FUZZY MODELS AND ALGORITHMS FOR PATTERN RECOGNITION AND IMAGE PROCESSING

by

James C. Bezdek
James Keller
Raghu Krisnapuram
Nikhil R. Pal

THE HANDBOOK OF FUZZY SETS SERIES
SERIES EDITORS: Didier Dubois and Henri Prade

FUZZY MODELS AND ALGORITHMS FOR PATTERN RECOGNITION AND IMAGE PROCESSING

James C. Bezdek
University of West Florida

James Keller
University of Missouri

Raghu Krisnapuram
Colorado School of Mines

Nikhil R. Pal
Indian Statistical Institute

Springer Science+Business Media, LLC

Library of Congress Cataloging-in-Publication Data

Fuzzy models and algorithms for pattern recognition and image processing / James C. Bezdek ... [et al.].
p. cm. -- (Handbooks of fuzzy sets series ; FSHS 4)
Includes bibliographical references and index.
ISBN 978-0-387-24515-7 **ISBN 978-0-387-24579-9 (eBook)**
DOI 10.1007/978-0-387-24579-9

1. Optical pattern recognition. 2. Fuzzy algorithms. 3. Cluster analysis. 4. Image processing. 5. Computer vision. I. Bezdek, James C., 1939- . II. Series.
TA1650.F89 1999
006.4'2--dc21 99-27835
CIP

Originally published by Kluwer Academic Publishers in 1999
Softcover reprint of the hardcover 1st edition 1999

Printed on acid-free paper.

Contents

The authors Rather than compile many chapters written by various authors who use different notations and semantic descriptions for the same models, we decided to have a small team of four persons write the entire volume. Each of us assumed the role of lead author for one or more of the chapters, and the other authors acted like consultants to the lead author. Each of us helped the lead author by contributing examples, references, diagrams or text here and there; and we all reviewed the entire volume three times. Whether this approach was successful remains to be seen.

The plan What we tried to do is this: identify the important work that has been done in fuzzy pattern recognition, describe it, analyze it, and illustrate it with examples that an interested reader can follow. As with all projects of this kind, the material inevitably reflects some bias on the part of its authors (after all, the easiest examples to give already live in our own computers). Moreover, this has become an enormous field, and the truth is that it is now far too large for us to even *know about* many important and useful papers that go unrecognized here. We apologize for our bias and our ignorance, and accept any and all blame for errors of fact and/or omission. How current is the material in the book? Knuth (1968) stated that "It is generally very difficult to keep up with a field that is economically profitable, and so it is only natural to expect that many of the techniques described here eventually be superseded by better ones". We cannot say it better.

The numbering system The atomic unit for the numbering system is the chapter. Figures, tables, examples and equations are all numbered consecutively within each chapter. For example, Figure 3.5 is Figure 5 of Chapter 3. The beginning and end of examples are enclosed by goofy looking brackets, like this:

Example 5.4 Did you ever have to finally decide? To pick up on one and let the other one ride, so many changes,.......

The algorithms: art, science and voodoo There are a lot of algorithms in the book. We ran many, but not certainly not all, of the experiments ourselves. We have given pseudo code for quite a few algorithms, and it is really pseudo in the sense that it is a mixture of three or four programming languages and writing styles. Our intent is to maximize clarity and minimize dependence on a particular language, operating system, compiler, host platform, and so on. We hope you can read the pseudo code, and that you can convert it into working programs with a minimum of trouble.

Almost all algorithms have parameters that affect their performance. Science is about quantitative models of our physical world, while art tries to express the qualitative content of our lives. When you read this book you will encounter lots of parameters that are user-defined, together with evasive statements like "pick a value for k that is close to 1", or "don't use high values for m". What do instructions such as these mean? Lots of things: (i) we don't have better advice; (ii) the inventor of the algorithm tried lots of values, and values in the range mentioned produced the best results for her or him; (iii) 0.99 is closer to 1 than 0.95, and 22 is higher than 1.32, you may never know which choice is better, and (unfortunately) this can make all the difference in your application; (iv) sometimes we don't know why things work the way they do, but we should be happy if they work right this time - call it voodoo, or call it luck, but if it works, take it.

Is this cynical? No, it's practical. Science is *NOT* exact, it's a sequence of successively better approximations by models we invent to the physical reality of processes we initiate, observe or control. There's a lot of art in science, and this is nowhere more evident than in pattern recognition, because here, the data always have the last word. We are always at the mercy of an unanticipated situation in the data; unusual structures, missing observations, improbable events that cause outliers, uncertainty about the interactions between variables, useless choices for numerical representation, sensors that don't respect our design goals, computers that lose bits, computer programs that have an undetected flaw, and so on. When you read about and experiment with algorithmic parameters, have an open mind - anything is possible, and usually is.

The data Most of the numerical examples use small data sets that may seem contrived to you, and some of them are. There is much to be said for the pedagogical value of using a few points in the plane when studying and illustrating properties of various models. On the other hand, there are certain risks too. Sometimes conclusions that are legitimate for small, specialized data sets become invalid in the face of large numbers of samples, features and classes. And of course, time and space complexity make their presence felt in very unpredictable ways as problem size grows.

There is another problem with data sets that everyone probably knows about, but that is much harder to detect and document, and that problem goes under the heading of, for example, "*will the real Iris data please stand up?*". Anderson's (1935) Iris data, which we think was first published in Fisher (1936), has become a popular set of labeled data for testing - and especially for comparing - clustering algorithms and classifiers. It is of course entirely appropriate and in the spirit of scientific inquiry to make and publish comparisons of models and their performance on common data sets, and the pattern recognition community has used Iris in perhaps a thousand papers for just this reason - - - or have we?

During the writing of this book we have discovered - perhaps others have known this for a long time, but we didn't - that there are at least two (and hence, probably half a dozen) different, well publicized versions of Iris. Specifically, vector 90, class 2 (Iris Versicolor) in Iris has the coordinates (5.5, 2.5, 4, 1.3) on p. 566, Johnson and Wichern (1992); and has the coordinates (5.5, 2.5, 5, 1.3) on p. 224 in Chien (1978). YIKES !! For the record, we are using the Iris data as published in Fisher (1936) and repeated in Johnson and Wichern (1992). We will use *Iris (?)* when we are not sure what data were used.

What this means is that many of the papers you have come to know and love that compare the performance of this and that using Iris may in fact have examples of algorithms that were executed using different data sets! What to do? Well, there isn't much we can do about this problem. We have checked our own files, and they all contain the data as listed in Fisher (1936) and Johnson and Wichern (1992). That's not too reassuring, but it's the best we can do. We have tried to check which Iris data set was used in the examples of other authors that are discussed in this book, but this is nearly impossible. We do not guarantee that all the results we discuss for "the" Iris data really pertain to the same numerical inputs. Indeed, the "Lena" image is the Iris data of image processing, - after all, the original Lena was a poor quality, 6 bit image, and more recent copies, including the ones we use in this book, come to us with higher resolution. To be sure, there is only one analog Lena (although PLAYBOY ran many), but there are probably many different digital Lenae.

Data get corrupted many ways, and in the electronic age, it should not surprise us to find (if we can) that this is a fairly common event. Perhaps the best solution to this problem would be to establish a central repository for common data sets. This has been tried several times without much success. Out of curiosity, on September 7, 1998 we fetched Iris from the anonymous FTP site "ftp.ics.uci.edu" under the directory "pub/machine-learning-databases", and discovered not one, but *two* errors in it! Specifically, two vectors in Iris Sestosa were wrong: vector 35 in Fisher (1936) is (4.9, 3.1, 1.5, 0.2) but in the machine learning electronic database it had coordinates (4.9, 3.1, 1.5, 0.1); and vector 38 in Fisher is (4.9, 3.6, 1.4, 0.1), but in the electronic database it was (4.9, 3.1, 1.5, 0.1). Finally, we are aware of several papers that used a version of Iris obtained by multiplying every value by 10, so that the data are integers, and the papers involved discuss 10*Iris as if they thought it was Iris. We don't think there is a way to correct all the databases out there which contain similar mistakes (we trust that the machine learning database will be fixed after our alert), but we have included a listing of Iris in Appendix 2 of this book (and, we hope it's right). What all this means for you, the pattern recognition aficionado is this: *pattern recognition* **is** *data, and not all data are created equally, much less replicated faithfully!*

Numerical results We have tried to give you all the information you need to *replicate* the outputs we report in numerical examples. There are a few instances where this was not possible (for example, when an iterative procedure was initialized randomly, or when the results were reported in someone's paper 10 or 15 years ago, or when the authors of a paper we discuss simply could not supply us with more details), and of course it's always possible that the code we ran implemented something other than we thought it did, or it simply had undetected programming errors. Also, we have rounded off or truncated the reported results of many calculations to make tables fit into the format of the book. Let us know if you find substantial differences between outputs you get (or got) and the results we report.

The references More than one reference system is one too many. We chose to reference books and papers by last names and years. As with any system, this one has advantages and disadvantages. Our scheme lets you find a paper quickly if you know the last name of the first author, but causes the problem of appending "a", "b" and so on to names that appear more than once in the same year. There may be a mistake or two, or even $\boldsymbol{O}(n)$ of them. Again, please let us know about it. We have divided the references into two groups: those actually cited in the text, and a second set of references that point to related material that, for one reason or another, just didn't find their way into the text discussion. Many of these uncited papers are excellent - please have a look at them.

The acronyms Acronyms, like the plague, seem to spread unchecked through the technical literature of pattern recognition. We four are responsible for quite a few of them, and so, we can hardly hold this bad habit against others. This book has several hundred acronyms in it, and we know you won't remember what many of them mean for more than a few pages. Consequently, Appendix 1 is a tabulation of the acronyms and abbreviations used in the text.

Acknowledgments The authors wish to acknowledge their gratitude to the following agencies, who graciously supplied partial support as shown during the writing of this book:

J. C. Bezdek :	ONR grant # N00014-96-1-0642 NSF grant # IRI-9003252
J. M. Keller :	ONR grant # N00014-96-1-0439 ARO MURI grant # DAAG55-97-1-0014
R. Krishnapuram:	ONR grant # N00014-96-1-0439 NSFgrant # IRI-9800899

We also want to express our thanks to Andrea Baraldi, Alma Blonda, Larry Hall, Lucy Kuncheva and Thomas Runkler, all of whom were kind enough to review various parts of the manuscript and/or supplied us with computations for several examples that we could not find in the literature, and whose helpful comments save us at least a few embarrassments.

The quotes Everyone nowadays seems to have a pithy quote at each chapter head, at the end of each email, on their web page, tattooed on their leg, etc., so we wanted to have some too. Rather than choose one quote for the book that all of us could live with (quite a range of tastes exists amongst us four), we decided to each supply one quote for this preface. We give the quotes here, but don't identify who contributed each one. That will be revealed in the pages of this volume - but only to those readers alert enough to *recognize the patterns*.

"What use are all these high-flying vaunts of yours?
O King of Birds! You will be the world's laughing stock.
What a marvel would it be if the hare
were to void turd the size of elephant dung!"

Vishnu Sharma, in Panchatantra, circa AD 400

"Only the mediocre are always at their best"

Blue Wave, circa 1995

"All uncertainty is fruitful ... so long as it is accompanied by the wish to understand"

Antonio Machado, Juan de Mairena, 1943

"You gotta pay your dues if you want to play the blues, and you know that don't come easy"

Ringo Starr, circa 1973

You may think you know which of us contributed each of these quotes - but you might be surprised. Life is full of surprises, and so is this book. We hope you enjoy both.

Jim Bezdek
Jim Keller
Rags Krishnapuram
Nik Pal