Machine Learning in Python®

Machine Learning in Python®

Essential Techniques for Predictive Analysis

Michael Bowles

WILEY

Machine Learning in Python®: Essential Techniques for Predictive Analysis

Published by
John Wiley & Sons, Inc.
10475 Crosspoint Boulevard
Indianapolis, IN 46256
www.wiley.com

ISBN: 978-1-118-96174-2
ISBN: 978-1-118-96176-6 (ebk)
ISBN: 978-1-118-96175-9 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Library of Congress Control Number: 2015930541

To my children, Scott, Seth, and Cayley. Their blossoming lives and selves bring me more joy than anything else in this world.

To my close friends David and Ron for their selfless generosity and steadfast friendship.

To my friends and colleagues at Hacker Dojo in Mountain View, California, for their technical challenges and repartee.

To my climbing partners. One of them, Katherine, says climbing partners make the best friends because "they see you paralyzed with fear, offer encouragement to overcome it, and celebrate when you do."

About the Author

Dr. Michael Bowles (Mike) holds Bachelor's and Master's degrees in Mechanical Engineering, an Sc.D. in Instrumentation, and an MBA. He has worked in academia, technology, and business. Mike currently works with startup companies where machine learning is integral to success. He serves variously as part of the management team, a consultant, or advisor. He also teaches machine learning courses at Hacker Dojo, a co-working space and startup incubator in Mountain View, California.

Mike was born in Oklahoma and earned his Bachelor's and Master's degrees there. Then after a stint in Southeast Asia, Mike went to Cambridge for his Sc.D. and then held the C. Stark Draper Chair at MIT after graduation. Mike left Boston to work on communications satellites at Hughes Aircraft company in Southern California, and then after completing an MBA at UCLA moved to the San Francisco Bay Area to take roles as founder and CEO of two successful venture-backed startups.

Mike remains actively involved in technical and startup-related work. Recent projects include the use of machine learning in automated trading, predicting biological outcomes on the basis of genetic information, natural language processing for website optimization, predicting patient outcomes from demographic and lab data, and due diligence work on companies in the machine learning and big data arenas. Mike can be reached through `www.mbowles.com`.

About the Technical Editor

Daniel Posner holds Bachelor's and Master's degrees in Economics and is completing a Ph.D. in Biostatistics at Boston University. He has provided statistical consultation for pharmaceutical and biotech firms as well as for researchers at the Palo Alto VA hospital.

Daniel has collaborated with the author extensively on topics covered in this book. In the past, they have written grant proposals to develop web-scale gradient boosting algorithms. Most recently, they worked together on a consulting contract involving random forests and spline basis expansions to identify key variables in drug trial outcomes and to sharpen predictions in order to reduce the required trial populations.

Credits

Executive Editor
Robert Elliott

Project Editor
Jennifer Lynn

Technical Editor
Daniel Posner

Production Editor
Dassi Zeidel

Copy Editor
Keith Cline

Manager of Content Development & Assembly
Mary Beth Wakefield

Marketing Director
David Mayhew

Marketing Manager
Carrie Sherrill

Professional Technology & Strategy Director
Barry Pruett

Business Manager
Amy Knies

Associate Publisher
Jim Minatel

Project Coordinator, Cover
Brent Savage

Proofreader
Word One New York

Indexer
Johnna VanHoose Dinse

Cover Designer
Wiley

Acknowledgments

I'd like to acknowledge the splendid support that people at Wiley have offered during the course of writing this book. It began with Robert Elliot, the acquisitions editor, who first contacted me about writing a book; he was very easy to work with. It continued with Jennifer Lynn, who has done the editing on the book. She's been very responsive to questions and very patiently kept me on schedule during the writing. I thank you both.

I also want to acknowledge the enormous comfort that comes from having such a sharp, thorough statistician and programmer as Daniel Posner doing the technical editing on the book. Thank you for that and thanks also for the fun and interesting discussions on machine learning, statistics, and algorithms. I don't know anyone else who'll get as deep as fast.

Contents at a Glance

Contents

Introduction

Extracting actionable information from data is changing the fabric of modern business in ways that directly affect programmers. One way is the demand for new programming skills. Market analysts predict demand for people with advanced statistics and machine learning skills will exceed supply by 140,000 to 190,000 by 2018. That means good salaries and a wide choice of interesting projects for those who have the requisite skills. Another development that affects programmers is progress in developing core tools for statistics and machine learning. This relieves programmers of the need to program intricate algorithms for themselves each time they want to try a new one. Among general-purpose programming languages, Python developers have been in the forefront, building state-of-the-art machine learning tools, but there is a gap between having the tools and being able to use them efficiently.

Programmers can gain general knowledge about machine learning in a number of ways: online courses, a number of well-written books, and so on. Many of these give excellent surveys of machine learning algorithms and examples of their use, but because of the availability of so many different algorithms, it's difficult to cover the details of their usage in a survey.

This leaves a gap for the practitioner. The number of algorithms available requires making choices that a programmer new to machine learning might not be equipped to make until trying several, and it leaves the programmer to fill in the details of the usage of these algorithms in the context of overall problem formulation and solution.

This book attempts to close that gap. The approach taken is to restrict the algorithms covered to two families of algorithms that have proven to give optimum performance for a wide variety of problems. This assertion is supported by their dominant usage in machine learning competitions, their early inclusion in newly developed packages of machine learning tools, and their performance in comparative studies (as discussed in Chapter 1, "The Two Essential Algorithms for Making Predictions"). Restricting attention to two algorithm families makes it possible to provide good coverage of the principles of operation and to run through the details of a number of examples showing how these algorithms apply to problems with different structures.

The book largely relies on code examples to illustrate the principles of operation for the algorithms discussed. I've discovered in the classes I teach at Hacker Dojo in Mountain View, California, that programmers generally grasp principles more readily by seeing simple code illustrations than by looking at math.

This book focuses on Python because it offers a good blend of functionality and specialized packages containing machine learning algorithms. Python is an often-used language that is well known for producing compact, readable code. That fact has led a number of leading companies to adopt Python for prototyping and deployment. Python developers are supported by a large community of fellow developers, development tools, extensions, and so forth. Python is widely used in industrial applications and in scientific programming, as well. It has a number of packages that support computationally-intensive applications like machine learning, and it is a good collection of the leading machine learning algorithms (so you don't have to code them yourself). Python is a better general-purpose programming language than specialized statistical languages such as R or SAS (Statistical Analysis System). Its collection of machine learning algorithms incorporates a number of top-flight algorithms and continues to expand.

Who This Book Is For

This book is intended for Python programmers who want to add machine learning to their repertoire, either for a specific project or as part of keeping their toolkit relevant. Perhaps a new problem has come up at work that requires machine learning. With machine learning being covered so much in the news these days, it's a useful skill to claim on a resume.

This book provides the following for Python programmers:

- A description of the basic problems that machine learning attacks
- Several state-of-the-art algorithms
- The principles of operation for these algorithms
- Process steps for specifying, designing, and qualifying a machine learning system
- Examples of the processes and algorithms
- Hackable code

To get through this book easily, your primary background requirements include an understanding of programming or computer science and the ability to read and write code. The code examples, libraries, and packages are all Python, so the book will prove most useful to Python programmers. In some cases, the book runs through code for the core of an algorithm to demonstrate the operating principles, but then uses a Python package incorporating the algorithm to apply the algorithm to problems. Seeing code often gives programmers an intuitive grasp of an algorithm in the way that seeing the math does for others. Once the understanding is in place, examples will use developed Python packages with the bells and whistles that are important for efficient use (error checking, handling input and output, developed data structures for the models, defined predictor methods incorporating the trained model, and so on).

In addition to having a programming background, some knowledge of math and statistics will help get you through the material easily. Math requirements include some undergraduate-level differential calculus (knowing how to take a derivative and a little bit of linear algebra), matrix notation, matrix multiplication, and matrix inverse. The main use of these will be to follow the derivations of some of the algorithms covered. Many times, that will be as simple as taking a derivative of a simple function or doing some basic matrix manipulations. Being able to follow the calculations at a conceptual level may aid your understanding of the algorithm. Understanding the steps in the derivation can help you to understand the strengths and weaknesses of an algorithm and can help you to decide which algorithm is likely to be the best choice for a particular problem.

This book also uses some general probability and statistics. The requirements for these include some familiarity with undergraduate-level probability and concepts such as the mean value of a list of real numbers, variance, and correlation. You can always look through the code if some of the concepts are rusty for you.

This book covers two broad classes of machine learning algorithms: penalized linear regression (for example, Ridge and Lasso) and ensemble methods (for example, Random Forests and Gradient Boosting). Each of these families contains variants that will solve regression and classification problems. (You learn the distinction between classification and regression early in the book.)

Readers who are already familiar with machine learning and are only interested in picking up one or the other of these can skip to the two chapters covering that family. Each method gets two chapters—one covering principles of operation and the other running through usage on different types of problems. Penalized linear regression is covered in Chapter 4, "Penalized Linear Regression," and Chapter 5, "Building Predictive Models Using Penalized Linear Methods." Ensemble methods are covered in Chapter 6, "Ensemble Methods," and Chapter 7, "Building Predictive Models with Python." To familiarize yourself with the problems addressed in the chapters on usage of the algorithms, you might find it helpful to skim Chapter 2, "Understand the Problem by Understanding the Data," which deals with data exploration. Readers who are just starting out with machine learning and want to go through from start to finish might want to save Chapter 2 until they start looking at the solutions to problems in later chapters.

What This Book Covers

As mentioned earlier, this book covers two algorithm families that are relatively recent developments and that are still being actively researched. They both depend on, and have somewhat eclipsed, earlier technologies.

Penalized linear regression represents a relatively recent development in ongoing research to improve on ordinary least squares regression. Penalized linear regression has several features that make it a top choice for predictive analytics. Penalized linear regression introduces a tunable parameter that makes it possible to balance the resulting model between overfitting and underfitting. It also yields information on the relative importance of the various inputs to the predictions it makes. Both of these features are vitally important to the process of developing predictive models. In addition, penalized linear regression yields best prediction performance in some classes of problems, particularly underdetermined problems and problems with very many input parameters such as genetics and text mining. Furthermore, there's been a great deal of recent development of coordinate descent methods, making training penalized linear regression models extremely fast.

To help you understand penalized linear regression, this book recapitulates ordinary linear regression and other extensions to it, such as stepwise regression. The hope is that these will help cultivate intuition.

Ensemble methods are one of the most powerful predictive analytics tools available. They can model extremely complicated behavior, especially for problems that are vastly overdetermined, as is often the case for many web-based prediction problems (such as returning search results or predicting ad click-through rates). Many seasoned data scientists use ensemble methods as their first try because of their performance. They are also relatively simple to use, and they also rank variables in terms of predictive performance.

Ensemble methods have followed a development path parallel to penalized linear regression. Whereas penalized linear regression evolved from overcoming the limitations of ordinary regression, ensemble methods evolved to overcome the limitations of binary decision trees. Correspondingly, this book's coverage of ensemble methods covers some background on binary decision trees because ensemble methods inherit some of their properties from binary decision trees. Understanding them helps cultivate intuition about ensemble methods.

How This Book Is Structured

This book follows the basic order in which you would approach a new prediction problem. The beginning involves developing an understanding of the data and determining how to formulate the problem, and then proceeds to try an algorithm and measure the performance. In the midst of this sequence, the book outlines the methods and reasons for the steps as they come up. Chapter 1 gives a more thorough description of the types of problems that this book covers and the methods that are used. The book uses several data sets from the UC Irvine data repository as examples, and Chapter 2 exhibits some of the methods and tools that you can use for developing insight into a new data set. Chapter 3, "Predictive Model Building: Balancing Performance, Complexity, and Big Data," talks about the difficulties of predictive analytics and techniques for addressing them. It outlines the relationships between problem complexity, model complexity, data set size, and predictive performance. It discusses overfitting and how to reliably sense overfitting. It talks about performance metrics for different types of problems. Chapters 4 and 5, respectively, cover the background on penalized linear regression and its application to problems explored in Chapter 2. Chapters 6 and 7 cover background and application for ensemble methods.

What You Need to Use This Book

To run the code examples in the book, you need to have Python 2.x, SciPy, NumPy, Pandas, and scikit-learn. These can be difficult to install due to cross-dependencies and version issues. To make the installation easy, I've used a free distribution of these packages that's available from Continuum Analytics (`http://continuum.io/`). Their Anaconda product is a free download and includes Python 2.x and all the packages you need to run the code in this book (and more). I've run the examples on Ubuntu 14.04 Linux but haven't tried them on other operating systems.

Conventions

To help you get the most from the text and keep track of what's happening, we've used a number of conventions throughout the book.

WARNING **Boxes like this one hold important, not-to-be forgotten information that is directly relevant to the surrounding text.**

NOTE **Notes, tips, hints, tricks, and asides to the current discussion are offset and appear like this.**

As for styles in the text:

- We *highlight* new terms and important words when we introduce them.
- We show keyboard strokes like this: Ctrl+A.
- We show filenames, URLs, and code within the text like so: `persistence.properties`.
- We present code in two different ways:

```
We use a monofont type with no highlighting for most code examples.
We use bold to emphasize code that's particularly important in
the present context.
```

Source Code

As you work through the examples in this book, you may choose either to type in all the code manually or to use the source code files that accompany the book. All the source code used in this book is available for download from `http://www.wiley.com/go/pythonmachinelearning`. You will find the code snippets from the source code are accompanied by a download icon and note indicating the name of the program so that you know it's available for download and can easily locate it in the download file. Once at the site, simply locate the book's title (either by using the Search box or by using one of the title lists) and click the Download Code link on the book's detail page to obtain all the source code for the book.

NOTE **Because many books have similar titles, you may find it easiest to search by ISBN; this book's ISBN is 978-1-118-96174-2.**

After you download the code, just decompress it with your favorite compression tool.

Errata

We make every effort to ensure that no errors appear in the text or in the code. However, no one is perfect, and mistakes do occur. If you find an error in one of our books, like a spelling mistake or faulty piece of code, we would be very grateful for your feedback. By sending in errata, you might save another reader hours of frustration, and at the same time you will be helping us provide even higher-quality information.

To find the errata page for this book, go to `http://www.wiley.com` and locate the title using the Search box or one of the title lists. Then, on the book details page, click the Book Errata link. On this page, you can view all errata that has been submitted for this book and posted by Wiley editors.

Machine Learning in Python®