech recognition (e.g., helping handicapped patients to control machines)
- Bioinformatics (e.g., DNA sequence analysis to detect genes related to particular diseases)
- Remote sensing (e.g., land use and crop yield)
- Astronomy (classifying galaxies based on their shapes; or automated searches such as the Search for Extra-Terrestrial Intelligence (SETI) which analyzes radio telescope data in an attempt to locate signals that might be artificial in origin)

The methods used have been developed in various fields, often independently. In statistics, going from particular observations to general descriptions is called inference, learning [i.e., using example (training) data] is called estimation, and classification is known as discriminant analysis (McLachlan 1992). In engineering, classification is called pattern recognition and the approach is nonparametric and much more empirical (Duda et al. 2001). Other approaches have their origins in machine learning (Alpaydin 2010), artificial intelligence (Russell and Norvig 2002), artificial neural networks (Bishop 2006), and data mining (Han and Kamber 2006). We will incorporate techniques from these different emphases to give a more unified treatment (Fig. 1.2).

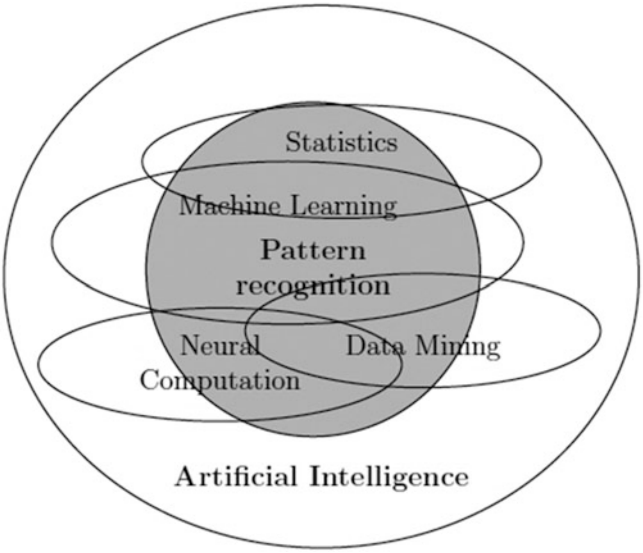

Fig. 1.2 Pattern recognition and related fields

1.2 Classification

Classification is often the final step in a general process (Fig. 1.3). It involves sorting objects into separate classes. In the case of an image, the acquired image is segmented to isolate different objects from each other and from the background, and the different objects are *labeled*. A typical pattern recognition system contains a sensor, a preprocessing mechanism (prior to segmentation), a feature extraction mechanism, a set of examples (*training data*) already classified (post-processing), and a classification algorithm. The *feature extraction* step reduces the data by measuring certain characteristic properties or *features* (such as size, shape, and texture) of the labeled objects. These features (or, more precisely, the values of these features) are then passed to a *classifier* that evaluates the evidence presented and makes a decision regarding the class each object should be assigned, depending on whether the values of its features fall inside or outside the tolerance of that class. This process is used, for example, in classifying lesions as benign or malignant.

The quality of the acquired image depends on the resolution, sensitivity, bandwidth and signal-to-noise ratio of the imaging system. Pre-processing steps such as image enhancement (e.g., brightness adjustment, contrast enhancement, image averaging, frequency domain filtering, edge enhancement) and image restoration (e.g., photometric correction, inverse filtering, Wiener filtering) may be required prior to segmentation, which is often a challenging process. Typically enhancement will precede restoration. Often these are performed sequentially, but more sophisticated tasks will require feedback i.e., advanced processing steps will pass parameters back to preceding steps so that the processing includes a number of iterative loops.

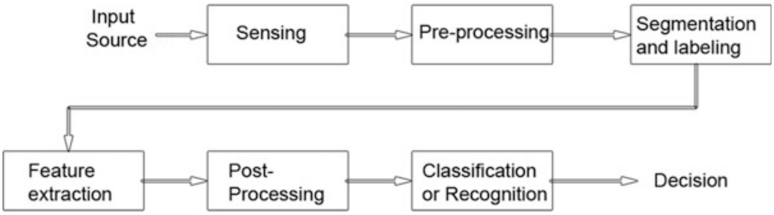

Fig. 1.3 A general classification system

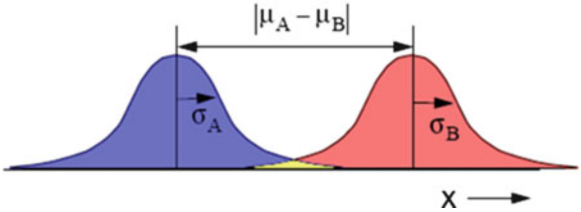

Fig. 1.4 A good feature, *x*, measured for two different classes (*blue* and *red*) should have small intra-class variations and large inter-class variations

The quality of the features is related to their ability to discriminate examples from different classes. Examples from the same class should have similar feature values, while examples from different classes should have different feature values, i.e., good features should have small intra-class variations and large inter-class variations (Fig. 1.4). The measured features can be transformed or mapped into an alternative feature space, to produce better features, before being sent to the classifier.

We have assumed that the features are continuous (i.e., quantitative), but they could be categorical or non-metric (i.e., qualitative) instead, which is often the case in data mining. Categorical features can either be nominal (i.e., unordered, e.g., zip codes, employee ID, gender) or ordinal [i.e., ordered, e.g., street numbers, grades, degree of satisfaction (very bad, bad, OK, good, very good)]. There is some ability to move data from one type to another, e.g., continuous data could be discretized into ordinal data, and ordinal data could be assigned integer numbers (although they would lack many of the properties of real numbers, and should be treated more like symbols). The preferred features are always the most informative (and, therefore in this context, the most discriminating). Given a choice, scientific applications will generally prefer continuous data since more operations can be performed on them (e.g., mean and standard deviation). With categorical data, there may be doubts as to whether all relevant categories have been accounted for, or they may evolve with time.

Humans are adept at recognizing objects within an image, using size, shape, color, and other visual clues. They can do this despite the fact that the objects may appear from different viewpoints and under different lighting conditions, have

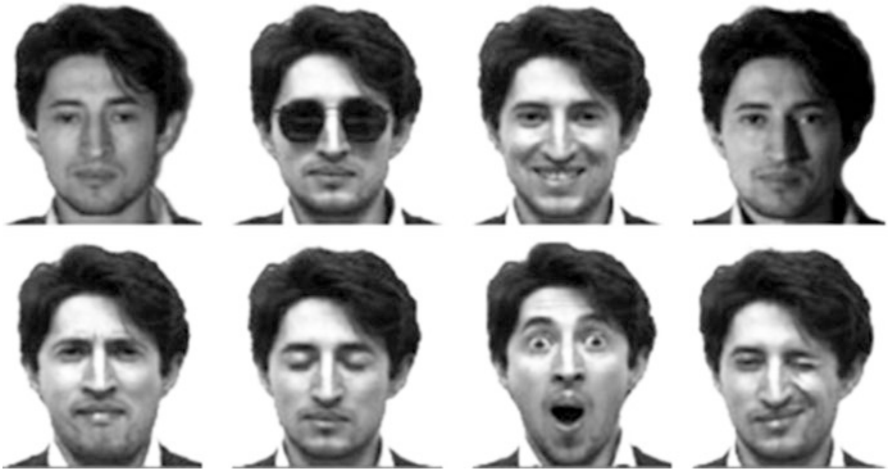

Fig. 1.5 Face recognition needs to be able to handle different expressions, lighting, and occlusions

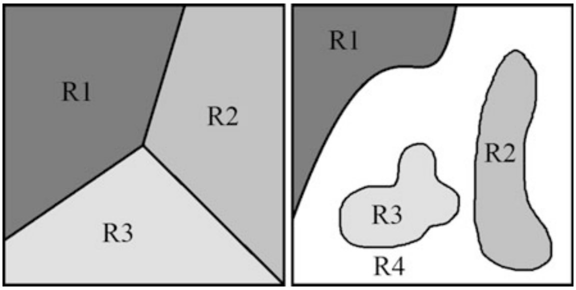

Fig. 1.6 Classes mapped as decision regions, with decision boundaries

different sizes, or be rotated. We can even recognize them when they are partially obstructed from view (Fig. 1.5). These tasks are challenging for machine vision systems in general.

The goal of the classifier is to classify new data (test data) to one of the classes, characterized by a decision region. The borders between decision regions are called decision boundaries (Fig. 1.6).

Classification techniques can be divided into two broad areas: *statistical* or *structural* (or *syntactic*) techniques, with a third area that borrows from both, sometimes called *cognitive methods*, which include *neural networks* and *genetic algorithms*. The first area deals with objects or patterns that have an underlying and quantifiable statistical basis for their generation and are described by quantitative features such as length, area, and texture. The second area deals with objects best described by qualitative features describing structural or syntactic relationships inherent in the object. Statistical classification methods are more popular than structural methods; cognitive methods have gained popularity over the last decade or so. The models are not necessarily independent and hybrid systems involving multiple classifiers are increasingly common (Fu 1983).

1.3 Organization of the Book

In Chap. 2, we will look at the classification process in detail and the different approaches to it, and will look at a few examples of classification tasks. In Chap. 3, we will look at non-metric methods such as decision trees; and in Chap. 4, we will consider probability theory, leading to Bayes' Rule and the roots of statistical pattern recognition. Chapter 5 considers supervised learning—and examples of both parametric and non-parametric learning. We will look at different ways to evaluate the performance of classifiers in Chap. 5. Chapter 6 considers the curse of dimensionality and how to keep the number of features to a useful minimum. Chapter 7 considers unsupervised learning techniques, and Chap. 8 looks at ways to evaluate the performance of the various classifiers. Chapter 9 will consider stochastic methods, and Chap. 10 will discuss some interesting classification problems.

By judiciously avoiding some of the details, the material can be covered in a single semester. Alternatively, fully featured (!!) and with a healthy dose of exercises/applications and some project work, it would form the basis for two semesters of work. The independent reader, on the other hand, can follow the material at his or her own pace and should find sufficient amusement for a few months! Enjoy, and happy studying!

1.4 Exercises

1. List a number of applications of classification, additional to those mentioned in the text.
2. Consider the data of four adults, indicating their weight (actually, their mass) and their health status. Devise a simple classifier that can properly classify all four patterns.

Weight (kg)	Class label
50	Unhealthy
60	Healthy
70	Healthy
80	Unhealthy

How is a fifth adult of weight 76 kg classified using this classifier?

3. Consider the following items bought in a supermarket and some of their characteristics:

Item no.	Cost ($)	Volume (cm^3)	Color	Class label
1	20	6	Blue	Inexpensive
2	50	8	Blue	Inexpensive
3	90	10	Blue	Inexpensive
4	100	20	Red	Expensive
5	160	25	Red	Expensive
6	180	30	Red	Expensive

Which of the three features (cost, volume and color) is the best classifier?

4. Consider the problem of classifying objects into circles and ellipses. How would you classify such objects?

References

Alpaydin, E.: Introduction to Machine learning, 2nd edn. MIT Press, Cambridge (2010)

Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, Oxford (2006)

Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley, New York (2001)

Fu, K.S.: A step towards unification of syntactic and statistical pattern recognition. IEEE Trans. Pattern Anal. Mach. Intell. **5**, 200–205 (1983)

Han, J., Kamber, M.: Data Mining: Concepts and Techniques, 2nd edn. Morgan Kaufmann, San Francisco (2006)

McLachlan, G.J.: Discriminant Analysis and Statistical Pattern Recognition. Wiley, New York (1992)

Russell, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 2nd edn. Prentice Hall, New York (2002)

Chapter 2
Classification

2.1 The Classification Process

A general classification system, without feedback between stages, is shown in Fig. 2.1.

The *sensing/acquisition* stage uses a transducer such as a camera or a microphone. The acquired signal (e.g., an image) must be of sufficient quality that distinguishing "features" can be adequately measured. This will depend on the characteristics of the transducer, but for a camera this would include the following: its resolution, dynamic range, sensitivity, distortion, signal-to-noise ratio, whether focused or not, etc.

Pre-processing is often used to condition the image for segmentation. For example, smoothing of the image (e.g., by convolution with a Gaussian mask) mitigates the confounding effect of noise on segmentation by thresholding (since the random fluctuations comprising noise can result in pixels being shifted across a threshold and being misclassified). Pre-processing with a median mask effectively removes shot (i.e., salt-and-pepper) noise. Removal of a variable background brightness and histogram equalization are often used to ensure even illumination.

Depending on the circumstances, we may have to handle missing data (Batista and Monard 2003), and detect and handle outlier data (Hodge and Austin 2004).

Segmentation partitions an image into regions that are meaningful for a particular task—the *foreground*, comprising the objects of interest, and the *background*, everything else. There are two major approaches—region-based methods, in which similarities are detected, and boundary-based methods, in which discontinuities (edges) are detected and linked to form continuous boundaries around regions.

Region-based methods find connected regions based on some similarity of the pixels within them. The most basic feature in defining the regions is image gray level or brightness, but other features such as color or texture can also be used. However, if we require that the pixels in a region be very similar, we may over-segment the image, and if we allow too much dissimilarity we may merge what should be separate objects. The goal is to find regions that correspond to objects as humans see them, which is not an easy goal.

G. Dougherty, *Pattern Recognition and Classification: An Introduction*,
DOI 10.1007/978-1-4614-5323-9_2, © Springer Science+Business Media New York 2013

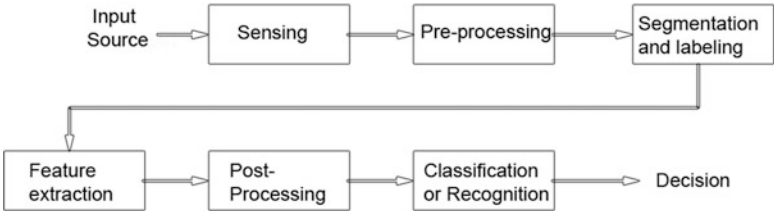

Fig. 2.1 A general classification process

Region-based methods include thresholding [either using a global or a locally adaptive threshold; optimal thresholding (e.g., Otsu, isodata, or maximum entropy thresholding)]. If this results in overlapping objects, thresholding of the distance transform of the image or using the watershed algorithm can help to separate them. Other region-based methods include region growing (a bottom-up approach using "seed" pixels) and split-and-merge (a top-down quadtree-based approach).

Boundary-based methods tend to use either an edge detector (e.g., the Canny detector) and edge linking to link any breaks in the edges, or boundary tracking to form continuous boundaries. Alternatively, an active contour (or snake) can be used; this is a controlled continuity contour which elastically snaps around and encloses a target object by locking on to its edges.

Segmentation provides a simplified, binary image that separates objects of interest (foreground) from the background, while retaining their shape and size for later measurement. The foreground pixels are set to "1" (white), and the background pixels set to "0" (black). It is often desirable to label the objects in the image with discrete numbers. Connected components labeling scans the segmented, binary image and groups its pixels into components based on pixel connectivity, i.e., all pixels in a connected component share similar pixel values and are in some way connected with each other. Once all groups have been determined, each pixel is labeled with a number (1, 2, 3, . . .), according to the component to which it was assigned, and these numbers can be looked up as gray levels or colors for display (Fig. 2.2).

One obvious result of labeling is that the objects in an image can be readily counted. More generally, the labeled binary objects can be used to *mask* the original image to isolate each (grayscale) object but retain its original pixel values so that its properties or features can be measured separately. Masking can be performed in several different ways. The binary mask can be used in an overlay, or alpha channel, in the display hardware to prevent pixels from being displayed. It is also possible to use the mask to modify the stored image. This can be achieved either by multiplying the grayscale image by the binary mask or by bit-wise ANDing the original image with the binary mask. Isolating features, which can then be measured independently, are the basis of *region-of-interest (RoI) processing*.

Post-processing of the segmented image can be used to prepare it for feature extraction. For example, partial objects can be removed from around the periphery of the image (e.g., Fig. 2.2e), disconnected objects can be merged, objects smaller or larger than certain limits can be removed, or holes in the objects or background can be filled by morphological opening or closing.