BIG DATA, DATA MINING, AND MACHINE LEARNING

Value Creation for Business Leaders and Practitioners

Jared Dean

WILEY

Additional praise for *Big Data, Data Mining, and Machine Learning: Value Creation for Business Leaders and Practitioners*

"Jared's book is a great introduction to the area of High Powered Analytics. It will be useful for those who have experience in predictive analytics but who need to become more versed in how technology is changing the capabilities of existing methods and creating new possibilities. It will also be helpful for business executives and IT professionals who'll need to make the case for building the environments for, and reaping the benefits of, the next generation of advanced analytics."

—Jonathan Levine, Senior Director, Consumer Insight Analysis at Marriott International

"The ideas that Jared describes are the same ideas that being used by our Kaggle contest winners. This book is a great overview for those who want to learn more and gain a complete understanding of the many facets of data mining, knowledge discovery and extracting value from data."

—Anthony Goldbloom Founder and CEO of Kaggle

"The concepts that Jared presents in this book are extremely valuable for the students that I teach and will help them to more fully understand the power that can be unlocked when an organization begins to take advantage of its data. The examples and case studies are particularly useful for helping students to get a vision for what is possible. Jared's passion for analytics comes through in his writing, and he has done a great job of making complicated ideas approachable to multiple audiences."

—Tonya Etchison Balan, Ph.D., Professor of Practice, Statistics, Poole College of Management, North Carolina State University

Big Data, Data Mining, and Machine Learning

Wiley & SAS Business Series

The Wiley & SAS Business Series presents books that help senior-level managers with their critical management decisions.

Titles in the Wiley & SAS Business Series include:

Activity-Based Management for Financial Institutions: Driving Bottom-Line Results by Brent Bahnub

Analytics in a Big Data World: The Essential Guide to Data Science and its Applications by Bart Baesens

Bank Fraud: Using Technology to Combat Losses by Revathi Subramanian

Big Data Analytics: Turning Big Data into Big Money by Frank Ohlhorst

Branded! How Retailers Engage Consumers with Social Media and Mobility by Bernie Brennan and Lori Schafer

Business Analytics for Customer Intelligence by Gert Laursen

Business Analytics for Managers: Taking Business Intelligence beyond Reporting by Gert Laursen and Jesper Thorlund

The Business Forecasting Deal: Exposing Bad Practices and Providing Practical Solutions by Michael Gilliland

Business Intelligence Applied: Implementing an Effective Information and Communications Technology Infrastructure by Michael Gendron

Business Intelligence and the Cloud: Strategic Implementation Guide by Michael S. Gendron

Business Intelligence Success Factors: Tools for Aligning Your Business in the Global Economy by Olivia Parr Rud

Business Transformation: A Roadmap for Maximizing Organizational Insights by Aiman Zeid

CIO Best Practices: Enabling Strategic Value with Information Technology, second edition by Joe Stenzel

Connecting Organizational Silos: Taking Knowledge Flow Management to the Next Level with Social Media by Frank Leistner

Credit Risk Assessment: The New Lending System for Borrowers, Lenders, and Investors by Clark Abrahams and Mingyuan Zhang

Credit Risk Scorecards: Developing and Implementing Intelligent Credit Scoring by Naeem Siddiqi

The Data Asset: How Smart Companies Govern Their Data for Business Success by Tony Fisher

Delivering Business Analytics: Practical Guidelines for Best Practice by Evan Stubbs

Demand-Driven Forecasting: A Structured Approach to Forecasting, second edition by Charles Chase

Demand-Driven Inventory Optimization and Replenishment: Creating a More Efficient Supply Chain by Robert A. Davis

Developing Human Capital: Using Analytics to Plan and Optimize Your Learning and Development Investments by Gene Pease, Barbara Beresford, and Lew Walker

The Executive's Guide to Enterprise Social Media Strategy: How Social Networks Are Radically Transforming Your Business by David Thomas and Mike Barlow

Economic and Business Forecasting: Analyzing and Interpreting Econometric Results by John Silvia, Azhar Iqbal, Kaylyn Swankoski, Sarah Watt, and Sam Bullard

Executive's Guide to Solvency II by David Buckham, Jason Wahl, and Stuart Rose

Fair Lending Compliance: Intelligence and Implications for Credit Risk Management by Clark R. Abrahams and Mingyuan Zhang

Foreign Currency Financial Reporting from Euros to Yen to Yuan: A Guide to Fundamental Concepts and Practical Applications by Robert Rowan

Harness Oil and Gas Big Data with Analytics: Optimize Exploration and Production with Data Driven Models by Keith Holdaway

Health Analytics: Gaining the Insights to Transform Health Care by Jason Burke

Heuristics in Analytics: A Practical Perspective of What Influences Our Analytical World by Carlos Andre Reis Pinheiro and Fiona McNeill

Human Capital Analytics: How to Harness the Potential of Your Organization's Greatest Asset by Gene Pease, Boyce Byerly, and Jac Fitz-enz

Implement, Improve and Expand Your Statewide Longitudinal Data System: Creating a Culture of Data in Education by Jamie McQuiggan and Armistead Sapp

Information Revolution: Using the Information Evolution Model to Grow Your Business by Jim Davis, Gloria J. Miller, and Allan Russell

Killer Analytics: Top 20 Metrics Missing from your Balance Sheet by Mark Brown

Manufacturing Best Practices: Optimizing Productivity and Product Quality by Bobby Hull

Marketing Automation: Practical Steps to More Effective Direct Marketing by Jeff LeSueur

Mastering Organizational Knowledge Flow: How to Make Knowledge Sharing Work by Frank Leistner

The New Know: Innovation Powered by Analytics by Thornton May

Performance Management: Integrating Strategy Execution, Methodologies, Risk, and Analytics by Gary Cokins

Predictive Business Analytics: Forward-Looking Capabilities to Improve Business Performance by Lawrence Maisel and Gary Cokins

Retail Analytics: The Secret Weapon by Emmett Cox

Social Network Analysis in Telecommunications by Carlos Andre Reis Pinheiro

Statistical Thinking: Improving Business Performance, second edition, by Roger W. Hoerl and Ronald D. Snee

Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics by Bill Franks

Too Big to Ignore: The Business Case for Big Data by Phil Simon

The Value of Business Analytics: Identifying the Path to Profitability by Evan Stubbs

The Visual Organization: Data Visualization, Big Data, and the Quest for Better Decisions by Phil Simon

Visual Six Sigma: Making Data Analysis Lean by Ian Cox, Marie A. Gaudard, Philip J. Ramsey, Mia L. Stephens, and Leo Wright

Win with Advanced Business Analytics: Creating Business Value from Your Data by Jean Paul Isson and Jesse Harriott

For more information on any of the above titles, please visit www.wiley.com.

Big Data, Data Mining, and Machine Learning

Value Creation for Business Leaders and Practitioners

Jared Dean

WILEY

Cover Design: Wiley
Cover Image: © iStockphoto / elly99

Published by John Wiley & Sons, Inc., Hoboken, New Jersey.
Published simultaneously in Canada.

Library of Congress Cataloging-in-Publication Data:

Dean, Jared, 1978-
Big data, data mining, and machine learning : value creation for business leaders and practitioners / Jared Dean.
1 online resource.—(Wiley & SAS business series)
Includes index.
ISBN 978-1-118-92069-5 (ebk); ISBN 978-1-118-92070-1 (ebk); ISBN 978-1-118-61804-2 (hardback) 1. Management—Data processing. 2. Data mining. 3. Big data. 4. Database management. 5. Information technology—Management. I. Title.
HD30.2
658′.05631—dc23

2014009116

Printed in the United States of America
10 9 8 7 6 5 4 3 2 1

To my wife, without whose help, love, and devotion, this book would not exist. Thank you, Katie!

For Geoffrey, Ava, Mason, and Chase: Remember that the quickest path to easy is through hard.

Contents

Foreword

I love the field of predictive analytics and have lived in this world for my entire career. The mathematics are fun (at least for me), but turning what the algorithms uncover into solutions that a company uses and generates profit from makes the mathematics worthwhile. In some ways, Jared Dean and I are unusual in this regard; we really do love seeing these solutions work for organizations we work with. What amazes us, though, is that this field that we used to do in the back office, a niche of a niche, has now become one of the sexiest jobs of the twenty-first century. How did this happen?

We live in a world where data is collected in ever-increasing amounts, summarizing more of what people and machines do, and capturing finer granularity of their behavior. These three ways to characterize data are sometimes described as volume, variety, and velocity—the definition of big data. They are collected because of the perceived value in the data even if we don't know exactly what we will do with it. Initially, many organizations collect it and report summaries, often using approaches from business intelligence that have become commonplace.

But in recent years, a paradigm shift has taken place. Organizations have found that predictive analytics transforms the way they make decisions. The algorithms and approaches to predictive modeling described in this book are not new for the most part; Jared himself describes the big-data problem as nothing new. The algorithms he describes are all at least 15 years old, a testimony to their effectiveness that fundamentally new algorithms are not needed. Nevertheless, predictive modeling is in fact new to many organizations as they try to improve decisions with data. These organizations need to gain an understanding not only of the science and principles of predictive modeling but how to apply the principles to problems that defy the standard approaches and answers.

But there is much more to predictive modeling than just building predictive models. The operational aspects of predictive modeling

projects are often overlooked and are rarely covered in books and courses. First, this includes specifying hardware and software needed for a predictive modeling. As Jared describes, this depends on the organization, the data, and the analysts working on the project. Without setting up analysts with the proper resources, projects flounder and often fail. I've personally witnessed this on projects I have worked on, where hardware was improperly specified causing me to spend a considerable amount of time working around the limitations in RAM and processing speed.

Ultimately, the success of predictive modeling projects is measured by the metric that matters to the organization using it, whether it be increased efficiency, ROI, customer lifetime value, or soft metrics like company reputation. I love the case studies in this book that address these issues, and you have a half-dozen here to whet your appetite. This is especially important for managers who are trying to understand how predictive modeling will impact their bottom line.

Predictive modeling is science, but successful implementation of predictive modeling solutions requires connecting the models to the business. Experience is essential to recognize these connections, and there is a wealth of experience here to draw from to propel you in your predictive modeling journey.

Dean Abbott
Abbott Analytics, Inc.
March 2014

Preface

This book project was first presented to me during my first week in my current role of managing the data mining development at SAS. Writing a book has always been a bucket-list item, and I was very excited to be involved. I've come to realize why so many people want to write books, but why so few get the chance to see their thoughts and ideas bound and published.

I've had the opportunity during my studies and professional career to be front and center to some great developments in the area of data mining and to study under some brilliant minds. This experience helped position me with the skills and experience I needed to create this work.

Data mining is a field I love. Ever since childhood, I've wanted to explain how things work and understand how systems function both in the "average" case but also at the extremes. From elementary school through high school, I thought engineering would be the job that would couple both my curiosity and my desire to explain the world around me. However, before my last year as an undergraduate student, I found statistics and information systems, and I was hooked.

In Part One of the book, I explore the foundations of hardware and system architecture. This is a love that my parents were kind enough to indulge me in, in a day when computers cost much much more than $299. The first computer in my home was an Apple IIc, with two 5.25" floppy disk drives and no hard drive. A few years later I built an Intel 386 PC from a kit, and I vividly remember playing computer games and hitting the turbo button to move the CPU clock speed from 8 MHz to 16 MHz. I've seen Moore's Law firsthand, and it still amazes me that my smartphone holds more computing power than the computers used in the Mercury space program, the Apollo space program, and the Orbiter space shuttle program combined.

After I finished my undergraduate degree in statistics, I began to work for the federal government at the U.S. Bureau of the Census. This is where I got my first exposure to big data. Prior to joining the Census

Bureau, I had never written a computer program that took more than a minute to run (unless the point was to make the program run for more than a minute). One of my first projects was working with the Master Address File (MAF),[1] which is an address list maintained by the Census Bureau. This address list is also the primary survey frame for current surveys that the Census Bureau administers (yes, there is lots of work to do the other nine years). The list has more than 300 million records, and combining all the address information, longitudinal information, and geographic information, there are hundreds of attributes associated with each housing unit. Working with such a large data set was where I first learned about programming efficiency, scalability, and hardware optimization. I'm grateful to my patient manager, Maryann, who gave me the time to learn and provided me with interesting, valuable projects that gave me practical experience and the opportunity to innovate. It was a great position because I got to try new techniques and approaches that had not been studied before in that department. As with any new project, some ideas worked great and others failed. One specific project I was involved in was trying to identify which blocks (the Census Bureau has the United States divided up into unique geographic areas—the hierarchy is state, county, track, block group, and block; there are about 8.2 million blocks in the United States) from Census 2000 had been overcounted or undercounted. Through the available data, we did not have a way to verify that our model for predicting the deviation of actual housing unit count from reported housing unit count was accurate. The program was fortunate to have funding from congress to conduct field studies to provide feedback and validation of the models. This was the first time I had heard the term "data mining" and I was first exposed to SAS™ Enterprise Miner® and CART® by Salford Systems. After a period of time working for the Census Bureau, I realized that I needed more education to achieve my career goals, and so I enrolled in the statistics department at George Mason University in Fairfax, VA.

During graduate school, I learned in more detail about the algorithms common to the fields of data mining, machine learning, and statistics; these included survival analysis, survey sampling, and

[1] The MAF is created during decennial census operations for every housing unit, or potential housing unit, in the United States.

computational statistics. Through my graduate studies, I was able to merge the lessons taught in the classroom to the practical data analysis and innovations required in the office. I acquired an understanding of the theory and the relative strengths and weaknesses of different approaches for data analysis and predictive analytics.

After graduate school, I changed direction in my career, moving from a data analysis[2] role and becoming a software developer. I went to work for SAS Institute Inc., where I was participating in the creation of the software that I had previously used. I had moved from using the software to building it. This presented new challenges and opportunities for growth as I learned about the rigorous numerical validation that SAS imposes on the software, along with its thorough documentation and tireless effort to make new software enhancements consistent with existing software and to consistently deliver new software features that customers need.

During my years at SAS, I've come to thoroughly understand how the software is made and how our customers use it. I often get the chance to visit with customers, listen to their business challenges, and recommend methods or process that help lead them to success; creating value for their organizations.

It is from this collection of experience that I wrote this book, along with the help of the wonderful staff and my colleagues both inside and outside of SAS Institute.

[2] I was a data scientist before the term was invented

Acknowledgments

I would like to thank all those who helped me to make this book a reality. It was a long journey and a wonderful learning and growing experience.

Patrick Hall, thank you for your validation of my ideas and contributing many of your own. I appreciate that I could discuss ideas and trends with you and get thoughtful, timely, and useful feedback.

Joseph Pingenot, Ilknur Kabul, Jorge Silva, Larry Lewis, Susan Haller, and Wendy Czika, thank you for sharing your domain knowledge and passion for analytics.

Michael Wallis, thank you for your help in the text analytics area and developing the *Jeopardy!* example.

Udo Sglavo and Taiyeong Lee, thank you for reviewing and offering significant contributions in the analysis of times series data mining.

Barbara Walters and Vicki Jones, thank you for all the conversations about reads and feeds in understanding how the hardware impacted the software.

Jared Peterson for his help in downloading the data from my Nike+ FuelBand.

Franklin So, thank you for your excellent description of a customer's core business problem.

Thank you Grandma Catherine Coyne, who sacrificed many hours to help a fellow author in editing the manuscript to greatly improve its readability. I am very grateful for your help and hope that when I am 80-something I can be half as active as you are.

I would also like to thank the staff of SAS Press and John Wiley & Sons for the feedback and support through all phases of this project, including some major detours along the way.

Finally, I need to acknowledge my wife, Katie, for shouldering many burdens as I researched, wrote, edited, and wrote more. Meeting you was the best thing that has happened to me in my whole life.

Introduction

Hiding within those mounds of data is knowledge that could change the life of a patient, or change the world.

—Atul Butte, Stanford University

"Cancer" is the term given for a class of diseases in which abnormal cells divide in an uncontrolled fashion and invade body tissues. There are more than 100 unique types of cancer. Most are named after the location (usually an organ) where they begin. Cancer begins in the cells of the body. Under normal circumstances, the human body controls the production of new cells to replace cells that are old or have become damaged. Cancer is not normal. In patients with cancer, cells do not die when they are supposed to and new cells form when they are not needed (like when I ask my kids to use the copy machine and I get back ten copies instead of the one I asked for). The extra cells may form a mass of tissue; this is referred to as a tumor. Tumors come in two varieties: benign tumors, which are not cancerous, and malignant tumors, which are cancerous. Malignant tumors spread through the body and invade the tissue. My family, like most I know, has lost a family member to the disease. There were an estimated 1.6 million new cases of cancer in the United States in 2013 and more than 580,000 deaths as a result of the disease.

An estimated 235,000 people in the United States were diagnosed with breast cancer in 2014, and about 40,000 people will die in 2014 as a result of the disease. The most common type of

breast cancer is ductal carcinoma, which begins in the lining of the milk ducts. The next most common type of breast cancer is lobular carcinoma. There are a number of treatment options for breast cancer including surgery, chemotherapy, radiation therapy, immunotherapy, and vaccine therapy. Often one or more of the treatment options is used to help ensure the best outcome for patients. About 60 different drugs are approved by the Food and Drug Administration (FDA) for the treatment of breast cancer. The course of treatment and which drug protocols should be used is decided based on consultation between the doctor and patient, and a number of factors go into those decisions.

One of the FDA-approved drug treatments for breast cancer is tamoxifen citrate. It is sold under the brand name of Nolvadex and was first prescribed in 1969 in England but approved by the FDA in 1998. Tamoxifen is normally taken as a daily tablet with doses of 10 mg, 20 mg, or 40 mg. It carries a number of side effects including nausea, indigestion, and leg cramps. Tamoxifen has been used to treat millions of women and men diagnosed with hormone-receptor-positive breast cancer. Tamoxifen is often one of the first drugs prescribed for treating breast cancer because it has a high success rate of around 80%.

Learning that a drug is 80% successful gives us hope that tamoxifen will provide good patient outcomes, but there is one important detail about the drug that was not known until the big data era. It is that tamoxifen is not 80% effective in patients but 100% effective in 80% of patients and ineffective in the rest. That is a life-changing finding for thousands of people each year. Using techniques and ideas discussed in this book, scientists were able to identify genetic markers that can identify, in advance, if tamoxifen will effectively treat a person diagnosed with breast cancer. This type of analysis was not possible before the era of big data. Why was it not possible? Because the volume and granularity of the data was missing; volume came from pooling patient results and granularity came from DNA sequencing. In addition to the data, the computational resources needed to solve a problem like this were not readily available to most scientists outside of the super computing lab. Finally the third component, the algorithms or modeling techniques needed to understand this relationship, have matured greatly in recent years.

The story of Tamoxifen highlights the exciting opportunities that are available to us as we have more and more data along with computing resources and algorithms that aid in classification and prediction. With knowledge like that was gained by the scientists studying tamoxifen, we can begin to reshape the treatment of disease and disrupt positively many other areas of our lives. With these advances we can avoid giving the average treatment to everyone but instead determine which people will be helped by a particular drug. No longer will a drug be 5% effective; now we can identify which 5% of patients the drug will help. The concept of personalized medicine has been discussed for many years. With advances in working with big data and improved predictive analytics, it is more of a reality than ever. A drug with a 2% success rate will never be pursued by a drug manufacturer or approved by the FDA unless it can be determined which patients it will help. If that information exists, then lives can be saved. Tamoxifen is one of many examples that show us the potential that exists if we can take advantage of the computational resources and are patient enough to find value in the data that surrounds us.

We are currently living in the big data era. That term "big data" was first coined around the time the big data era began. While I consider the big data era to have begun in 2001, the date is the source of some debate and impassioned discussion on blogs—and even the *New York Times*. The term "big data" appears to have been first used, with its currently understood context, in the late 1990s. The first academic paper was presented in 2000, and published in 2003, by Francis X. Diebolt—"Big Data Dynamic Factor Models for Macroeconomic Measurement and Forecasting"—but credit is largely given to John Mashey, the chief scientist for SGI, as the first person to use the term "big data." In the late 1990s, Mashey gave a series of talks to small groups about this big data tidal wave that was coming. The big data era is an era described by rapidly expanding data volumes, far beyond what most people imagined would ever occur.

The large data volume does not solely classify this as the big data era, because there have always been data volumes larger than our ability to effectively work with the data have existed. What sets the current time apart as the big data era is that companies, governments,

and nonprofit organizations have experienced a shift in behavior. In this era, they want to start using all the data that it is possible for them to collect, for a current or future unknown purpose, to improve their business. It is widely believed, along with significant support through research and case studies, that organizations that use data to make decisions over time in fact do make better decisions, which leads to a stronger, more viable business. With the velocity at which data is created increasing at such a rapid rate, companies have responded by keeping every piece of data they could possibly capture and valuing the future potential of that data higher than they had in the past. How much personal data do we generate? The first question is: What is personal data? In 1995, the European Union in privacy legislation defined it as any information that could identify a person, directly or indirectly. International Data Corporation (IDC) estimated that 2.8 zettabytes[1] of data were created in 2012 and that the amount of data generated each year will double by 2015. With such a large figure, it is hard to understand how much of that data is actually about you. It breaks down to about 5 gigabytes of data per day for the average American office worker. This data consists of email, downloaded movies, streamed audio, Excel spreadsheets, and so on. In this data also includes the data that is generated as information moves throughout the Internet. Much of this generated data is not seen directly by you or me but is stored about us. Some examples of nondirect data are things like traffic camera footage, GPS coordinates from our phones, or toll transactions as we speed through automated E-ZPass lanes.

Before the big data era began, businesses assigned relatively low value to the data they were collecting that did not have immediate value. When the big data era began, this investment in collecting and storing data for its potential future value changed, and organizations made a conscious effort to keep every potential bit of data. This shift in behavior created a virtuous circle where data was stored and then, because data was available, people were assigned to find value in it for the organization. The success in finding value led to more data being gathered and so on. Some of the data stored was a dead end, but many times the

[1] A zettabyte is 1 billion terabytes.

results were confirmed that the more data you have, the better off you are likely to be. The other major change in the beginning of the big data era was the rapid development, creation, and maturity of technologies to store, manipulate, and analyze this data in new and efficient ways.

Now that we are in the big data era, our challenge is not getting data but getting the right data and using computers to augment our domain knowledge and identify patterns that we did not see or could not find previously.

Some key technologies and market disruptions have led us to this point in time where the amount of data being collected, stored, and considered in analytical activities has grown at a tremendous rate. This is due to many factors including Internet Protocol version 6 (IPv6), improved telecommunications equipment, technologies like RFID, telematics sensors, the reduced per unit cost of manufacturing electronics, social media, and the Internet.

Here is a timeline that highlights some of the key events leading up to the big data era and events that continue to shape the usage of big data and the future of analytics.

BIG DATA TIMELINE

Here are a number of items that show influential events that prepared the way for the big data era and significant milestones during the era.

1991

- The Internet, or World Wide Web as we know it, is born. The protocol Hypertext Transfer Protocol (HTTP) becomes the standard means for sharing information in this new medium.

1995

- Sun releases the Java platform. Java, invented in 1991, has become the second most popular language behind C. It dominates the Web applications space and is the de facto standard for middle-tier applications. These applications are the source for recording and storing web traffic.
- Global Positioning System (GPS) becomes fully operational. GPS was originally developed by DARPA (Defense Advanced Research Projects Agency) for military applications in the early 1970s.

This technology has become omnipresent in applications for car and airline navigation and finding a missing iPhone.

1998

- Carlo Strozzi develops an open-source relational database and calls it NoSQL. Ten years later, a movement to develop NoSQL databases to work with large, unstructured data sets gains momentum.
- Google is founded by Larry Page and Sergey Brin, who worked for about a year on a Stanford search engine project called BackRub.

1999

- Kevin Ashton, cofounder of the Auto-ID Center at the Massachusetts Institute of Technology (MIT), invents the term "the Internet of Things."

2001

- Wikipedia is launched. The crowd-sourced encyclopedia revolutionized the way people reference information.

2002

- Version 1.1 of the Bluetooth specification is released by the Institute of Electrical and Electronics Engineers (IEEE). Bluetooth is a wireless technology standard for the transfer of data over short distances. The advancement of this specification and its adoption lead to a whole host of wearable devices that communicate between the device and another computer. Today nearly every portable device has a Bluetooth receiver.

2003

- According to studies by IDC and EMC, the amount of data created in 2003 surpasses the amount of data created in all of human history before then. It is estimated that 1.8 zettabytes (ZB) was created in 2011 alone (1.8 ZB is the equivalent of 200 billion high-definition movies, each two hours long, or 47 million years of footage with no bathroom breaks).
- LinkedIn, the popular social networking website for professionals, launches. In 2013, the site had about 260 million users.

2004

- Wikipedia reaches 500,000 articles in February; seven months later it tops 1 million articles.
- Facebook, the social networking service, is founded by Mark Zuckerberg and others in Cambridge, Massachusetts. In 2013, the site had more than 1.15 billion users.

2005

- The Apache Hadoop project is created by Doug Cutting and Mike Caferella. The name for the project came from the toy elephant of Cutting's young son. The now-famous yellow elephant becomes a household word just a few years later and a foundational part of almost all big data strategies.
- The National Science Board recommends that the National Science Foundation (NSF) create a career path for "a sufficient number of high-quality data scientists" to manage the growing collection of digital information.

2007

- Apple releases the iPhone and creates a strong consumer market for smartphones.

2008

- The number of devices connected to the Internet exceeds the world's population.

2011

- IBM's Watson computer scans and analyzes 4 terabytes (200 million pages) of data in seconds to defeat two human players on the television show *Jeopardy!* (There is more about the show in Part Two.)
- Work begins in UnQL, a query language for NoSQL databases.
- The available pools in the IPv4 address space have all been assigned. IPv4 is a standard for assigning an Internet protocol (IP) address. The IPv4 protocol was based on a 32-bit number, meaning there are 2^{32} or 4.5 billion unique addresses available. This event shows the real demand and quantity of Internet-connected devices.

2012

- The Obama administration announces the Big Data Research and Development Initiative, consisting of 84 programs in six departments. The NSF publishes "Core Techniques and Technologies for Advancing Big Data Science & Engineering."
- IDC and EMC estimate that 2.8 ZB of data will be created in 2012 but that only 3% of what could be usable for big data is tagged and less is analyzed. The report predicts that the digital world will by 2020 hold 40 ZB, 57 times the number of grains of sand on all the beaches in the world.
- The *Harvard Business Review* calls the job of data scientist "the sexiest job of the 21st century."

2013

- The democratization of data begins. With smartphones, tablets, and Wi-Fi, everyone generates data at prodigious rates. More individuals access large volumes of public data and put data to creative use.

The events of the last 20 years have fundamentally changed the way data is treated. We create more of it each day; it is not a waste product but a buried treasure waiting to be discovered by curious, motivated researchers and practitioners who see these trends and are reaching out to meet the current challenges.

WHY THIS TOPIC IS RELEVANT NOW

You've read this far in the book because I expect you are looking for ideas and information to help you turn data into information and knowledge. What I hope you learn in the subsequent pages are strategies and concrete ideas for accomplishing your business objective or personal edification regarding how you can harness the data to better your situation, whether in the office, the home, or a fantasy football league.

You should also understand that this is not a new problem—data has always been "too big" to work with effectively. This problem has only been exacerbated as now individuals are generating so much more data than ever before as they go through their daily lives. This

increase in data, however, has caused the information management industry to provide better solutions than ever on how to store, manage, and analyze the data we are producing.

In addition, we also have more opportunity to engage with data. A simple example that is discussed in more detail in Part Two is the recommendations you get from Amazon. That small application at the bottom of its web pages illustrates this point very well. In order to make these recommendations, Amazon can use a few different techniques that mostly center on three pieces of information; how you are similar to other shoppers, similar shoppers' opinions of the product you are viewing, and what product similar shoppers ultimately purchased. Alternatively, Amazon could make recommendations from an item point of view. Take, for example, my recent purchase of a baseball glove. The recommendations included items like baseball bats, baseballs, baseball glove oil, and other baseball-related equipment. These recommendations are based on item-to-item recommendations. Baseball gloves are usually sold with baseballs, bats, and glove oil so Amazon recommends them to me. The other method is to look at my profile and find users who have purchased similar items or have similar details as they relate to Amazon and then recommend to me what they purchased. To be effective at making recommendations requires a real commitment to recording, storing, and analyzing extremely large volumes of data.

I think you only need to look up from this book to see a device that is generating data at this very moment. That data will soon be used to inform some business process, recommendation, or public safety issue. This not a future or theoretical problem, this is now.

IS BIG DATA A FAD?

> *Data! Data! Data! he cried impatiently. I can't make bricks without clay!*
>
> —Sherlock Holmes, "The Adventure in the Copper Beeches"

Many informed individuals in the analytics and information technology (IT) communities are becoming sensitive to the actual term "big data." It has no doubt been co-opted for self-promotion by many people and organizations with little or no ties to storing and processing

large amounts of data or data that requires large amounts of computation. Aside from marketing and promotional mischaracterizations, the term has become vague to the point of near meaninglessness even in some technical situations. "Big data" once meant petabyte scale, unstructured chunks of data mined or generated from the Internet.

I submit that "big data" has expanded to mean a situation where the logistics of storing, processing, or analyzing data have surpassed traditional operational abilities of organizations; said another way, you now have too much data to store effectively or compute efficiently using traditional methods. This may also include having too little time to process the data to use the information to make decisions. In addition, big data means using all available data in your models, not just samples. Big data indicates the capability of using entire data sets instead of just segments as in years past. In the purview of wider popular usage, the definition of this term will likely continue to broaden.

For certain leaders in the field, "big data" is fast becoming, or perhaps always was, just data. These people are long accustomed to dealing with the large amounts of data that other fields are just beginning to mine for information. For evidence, look to the larger web companies and certain entities within the U.S. government that have been using extremely large amounts of unstructured data operationally for years, long before anyone ever coined the term "big data." To them, it was just "data." In addition, the big banks and insurance companies have been pushing up against the limits of commercial, column-oriented data storage technologies for decades, and to them this was just "data" too. Consider whether the scale at which Google is indexing all available information, or which the National Security Agency is recording, has really changed since before the term "big data" entered the popular lexicon. It was difficult to comprehend how much data this was before, and it is still is just as hard to comprehend. However, to these leaders in the field, dealing with it is just a day's work. The rest of world is now joining these industries in storing, computing, and analyzing these immense amounts of data, and now we have a word to describe it and a time period to reference. Figure I.1 shows the popularity of the term "big data" as it came into common usage beginning in 2011. Since the 1940s when computer was a job title or in the 1960s when file transfer involved

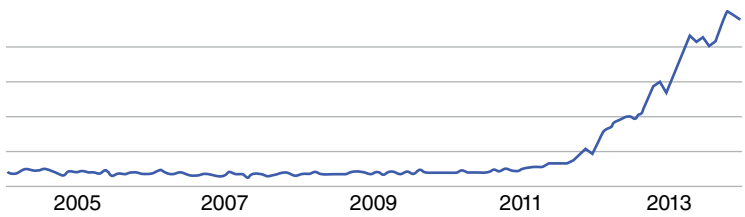

Figure I.1 Trend of Google Searches of "Big Data" over Time Showing the Popularity of the Term
Source: Google Trends

moving a carton of punch cards from one location to another and hoping you did not trip, organizations have had data challenges. Today those challenges are just on a larger scale. Nearly every company must deal with these challenges or accept the idea that the company itself may become irrelevant. *Forbes* in 2013 published an article that said that companies without a big data strategy miss out on $71.2 million per year. If you could raise revenue over $70 million this year and each subsequent year, I am sure your future would be very bright in your organization. The key to capitalize on this opportunity is to have a well-thought-out strategy on big data and execute to the strategy.

To be clear, the solutions surrounding the storage, processing, and analyzing "big data" are not a fad even if the term turns out to be one. Although some are overhyped right now, they are extremely valuable and in some cases actually revolutionary technologies. They have drawn such attention for this reason. Data is not magic—it's just a valuable raw material that can be refined and distilled into valuable specific insights. These insights are then converted to information that eventually creates knowledge.

The bigger the data, the more resource-intensive it is to work with, the better the value of the information must be to make the trade-off a wise business decision. While there is simply no underlying principle stating that the size of data is positively correlated with its value, the size of a data set is positively correlated with its cost to maintain. The value of using big data is defined by how valuable the information gleaned from its process is compared to the time and resources it took to process that information.

This being said, there is a great deal of evidence that prudent data analysis can create value, whether knowledge or monetary, for most organizations. This is why the technologies that allow us to store, process, and analyze large amounts of data will continue to receive increased usage and are anything but a passing fad. For the leading analytic companies and entities, using large amounts of data has been a favorable value proposition for some time, and there is no reason for that trend to decelerate.

Data is the new oil. It has all of the same challenges in that it is plentiful but difficult and sometimes messy to extract. There are a few entities that control most of it, and a vast infrastructure has been built to transport, refine, and distribute it. With the performance of the required technologies increasing and their prices decreasing, if organizations currently struggle to deal efficiently with the medium-size data, they will be ill prepared and at a strategic disadvantage against their competition when data sizes increase, which they inevitably will do. If nothing else, companies, hospitals, and universities will face competition that will drive them to adopt technology to handle the growing volume of data. Other organizations, such as nongovernmental agencies, may be slower to invest in the new generation of data technologies and personnel. This is due to planning and implementation costs as well as the shortage of analytical professionals needed to produce value from this significant investment in hardware human capital; and some smaller organizations will not need sophisticated analysis to understand their operational environment.

WHERE USING BIG DATA MAKES A BIG DIFFERENCE

There have been so many news stories and hype about big data, and how it can transform your business, that it begins to sound like something you would find in Shangri-La. It is often portrayed as the answer to all things that cause problems for organizations. There are promises that it will identify the right customers for marketing campaigns and help academic institutions select the perfect students for their admissions process. Don't let skepticism turn you into a cynic.

It is indeed true that having more data, especially historical data, often will help model predictions be more accurate. Using additional

sources of data, such as social networks, will help organizations make better predictions about customer choices and preferences, because all of us are influenced to some degree by those in our social network, either the physical or the virtual one.

Consider a situation where I have a poor customer experience with my cable provider; so poor that I cancel all of my services and look for another provider. Does this situation make my friends, family, and associates more likely or less likely purchase new services? How does that knowledge of my cancellation because of poor customer service, along with knowing my close friends, family, and associates, affect the cable provider's action? This is a prime example of big data in action; five to ten years ago, this type of analysis would not have been possible because the data sources just did not exist. The answer to my question of how this affects those who know is that my family and friends are less likely to add new services and potentially may follow suit and cancel their cable service as well. Having more data about your customers, products, and processes allows you to consider these types of effects in predicting customers' future behavior. It also needs to be pointed out that having the data and doing the analysis are vital steps to taking advantage of the opportunity in the big data era, but unless the organization is equipped to use new data sources and methods in their processes and act on this information, all of the data and the best analysis in the world will not help it improve.

There is danger in not taking the time to know how much weight to give to this large amount of newly available information, when that information is compared to all of the other attributes that affect a person's decision-making process. Returning to the poor customer service and my social network example, it is clear that the people in my social network are now more likely to cancel services, but how much more likely? One company that takes many different metrics and creates a single aggregate score is Klout; the higher your Klout score, the more influential you are online. Klout uses comments, mentions, retweets, likes and so on to create this score. The company is able to measure online influence only because that is all the data to which it has access.

The question of sampling and big data is a hotly debated topic, and I have read and heard on several occasions that sampling is dead. As a

trained statistician and former employee of the U.S. Census Bureau, I would never say that sampling is dead or that it has no place in business today. Sampling is useful and valid. For certain types of problems, sampling from the population yields just as good a result as performing the same analysis using the entire population (all the data).

However, sampling cannot meet the objectives of many critical high-value projects, such as finding outliers. Companies that cling to sampling will miss out on opportunities to learn insights that can be found only by considering all the data. Outliers are one such example. In a statistical case, the term "outliers" usually has a negative connotation. But in many business problems, the outliers are your most profitable customers or the new market segments that can be exploited.

The best advice is to be an informed data user. Having seven years of customer history instead of three in August 2008 would not have helped in any way to predict people's upcoming spending habits in the United States. In just a few weeks, the financial markets were going to collapse and several large investment firms were going to declare bankruptcy or be purchased at fire sale prices. The U.S. government would begin a massive bailout of the financial markets that would, in some way, affect everyone in the industrialized world. No amount of data would have helped. No amount of data analysis modeling of the preceding months' spending could forecast what the months after the bailout would look like for the average consumer. In order to build useful models in that time, you needed competent practitioners who understood how to simulate and adjust for economic conditions that no one in the workforce had seen before.

There are, however, two major advantages of using all the available data in solving your analytical business problems. The first is technical, and the other is productivity through an improved workflow for analysts.

Technical Issue

Many statistical and machine learning techniques are averaging processes. An averaging process is an algorithm or methodology that seeks to minimize or maximize the overall error and therefore make

the best prediction for the average situation. Two examples of averaging algorithms are linear regression and neural networks. Both of these methods are explained in more detail in Part Two, but for now understand that regression seeks to minimize the overall error by fitting a line that minimizes the squared distance from the line to the data points. The square is used because the distances from the line to the data points will be both negative and positive. A neural network works by connecting all the input, or dependent variables, to a hidden set of variables and iteratively reweighting the connections between them until the classification of the holdout sample cannot be improved.

These averaging methods can be used on big data, and they will work very well. It is also very common for these methods to be very efficient in processing the data due to clever and persistent developers who have organized the computation in a way that takes advantage of modern computing systems. These systems, which have multiple cores, can each be working on a part of the problem; to get even greater speedups, multiple computers can be used in a distributed computing environment. Nonparametric techniques, such as a rank sign test, also fall into this category of an averaging model technique. I call this type of algorithm an averaging model process. Because we are seeking to average, sampling is a potential substitute that can provide comparable answers.

However, a second class of modeling problem is not averaging but is an extremity-based model or tail-based modeling process. These model types are used for a different objective and seek to find extreme or unusual values. These are sometimes referred to as outliers, but not in the strict statistical scenes. Instead, there are notable and unusual points or segments that are often the problems that are the most challenging to companies and carry the biggest return on investment in the present business climate. Examples of tail-based processes are fraud detection, offer optimization, manufacturing quality control, or microsegmentation in marketing.

Next I show why it is imperative to use complete data sets in these types of problems from several different industries and domains. In these cases and many others, these problems cannot be solved effectively without all the data, which is large and complex.

Fraud

If the rate of credit card fraud is 1 in 1,000 transactions,[2] and you sample 20% of your data to build your fraud model, it is likely that you will not have a single fraudulent activity. In fact, if you take anything less than all the data, you likely will never include the fraudulent activity. Predictive models work by using past performance to identify current behavior that has the same characteristics. Without having those fraudulent transactions, you will lack enough information to create an equation that can recognize fraud when it is happening. It is necessary to capture and analyze past fraudulent activity to create effective models for predicting future fraudulent activity.

In addition, if you are sampling the current incoming credit card transactions looking for fraud, you will miss those transactions that could have been flagged had they been present in the data being analyzed.

Optimization

For the optimization family of problems, consider the example of U.S. commercial airline traffic. If we want to understand the congestion points in the air traffic network, or model the effect of a tornado in Dallas, or a tropical storm in Miami, we need to use the entire data set. This is necessary to measure and accurately describe the scope of the problem and its effects on overall flight delays or cancellations and measure the costs to the industry. Imagine if a sample was taken in this case. First we would have to design a sampling scheme. Do you take a sequential sample of every tenth flight across the system? Do you weigh each airport by the number of commercial flights that occur there on the average day? Do you use a random sample? All of these schemes have shortfalls in seeing the complete picture, and those shortfalls will be most prominent in the calculation of the standard error and the confidence intervals of the prediction. To do these calculations correctly will take time to measure, calculate, and verify. If it is not done correctly, then your answer is wrong, but you will never know. The sampling also requires a front-loaded investment; before

[2] The incidence rate is actually much smaller, but this makes for easier math.

I can work with the sampled data, I have to invest the time to create the sample and validate that it is accurate. I also cannot see any type of result until a significant amount of work is already completed.

Relationship Management

Retention of telecommunications customers is a key component to revenue stream. Consider the challenge of predicting which customers are likely to cancel their contract and move to a different carrier. (This is referred to as attrition or churn.) It would be useful to know the typical behavior patterns of customers in the period of time before they cancel, including whom they call or from whom they receive calls. If you use only 10% of the calls each customer sends or receives, use only 10% of the customers, or look at only 10% of the numbers they call, you could be misled in predicting the true likelihood that a particular customer will cancel their contract. This is true for two reasons; first, since only a small percentage of customers leave each month, it would be probable that not a single dissatisfied costumer (or even a whole segment of customers) would be included in the sample. It would also be possible that some of a dissatisfied customer's calls are not included in the sample. However, without the complete set of calls for a given customer, it is much more difficult to identify the pattern that you are looking for. (This is like working on a puzzle that is missing most of the pieces.) With the inability to identify those customers likely to cancel their contract, the problem will grow over time. Given the significant costs to acquire new customers in the telecommunications market, implementing an effective strategy to keep existing customers is worth millions and millions of dollars in annual revenue.

Work Flow Productivity

The second consideration is ensuring that the productivity of the analyst stays as high as possible. Analytics has become a very hot topic in the past few years, and predictions from McKinsey & Company project a shortfall of 140,000 to 190,000 people with the analytical expertise and 1.5 million managers needed to evaluate and make decisions based on big data. This translates to a deficit of 50% to 60% of the required personnel by the year 2018 in the United States alone.

With this significant shortfall in capable people, the human capital you have already made in your organization needs to be preserved and improved. The analytical talent you already have in your organization will become more scarce as other organizations work to make better use of their big data through better analytics and governance. It will be critical to keep analytical talent engaged and productive.

From the same report:

> "Several issues will have to be addressed to capture the full potential of big data. Policies related to privacy, security, intellectual property, and even liability will need to be addressed in a big data world. Organizations need not only to put the right talent and technology in place but also structure workflows and incentives to optimize the use of big data. Access to data is critical—companies will increasingly need to integrate information from multiple data sources, often from third parties, and the incentives have to be in place to enable this." (McKinsey) www.mckinsey.com/insights/business_technology/big_data_the_next_frontier_for_innovation

As I mentioned in the prior section, the work to sample is a very front-loaded task; the majority of the work is done before any results can be created or exploration can begin. This is really backward from the optimum work flow. The best thing is to make the access to data exploration and quick modeling against the data simple and readily available. Organizations should enable the "failing fast" paradigm. Fail has a negative connotation, but failing fast is a useful strategy for determining which projects have merit and which do not. When people have the ability to work with the entire set of data, they can explore and prototype in a more efficient and natural way that does not require a great deal of up-front work to access the data. To enable this type of environment for your organization, ongoing commitments to capital and technological investments are required.

Making effective work flow and data computing resources for employees translates to large productivity gains and short timelines to pay back the return on investment. I have seen this transformation firsthand when I was working on a credit card modeling project for a large U.S. bank. Using the traditional methods (hardware and software), it

was taking many hours to solve the problem. When I switched to a new distributed computing environment, I was able to solve the same problem in two to three minutes. I no longer had to multitask across so many projects because each one had significant downtime while models were being built. I was able to try a number of algorithms and tune each one to a degree that would not have been possible before. The work flow was reminiscent of class projects in school where data volumes were small and software ran nearly instantaneously. This was the method I had been trained in, and it felt more natural. I saw immediate benefits in the form of better model lift, which the customer saw as millions of dollars in revenue increases.

The Complexities When Data Gets Large

Big data is not inherently harder to analyze than small data. The computation of a mean is still just the sum of the values divided by the number of observations, and computing a frequency table still requires reading the data and storing the number of times a distinct value occurs in the data. Both of these situations can be done by reading the data only one time. However, when data volumes gets large or when the data complexity increases, analytics run times can grow to the point that they take longer to compute than the operational constraints will allow. This can result in misleading results or a failure to find a result at all.

Nonlinear Relationships

In real data, there are often nonlinear relationships between variables. Sometimes these relationships can be described "well enough" using linear relationships, but sometimes they cannot. A linear relationship is sometimes hard to imagine, so let us use exiting a parking lot as an example. My family and a few others attended a symphony performance and fireworks show for the Fourth of July (Independence Day in the United States). We parked near each other in the same section of the parking lot, which was about a seven-minute walk from the venue. After the fireworks concluded, our group made our way to the exit, but one of the families became separated. Instead of taking the closest exit to our seats that is open only after the event, they took

a longer route through the venue to the main entrance where we had entered. This alternate route added about three minutes to their departure time. A parking lot after a major event is always something I, as a quantitative person, dread. I easily grow frustrated over inefficiency, and this exit situation is known to be poor and bordering on terrible. My family arrived at the car, loaded our cooler, chairs, and blankets, and began to drive to the exit. Traffic inside the parking lot was quite slow, because of poor visibility and all the pedestrian traffic leaving the venue. We proceeded to move with the traffic and following police direction made it home in about 40 minutes.[3] As we were arriving home, my wife received a text message from our friends who had taken the other exit, asking if we were stuck in the parking lot like they were. So, while our drive took twice as long as it does on a normal day, those three extra minutes added not three minutes (which is what we would expect from a linear relationship between time of departure and time of arrival) to their drive but almost another 45 minutes to their drive home (in addition to the 40 minutes it took my family) from the event. This example is one many can relate to, and it illustrates an important point: Knowing about your data can be a huge asset in applying analytics to your data.

A second example of nonlinear relationship is that of the space shuttle Challenger disaster in 1986. Even though it has been almost 30 years, I still remember sitting in Mrs. Goodman's class in my elementary school, with eager anticipation as we were going to see the Challenger liftoff and take a teacher, Sharon McAuliffe, into space. Many of you know the tragic events of that day and the findings of the NASA Commission. To review the details, 73 seconds after liftoff, the primary and secondary O-rings on the solid-state boosters failed and caused an explosion due to excess hydrogen gas and premature fuel ignition. This resulted in the Challenger being torn apart. The reason the O-rings failed is blamed primarily on the weather. That January day was only about 30 degrees at launch time,[4] much colder than any space shuttle launch NASA had attempted before. The cold weather

[3] The drive between my home and the venue takes about 20 minutes on the average day.

[4] The launch control parameters for the Space Shuttle were a 24-hour average temperature of 41 degrees F and not warmer than 99 degrees F for 30 consecutive minutes.

created a problem, because NASA personnel planned assuming a linear relationship between the air temperature and O-ring performance, but instead that relationship was nonlinear and the O-ring was actually much more brittle and ineffective than preplanning had anticipated. This was a tragic lesson to learn as it cost the lives of many remarkable people. After this incident, NASA changed a number of procedures in an effort to make space flight safer.[5]

In statistics, there are several terms of art to describe the shape or distribution of your data. The terms are: mean, standard deviation, skewness, and kurtosis. At this point, the important facts to understand and keep in mind are that: (1) there are often nonlinear relationships in real-world data; (2) as the data size increases, you are able to see those relationships more clearly; and, more frequently (3) nonlinear relationships can have a very significant effect on your results if you do not understand and control for them.

In Part One of this book, the focus is on the technology aspects of creating an analytical environment for data mining, machine learning, and working with big data and the trade-offs that result from certain technology choices. In Part Two, the focus is on algorithms and methods that can be used to gain information from your data. In Part Three, case studies show how, by utilizing these new technology advances and algorithms, organizations were able to make big impacts. Part Three also illustrates that using high-performance computing, analytical staff productivity went up in meaningful ways.

[5] I got to witness the safety protocol in action. I was at Cape Canaveral for the launch of STS-111, but it was scrubbed with less than an hour before liftoff.

PART ONE

The Computing Environment

With data collection, "the sooner the better" is always the best answer.

— Marissa Mayer

Data mining is going through a significant shift with the volume, variety, value and velocity of data increasing significantly each year. The volume of data created is outpacing the amount of currently usable data to such a degree that most organizations do not know what value is in their data. At the same that time data mining is changing, hardware capabilities have also undergone dramatic changes. Just as data mining is not one thing but a collection of many steps, theories, and algorithms, hardware can be dissected into a number of components. The corresponding component changes are not always in sync with this increased demand in data mining, machine learning, and big analytical problems.

The four components of disk, memory, central processing unit, and network can be thought of as four legs of the hardware platform stool. To have a useful stool, all the legs must be of the same length or users will be frustrated, stand up, and walk away to find a better stool; so too must the hardware system for data mining be in balance in regard to the components to give users the best experience for their analytical problems.

Data mining on any scale cannot be done without specialized software. In order to explain the evolution and progression of the hardware, there needs to be a small amount of background on the traditional interaction between hardware and software. Data mining software packages are discussed in detail in Part One.

In the past, traditional data mining software was implemented by loading data into memory and running a single thread of execution over the data. The process was constrained by the amount of memory available and the speed of a processor. If the process could not fit entirely into memory, the process would fail. The single thread of execution also failed to take advantage of multicore servers unless multiple users were on the system at the same time.

The main reason we are seeing dramatic changes in data mining is related to the changes in storage technologies as well as computational capabilities. However, all software packages cannot take advantage of current hardware capacity. This is especially true of the distributed computing model. A careful evaluation should be made to ensure that algorithms are distributed and effectively leveraging all the computing power available to you.

CHAPTER 1

Hardware

I am often asked what the best hardware configuration is for doing data mining. The only appropriate answer for this type of question is that it depends on what you are trying to do. There are a number of considerations to be weighed when deciding how to build an appropriate computing environment for your big data analytics.

STORAGE (DISK)

Storage of data is usually the first thing that comes to mind when the topic of big data is mentioned. It is the storage of data that allows us to keep a record of history so that it can be used to tell us what will likely happen in the future.

A traditional hard drive is made up of platters which are actual disks coated in a magnetized film that allow the encoding of 1s and 0s that make up data. The spindles that turn the vertically stacked platters are a critical part of rating hard drives because the spindles determine how fast the platters can spin and thus how fast the data can be read and written. Each platter has a single drive head; they both move in unison so that only one drive head is reading from a particular platter.

This mechanical operation is very precise and also very slow compared to the other components of the computer. It can be a large

contributor to the time required to solve high-performance data mining problems.

To combat the weakness of disk speeds, disk arrays[1] became widely available, and they provide higher throughput. The maximum throughput of a disk array to a single system from external storage subsystems is in the range of 1 to 6 gigabytes (GB) per second (a speedup of 10 to 50 times in data access rates).

Another change in disk drives as a response to the big data era is that their capacity has increased 50% to 100% per year in the last 10 years. In addition, prices for disk arrays have remained nearly constant, which means the price per terabyte (TB) has decreased by half per year.

This increase in disk drive capacity has not been matched by the ability to transfer data to/from the disk drive, which has increased by only 15% to 20% per year. To illustrate this, in 2008, the typical server drive was 500 GB and had a data transfer rate of 98 megabytes per second (MB/sec). The entire disk could be transferred in about 85 minutes (500 GB = 500,000 MB/98 MB/sec). In 2013, there were 4 TB disks that have a transfer rate of 150 MB/sec, but it would take about 440 minutes to transfer the entire disk. When this is considered in light of the amount of data doubling every few years, the problem is obvious. Faster disks are needed.

Solid state devices (SSDs) are disk drives without a disk or any moving parts. They can be thought of as stable memory, and their data read rates can easily exceed 450 MB/sec. For moderate-size data mining environments, SSDs and their superior throughput rates can dramatically change the time to solution. SSD arrays are also available, but SSDs still cost significantly more per unit of capacity than hard disk drives (HDDs). SSD arrays are limited by the same external storage bandwidth as HDD arrays. So although SSDs can solve the data mining problem by reducing the overall time to read and write the data, converting all storage to SSD might be cost prohibitive. In this case, hybrid strategies that use different types of devices are needed.

Another consideration is the size of disk drives that are purchased for analytical workloads. Smaller disks have faster access times, and

[1] A disk array is a specialized hardware storage that provides larger storage capacity and data access because of its specialized implementation. NetApp and EMC are two major vendors of disk arrays.

there can be advantages in the parallel disk access that comes from multiple disks reading data at the same time for the same problem. This is an advantage only if the software can take advantage of this type of disk drive configuration.

Historically, only some analytical software was capable of using additional storage to augment memory by writing intermediate results to disk storage. This extended the size of problem that could be solved but caused run times to go up. Run times rose not just because of the additional data load but also due to the slower access of reading intermediate results from disk instead of reading them from memory. For a typical desktop or small server system, data access to storage devices, particularly writing to storage devices, is painfully slow. A single thread of execution for an analytic process can easily consume 100 MB/sec, and the dominant type of data access is sequential read or write. A typical high-end workstation has a15K RPM SAS drive; the drive spins at 15,000 revolutions per minute and uses the SAS technology to read and write data at a rate of 100 to 150 MB/sec. This means that one or two cores can consume all of the disk bandwidth available. It also means that on a modern system with many cores, a large percentage of the central processing unit (CPU) resources will be idle for many data mining activities; this is not a lack of needed computation resources but the mismatch that exists among disk, memory, and CPU.

CENTRAL PROCESSING UNIT

The term "CPU" has had two meanings in computer hardware. CPU is used to refer to the plastic and steel case that holds all the essential elements of a computer. This includes the power supply, motherboard, peripheral cards, and so on. The other meaning of CPU is the processing chip located inside the plastic and steel box. In this book, CPU refers to the chip.

The speed of the CPU saw dramatic improvements in the 1980s and 1990s. CPU speed was increasing at such a rate that single threaded software applications would run almost twice as fast on new CPU versions as they became available. The CPU speedup was described by Gordon Moore, cofounder of Intel, in the famous Moore's law, which is an observation that the number of transistors and integrated circuits

that are able to be put in a given area doubles every two years and therefore instructions can be executed at twice the speed. This trend in doubling CPU speed continued into the 1990s, when Intel engineers observed that if the doubling trend continued, the heat that would be emitted from these chips would be as hot as the sun by 2010. In the early 2000s, the Moore's law free lunch was over, at least in terms of processing speed. Processor speeds (frequencies) stalled, and computer companies sought new ways to increase performance. Vector units, present in limited form in x86 since the Pentium MMX instructions, were increasingly important to attaining performance and gained additional features, such as single- and then double-precision floating point.

In the early 2000s, then, chip manufacturers also turned to adding extra threads of execution into their chips. These multicore chips were scaled-down versions of the multiprocessor supercomputers, with the cores sharing resources such as cache memory. The number of cores located on a single chip has increased over time; today many server machines offer two six-core CPUs.

In comparison to hard disk data access, CPU access to memory is faster than a speeding bullet; the typical access is in the range of 10 to 30 GB/sec. All other components of the computer are racing to keep up with the CPU.

Graphical Processing Unit

The graphical processing unit (GPU) has gotten considerable publicity as an unused computing resource that could reduce the run times of data mining and other analytical problems by parallelizing the computations. The GPU is already found in every desktop computer in the world.

In the early 2000s, GPUs got into the computing game. Graphics processing has evolved considerably from early text-only displays of the first desktop computers. This quest for better graphics has been driven by industry needs for visualization tools. One example is engineers using three-dimensional (3D) computer-aided design (CAD) software to create prototypes of new designs prior to ever building them. An even bigger driver of GPU computing has been the consumer video game industry, which has seen price and performance trends similar to the rest of the consumer computing industry. The relentless

drive to higher performance at lower cost has given the average user unheard-of performance both on the CPU and the GPU.

Three-dimensional graphics processing must process millions or billions of 3D triangles in 3D scenes multiple times per second to create animation. Placing and coloring all of these triangles in their 3D environment requires a huge number of very similar calculations. Initially, 3D graphics were done using a fixed rendering pipeline, which took the 3D scene information and turned it into pixels that could be presented to the user in a video or on the screen. This fixed pipeline was implemented in hardware, with various parts of the GPU doing different pieces of the problem of turning triangles into pixels. In the early 2000s, this fixed pipeline was gradually replaced by generalized software shaders, which were miniprograms that performed the operations of the earlier fixed hardware pipeline.

With these shaders, high-performance computing folks noticed that the floating-point coordinates and colors could look an awful lot like physics or chemistry problems if you looked at them just right. The more hardcore hacker types started creating graphics problems that looked a lot like nonsense except that the underlying calculations being done solved hard problems remarkably fast. The performance gains got noticed, and computing frameworks, which used the GPUs for doing nongraphics calculations, were developed. These calculations are the same type needed for data mining.

GPUs are a green field. Historically the ability to develop code to run on the GPU was restrictive and costly. Those programming interfaces for developing software that takes advantage of GPUs have improved greatly in the last few years. Software has only started to take advantage of the GPU, and it will be several years before the computations needed for data mining are efficiently delegated to the GPU for execution. When that time comes, the speedup in many types of data mining problems will be reduced from hours to minutes and from minutes to seconds.

MEMORY

Memory, or random access memory (RAM) as it is commonly referred to, is the crucial and often undervalued component in building a data mining platform. Memory is the intermediary between the storage of

data and the processing of mathematical operations that are performed by the CPU. Memory is volatile, which means that if it loses power, the data stored in it is lost.

In the 1980s and 1990s, the development of data mining algorithms was very constrained by both memory and CPU. The memory constraint was due to the 32-bit operating systems, which allow only 4 GB of memory to be addressed. This limit effectively meant that no data mining problem that required more than 4 GB of memory[2] (minus the software and operating system running on the machine) could be done using memory alone. This is very significant because the data throughput of memory is typically 12 to 30 GB/sec, and the fastest storage is only around 6 GB/sec with most storage throughput being much less.

Around 2004, commodity hardware (Intel and AMD) supported 64-bit computing. At the same time operating systems became capable of supporting larger amounts of memory, the actual price of memory dropped dramatically. In 2000, the average price of 1 MB of RAM was $1.12. In 2005, the average price was $0.185; and in 2010, it was $0.0122.

With this support of 64-bit computing systems that can address up to 8 TB of memory and the drop in memory prices, it was now possible to build data mining platforms that could store the entire data mining problem in memory. This in turn produced results in a fraction of the time.

Data mining algorithms often require all data and computation to be done in memory. Without external storage, the increase in virtual and real address space as well as the dramatic drop in the price of memory created an opportunity to solve many data mining problems that previously were not feasible.

To illustrate this example, consider a predictive modeling problem that uses a neural network algorithm. The neural network will perform an iterative optimization to find the best model. For each iteration, it will have to read the data one time. It is not uncommon for neural networks to make thousands of passes through the data to find

[2] The largest integer value that 32-bit operating systems can use to address or reference memory is 2^{32}–1, or 3.73 GB, of memory.

the optimal solution. If these passes are done in memory at 20 GB/sec versus on disk at 1 GB/sec, a problem that is only 10 seconds to solve in memory will be more than 3 minutes to solve using disk. If this scenario is repeated often, the productivity of the data miner plummets. In addition to the productivity of the human capital, if the data mining processes relied on disk storage, the computation would take many times longer to complete. The longer a process takes to complete, the higher the probability of some sort of hardware failure. These types of failure are typically unrecoverable, and the entire process must be restarted.

Memory speeds have increased at a much more moderate rate than processor speeds. Memory speeds have increased by 10 times compared to processor speeds, which have increased 10,000 times. Disk storage throughput has been growing at an even slower rate than memory. As a result, data mining algorithms predominantly maintain all data structures in memory and have moved to distributed computing to increase both computation and memory capacity. Memory bandwidth is typically in the 12 to 30 GB/sec range, and memory is very inexpensive. High-bandwidth storage maxes out in the 6 GB/sec range and is extremely expensive. It is much less expensive to deploy a set of commodity systems with healthy amounts of memory than to purchase expensive high-speed disk storage systems.

Today's modern server systems typically come loaded with between 64 GB and 256 GB of memory. To get fast results, the sizing of memory must be considered.

NETWORK

The network is the only hardware component that is always external to the computer.[3] It is the mechanism for computers to communicate to other computers. The many protocols and standards for network communication will not be discussed here beyond the very limited details in this section.

The network speed should be a factor only for a distributed computing environment. In the case of a single computer (workstation or

[3] Storage can sometimes be external in a storage area network (SAN).

server), the data, memory, and CPU should all be local, and performance of your analytical task will be unaffected by network speeds.

The standard network connection for an analytical computing cluster is 10 gigabit Ethernet (10 GbE), which has an upper-bound data transfer rate of 4 gigabytes per second (GB/sec). This data transfer rate is far slower than any of the other essential elements that have been discussed. Proprietary protocols like Infiniband® give better data throughput but still do not match the speed of the other components. For this reason, it is very important to minimize usage of the network for data movement or even nonessential communication between the different nodes in the computing appliance.

It is this network speed bottleneck that makes parallelization of a number of the data mining algorithms so challenging. Considerable skill in the software infrastructure, algorithm selection, and final implementation is required to fit a model efficiently and precisely using one of many algorithms while not moving any data and limiting communication between computers.

The network speed of your high-performance data mining platform will be important if you have a distributed computing environment. Because the network data transfer rate is much slower than other components, you must consider the network component when evaluating data mining software solutions.

CHAPTER 2

Distributed Systems

If you look back before electronic computers, the term "computer" meant a person who performed numerical calculations. Because roomfuls of these computers were in use for most of human history, arguably distributed (i.e., multiperson) computing dates back much farther than conventionally asserted.

Modern distributed computing arguably derives from the 1990s efforts to build "Beowulf" clusters of computers and from early ad hoc computing.[1] Beowulf clusters of computers were standard server or even desktop computers that were networked. The software on the computers then communicated closely with each other over the network to split the work to be done across the computers. Early ad hoc computing efforts, including distributed.net and SETI@Home, used idle home computers to search for cryptography keys and alien radio signals in radio astronomy data. Neither of these projects had the computing power in-house to solve the problem, nor sufficient budget to do it. However, the problems were simple enough that a large number

[1] I tried with limited success to build a Beowulf cluster in the late 1990s with cast-off computers.

of computers could make considerable progress with minimal coordination between them. Home computers running the distributed.net or SETI@Home software contacted the main server to get a chunk of work to be done (keys to check or radio signal data to examine).

Classical supercomputers were very large, very expensive machines that contained specialized fast hardware and processors. These supercomputers had contained in their specialized hardware features such as multiple central processing units and vector hardware. Vector hardware is used to perform identical operations on multiple pieces of data. For instance, a two-dimensional vector has components X and Y. Adding a set of vectors is done by components, so providing a single instruction that adds both X and Y simultaneously can result in doubling processing speed.

The cost of hardware has dropped dramatically. Both high-core/large-memory (massively parallel processing [MPP] systems) and clusters of moderate systems allow much larger data mining projects to be feasible. For big data analytics, the only solution is to move the analytics to the data; it is impractical to move the data to the analytics because of the time required for the data transfer.

Harder problems—problems that consume much larger volumes of data, with much higher numbers of variables—may now be considered for analysis. Cluster computing can be divided into two main groups of distributed computing systems. The first is the database, which has been a fixture in data centers for decades. The second is the distributed file system, which is currently dominated by Hadoop.

DATABASE COMPUTING

The relational database management system (RDBMS) or simply database has been around since the 1970s and until recently has been the most common place to store data generated by organizations. A number of huge vendors as well as open source projects provide database systems. Traditional RDBMSs are designed to support databases that are much larger than the memory or storage available on a single computer. From those early days, there are now a number of different databases that serve special purposes for organizations around high-performance data mining and big data analytics.

In-memory databases (IMDBs) were developed starting in the 1990s. IMDBs are now a popular solution used to accelerate mission-critical data transactions for finance, e-commerce, social media, information technology, and other industries. The idea behind IMDB technology is straightforward—holding data in memory and not on disk increases performance. However, there are notable drawbacks to IMDBs, namely the increased cost and volatility of memory.

MPP (massively parallel processing) databases began to evolve from traditional DBMS technologies in the 1980s. MPP databases are meant to serve many of the same operational, transactional, and analytic purposes as the previous generation of commercial databases but offer performance, availability, and scalability features designed to handle large volumes of data while utilizing standard user interfaces. MPP databases are positioned as the most direct update for organizational enterprise data warehouses (EDWs). The technology behind MPP databases usually involves clusters of commodity or specialized servers that hold data on multiple hard disks.

A big advantage of database computing is the time saved in moving the data to the analytics. In a truly big data problem, the time to move the data and the hardware resources needed to process it efficiently once it is moved make the strategy inefficient. Using software that can move the analytics to the data and process it in place leveraging the large computational resources that a distributed database provides will lead to faster models and shorter run times when compared to nondistributed computing systems.

FILE SYSTEM COMPUTING

There are many options in choosing a platform for file system computing, but the market is rapidly consolidating on Hadoop with its many distributions and tools that are compatible with its file system, such as Hive, MapReduce, and HBase.

Hadoop was created by Doug Cutting and Mike Cafarella. (See Figure 2.1.) The name "Hadoop" is not an acronym or a reference to anything really. It came from the name that Cutting's son had given to a stuffed yellow elephant. It was initially developed in 2004 based

on the work Cutting and Cafarella had done on Nutch[2] and a paper published by Google that introduced the MapReduce paradigm for processing data on large clusters. In 2008, Hadoop had become a top-level Apache project and was being used by several large data companies such as Yahoo!, Facebook, and *The New York Times*.

A current trend is to store all available data in Hadoop. Hadoop is attractive because it can store and manage very large volumes of data on commodity hardware and can expand easily by adding hardware resources with incremental cost. Traditionally, systems were sized based on expected volumes of data, without the expectation that data would accumulate in perpetuity. Hadoop has made large-scale accumulation of data feasible and potentially is a significant differentiator for competitive advantage. Those who can exploit the value of historical data successfully can gain a huge advantage in the future.

Hadoop is becoming the front-runner for housing large volumes of historical data. However, this data is rarely (actually probably never) in an appropriate form for data mining.

Also, for many problems, other data repositories are used to augment the data in Hadoop. For example, credit card transactions may be stored in Hadoop, but the cardholder account information may be

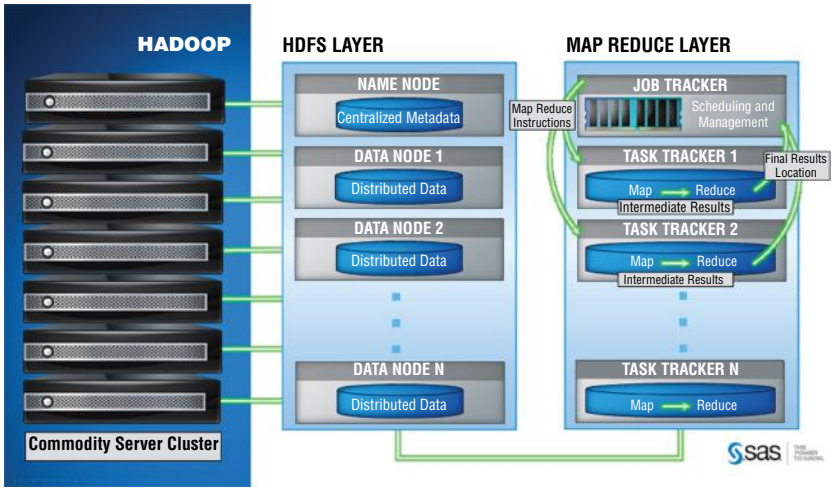

Figure 2.1 Graphical illustration of a Hadoop System

[2] Nutch is an Apache open source web crawling project.

stored and maintained in a traditional database. The data survey step includes identifying the data and the data repositories that will be used in the modeling process. Then the data must be combined, summarized, and stored for data mining. This work normally will be done in Hadoop because the computational cost per unit is lower than MPP or RDBMS. This hybrid approach to storing data is likely to be common practice for many years until either Hadoop matures to the level of databases or another technology emerges that makes both databases and Hadoop less desirable.

Cloud computing, because it is able to rapidly provision new servers or to bring down servers no longer needed, can be used in either capacity. However, not all of the high-end tweaks, such as specialized networking hardware, are available. Their flexibility allows one to rapidly change between simple ad hoc computing and coordinated computing and even to mix the models on the fly.

CONSIDERATIONS

Here are some questions to consider as you consider your high-performance data mining platform.

- What is the size of your data now?
- What is the anticipated growth rate of your data in the next few years?
- Is the data you are storing mostly structured data or unstructured data?
- What data movement will be required to complete your data mining projects?
- What percentage of your data mining projects can be solved on a single machine?
- Is your data mining software a good match for the computing environment you are designing?
- What are your users' biggest complaints about the current system?

Figure 2.2 compares a number of big data technologies. The figure highlights the different types of systems and their comparative strengths

	In-Memory Database	MPP Database	Big Data Appliance	Hadoop	NoSQL Database
Consistent	●	●	●	▲	▲
Available	●	●	●	▲	▲
Fault tolerant	●	●	▲	●	●
Suitable for real-time transactions	●	●	●	◆	◆
Suitable for analytics	▲	▲	●	●	◆
Suitable for extremely big data	◆	▲	▲	●	●
Suitable for unstructured data	◆	◆	▲	●	●

Figure 2.2 Comparison of Big Data Technologies

and weaknesses. The purchasing decision for the data mining computing platform will likely be made by the IT organization, but it must first understand the problems being solved now and the problems that are needed to be solved in the near future. The consideration of platform trade-offs and the needs of the organization, all shared in a transparent way, will lead to the best outcome for the entire organization and the individual stakeholders.

In Figure 2.2 the symbols have the following meaning:

- Circle: Meets widely held expectations.
- Triangle: Potentially meets widely held expectations.
- Diamond: Fails to meet widely held expectations

Big data/data mining projects must address how data moves through the entire end-to-end process.

The computing environment is critical to success: Your computing environment comprises four important resources to consider: network, disk, central processing units (CPUs), and memory. Time to solution goals, expected data volumes, and budget will direct your decisions

regarding computation resources. Appropriate computing platforms for data analysis depend on many dimensions, primarily the volume of data (initial, working set, and output data volumes), the pattern of access to the data, and the algorithm for analysis. These will vary depending on the phase of the data analysis.

CHAPTER 3

Analytical Tools

Information is the oil of the 21st century, and analytics is the combustion engine.

—Peter Sondergaard

When faced with a business challenge, people need tools to overcome issues and find ways to create value. In the analytics forum, frequently that begins very early on with deciding what tool should be used. This is the third consideration before work is begun. The first two, where to store the data and how to prepare it for analysis, were discussed in Chapters 1 and 2. This chapter details some tools that are commonly used. Some I have great familiarity with and, in fact, contributed to their development. Others I have used as tools, just as you or your teams will, and a few I have not used personally, but they also should be considered. I have tried to provide a balanced assessment of the strengths and weakness of each tool.

WEKA

Weka (Waikato Environment for Knowledge Analysis) is an open source data mining offering, fully implemented in Java, and primarily developed at the University of Waikato, New Zealand. Weka is notable for its broad range of extremely advanced training algorithms, its

work flow graphical user interface (GUI), and its incorporation of data visualization tools. Weka allows users access to its sophisticated data mining routines through a GUI designed for productive data analysis, a command line interface, and a Java application programming interface (API). However, Weka does not scale well for big data analytics, as it is limited to available RAM resources, typically on one machine. Users with 64-bit operating systems will have access to much larger RAM resources, but Weka's documentation directs users to its data preprocessing and filtering algorithms to sample big data before analysis. As many of the most powerful algorithms available in Weka are unavailable in other environments without custom software development, sampling may be the best practice for use cases necessitating Weka. In recent releases, Weka has made strides toward multithreading and simple multitasking. In Weka 3.6, some classifiers can train models on multiple cross-validation folds simultaneously, and the Weka server deployment allows for running data mining tasks concurrently on one machine or a cluster. Weka routines also provide advanced analytics components for two Java-based big data analysis environments: Pentaho and MOA (Massive Online Analysis).

Weka is a good option for people who may not have Java programming experience and need to get started quickly to prove value. I used Weka during a survey course in graduate school about data mining tools. It was a few years after Weka was completely rewritten to use Java. The interface provided big productivity gains for me, because I was not a very experienced Java programmer. I was able to analyze data with far less start-up investment in the programming language.

JAVA AND JVM LANGUAGES

For organizations looking to design custom analytics platforms from scratch, Java and several other languages that run on the Java Virtual Machine (JVM) are common choices. For large, concurrent, and networked applications, Java presents considerable development advantages over lower-level languages, without excessive performance sacrifices, especially in the realm of big data analytics. While languages that execute directly on native hardware, particularly FORTRAN and C, continue to

outperform Java in RAM and CPU-bound calculations, technological advances in the Java platform have brought its performance nearly in line with native languages for input/output and network-bound processes, like those at the core of many open source big data applications. Probably the most recognizable Java-based big data environment, Apache Hadoop, beat out several other technologies in the 2008 and 2009 TeraByte Sort Benchmark. It was the first Java-based technology to win the well-respected bench marking contest.

Java's advantages and disadvantages as a general-purpose programming language have been widely discussed in many forums, and will be addressed only briefly here. Developers of analytic applications often avoided Java in the past, because the exhaustive runtime library was unnecessary for numerical routines and the memory management by the JVM platform caused unacceptable slowdowns, to name just two reasons. As analytic applications grew dramatically in scale and complexity, development time became a more serious concern, and memory and CPU-bound performance became less so. In the specific context of building data mining applications, Java's strengths arise from its development efficiency: its rich libraries, many application frameworks, inherent support for concurrency and network communications, and a preexisting open source code base for data mining functionality, such as the Mahout and Weka libraries. Although at some cost to performance, development advantages also stem from the JVM platform. In addition to portability, the JVM provides memory management, memory profiling, and automated exception handling.

Scala and Clojure are newer languages that also run on the JVM and are used for data mining applications. Scala is an open source language developed in Switzerland at the École Polytechnique Fédédale de Lausanne. It was first released in 2003 and aims to be an improvement on Java.

Scala is used to construct data mining applications because it encompasses the development efficiencies of Java—through interoperability with the Java Runtime Environment—and adds functional programming, all with a more concise syntax. Scala allows developers to switch between functional programming and object-oriented (OO) programming paradigms, whereas Java's grammar and syntax tend to enforce

OO design. Functional languages are regarded positively for data mining applications because they handle memory differently from OO and procedural languages. For instance, a major concern in procedural and OO multithreaded applications is preventing multiple threads from changing the same variable at the same time. Functional programming languages avoid this by never changing variables. Even though it is a less common programming paradigm, functional programming is a valuable tool that should not be discounted. Scala is gaining in popularity and has been used to implement major open source projects like Akka and Spark. Akka is a toolkit for building large, parallel, and distributed applications on the JVM, and Spark is a high-performance data mining environment from the University of California Berkeley AMP Lab. Scala is also used commercially for various purposes by FourSquare, the Guardian, LinkedIn, Novell, Siemens, Sony, Twitter, and others.

Clojure was written by Rich Hickey and released in 2007. It is a functional programming language and a dialect of the Lisp programming language, with additional scripting and concurrency features. Although not designed specifically for big data applications, these intrinsic characteristics make Clojure an excellent candidate for building data analysis software. Clojure defaults to immutable data structures, removing an entire category of thread safety design concerns as compared to procedural and OO languages. Clojure treats data in a very holistic manner, presenting developers many convenient ways to process its comprehensive set of data structures and inheriting the concept of code as data from Lisp. This approach to sequences of data eliminates another class of common programming errors related to boundary conditions, specifically in comparison to C and FORTRAN. Clojure also inherits its simple syntax, lazy evaluation, and highly effective macro facility from Lisp. The macro facility allows for the creation of domain-specific languages (DSLs), and DSLs for SQL (ClojureQL) and Hadoop (Cascalog) integration have already been established. When Clojure's unique strengths are combined with those of the JVM platform and access to Java libraries, it becomes a formidable general-purpose programming asset. The first adopters of Clojure have generally been smaller, technologically or analytically advanced companies and start-ups. However, larger corporations and institutions, such as Citigroup and the Max Planck Institute, have reported using Clojure as well.

R

R is an open source fourth-generation programming language designed for statistical analysis. R grew from the commercial S language, which was developed at Bell Laboratories starting in the late 1970s. R was ported from S and SPLUS—another commercial implementation now licensed by TIBCO—by researchers at the University of Auckland, New Zealand, and first appeared in 1996.

R is very popular in the academic community, where I was first exposed to it. I used it to experiment with new analysis that I didn't have time or expertise to code myself and to complete homework assignments. At the time the software also had wonderful graphing functionality superior to others I had used. I leveraged this graphing facility in graduate school and while working at the U.S. Census Bureau. Although these graphing facilities are still present in R, it has lost its competitive advantage to other software.

R has grown to a position of prominence in the broad and growing data science community, enjoying wide usage in academic settings and increasing acceptance in the private sector. Large companies, such as Bank of America, Google, Intercontinental Hotels, Merck, Pfizer and Shell, are known to use R for a variety of purposes. The September 2013 TIOBE general survey of programming languages put R in 18th place in overall development language popularity, far outranking S and S-SPLUS, and in close proximity to commercial solutions like SAS (at 21st) and MATLAB (at 19th) (TIOBE, 2013).[1] R is also extremely customizable. There are thousands of extensions for R, up from about 1,600 in 2009. Extension packages incorporate everything from speech analysis, to genomic science, to text mining into R's baseline analysis functionality. R also boasts impressive graphics, free and polished integrated development environments (IDEs), programmatic access to and from many general-purpose languages, interfaces with popular proprietary analytics solutions including MATLAB and SAS, and even commercial support from Revolution Analytics.

[1] The TIOBE programming community index is a measure of popularity of programming languages, calculated from a number of search engine results for queries containing the name of the language.

R is experiencing a rapid growth in popularity and functionality, and many of the memory issues that made dealing with larger data difficult in previous versions of R have been resolved. Significant development work is currently under way to circumvent remaining memory limitations through cluster computing and interfaces to Hadoop. R's short update cycle has made this work available to R consumers through official packages like snow, multicore, and parallel, and younger projects like RHadoop and RHIPE. Aside from Rhadoop and RHIPE, these packages are not meant for true big data analysis but for CPU- and memory-intensive calculations of the "embarrassingly parallel" variety, often executed through R's lapply() construct, where a function is applied to all the elements of a list.

Snow is used to run R calculations on a cluster of computers. It uses sockets, MPI, PVM or NetWorkSpaces to communicate between cluster nodes.[2] The snow package employs traditional master-worker parallelization architecture, holds results in memory on the master node, and requires arguments to snow function calls to fit in memory. Thus, being memory bound, the package is used primarily for CPU-intensive computations, such as simulations, bootstrapping, and certain machine learning algorithms.

Multicore is a simple-to-install-and-implement package that splits sequential R processes running on a single POSIX-compliant machine (OS X, Linux, or UNIX) into a multithreaded process on the same machine. Processes executed simultaneously using multicore are limited to single-machine RAM and other shared memory constraints.

Parallel is a parallel execution package that comes standard in R 2.14 and higher. It expands out-of-the-box parallel execution to the cluster level for POSIX-compliant operating systems and to the multiprocessor level for Windows. Parallel shares a great deal of syntax,

[2] PVM (Parallel Virtual Machine) is a software package that permits a heterogeneous collection of Unix and/or Windows computers connected together by a network to be used as a single large parallel computer. Thus, large computational problems can be solved more cost effectively by using the aggregate power and memory of many computers. The software is very portable. NetWorkSpaces (NWS) is a powerful, open source software package that makes it easy to use clusters from within scripting languages like Matlab, Python, and R. While these scripting languages are powerful and flexible development tools, their inherent ease of use can, in many cases, cause natural limitations.

functionality, and limitations with both snow and multicore, and can use snow cluster objects.

Rhadoop and **Rhipe** allow programmatic access to Hadoop, using the R language. Map and reduce tasks can be run from R on a Hadoop cluster.

PYTHON

From its inception in the late 1980s, Python was designed to be an extensible, high-level language with a large standard library and a simple, expressive syntax. Python can be used interactively or programmatically, and it is often deployed for scripting, numerical analysis, and OO general-purpose and Web application development. Python is an interpreted language and typically performs computationally intensive tasks slower than native, compiled languages and JVM languages. However, Python programs can call compiled, native FORTRAN, C and C++ binaries at run time through several documented APIs. Python programs can be multithreaded, and like JVM languages, the Python platform conducts memory management and exception handling, relieving the developer from that responsibility. The Python language has also been ported to the JVM platform, in a project called Jython.

Python's general programming strengths combined with its many database, mathematical, and graphics libraries make it a good fit for projects in the data exploration and data mining problem domains. Python's inherent file handling and text manipulation capabilities, along with its ability to connect with most SQL and NoSQL databases, allow Python programs to load and process raw and tabular data with relative ease. Mathematical and statistical models can be implemented using the Scipy and Numpy libraries, which have strong support for linear algebra, numerical analysis, and statistical operations. Newer Python libraries, like Orange and Pattern, provide high-level data mining APIs, and the Matplotlib library can generate complex and nuanced data visualizations. Python can also be used to write map and reduce directives for MapReduce jobs using the Hadoop Streaming utility.

Python recently has become much more used in the data mining and data science communities with the maturity of the scikit-learn toolkit. The toolkit is currently on release 0.14 but offers a number of

useful algorithms and manipulation techniques. Python's scikit does not have any user interface beyond the programming IDE, which allows users flexibility but makes the initial learning curve steep and decreases productivity for routine tasks.

SAS

SAS is the leading analytical software on the market. In the 2013 IDC report ("Worldwide Business Analytics Software"), SAS not only had a significant market share but held more market share than the next 16 vendors combined. The SAS System was first sold in 1976, and two of the four cofounders are still active in the company—Dr. Jim Goodnight and John Sall. The SAS System is divided into a number of product areas including statistics, operations research, data management, engines for accessing data, and business intelligence (BI). For the topic of this book, the most relevant products are SAS/STAT®, SAS® Enterprise Miner™, and the SAS text analytics suite. SAS has been rated as the leading vendor in predictive modeling and data mining by the most recent Forrester Wave and Gartner Magic Quadrant. These ratings reveal the breadth and depth of the capability of the software. In my experience, there is no analytical challenge that I have not been able to accomplish with SAS. Having worked for the company for almost a decade, I have a much better understanding of the meticulous work and dedication of the development team and the tremendous lengths they go to ensure software quality, meet customer requests, and anticipate customers' future needs.

The SAS system is divided into two main areas: procedures to perform an analysis and the fourth-generation language that allows users to manipulate data. This is known as the DATA Step.

One of the unique achievements of SAS is the backward compatibility that is maintained release after release. I saw this firsthand at the U.S. Census Bureau. We had several SAS programs that were well over 15 years old, and they provided the same verifiable results with each update to SAS. This was essential because frequently the original author of the computer programs had moved on to other departments or in some cases no longer worked at the Census Bureau.

SAS uses a comprehensive scheme to provide the best computational performance while being flexible to individual customer environments. The procedure, or PROC as it is commonly referred to, is a module built for a specific analysis, like the REG procedure for building regression models or the UNIVARIATE procedure for doing descriptive statistics on a single variable. In a simple example, let us take a task of computing a mean for a given variable.

This can be trivially accomplished in any programing language because the mean is simple: $\sum_1^n i/n$. This can be computed in the data step with code like this:

```
Data a;
       Set b end=last;
Retain x;
Sum+x;
If last then mean=sum/_n_;
Run;
```

This example shows where a new data set that will be created from data set b. The retain statement tells SAS to keep the values when moving to a new observation (row); the if statement tells SAS that if this is the last observation in the data set, divide the sum by the number of observations, which is stored in the automatic variable _n_. This code is not difficult to write and explain, but there is an alternative approach that is better for SAS users:

```
Proc means data=b;
       Var x;
Run;
```

Besides being shorter, this code is also easier to read, uses code written for a specific purpose instead of general-purpose code, and I did not need to make a copy of the data just to add one new column. The procedure also handles and documents a number of behaviors that become very important when you develop your own software (e.g., working with ties, missing values, or other data quality issues). SAS has 82 procedures in the SAS/STAT product alone. Literally hundreds of procedures perform specific analyses. A major advantage of SAS over other software packages is the documentation. SAS/STAT has over 9,300 pages of documentation and Enterprise Miner has over 2,000 pages.

The SAS system processes data in memory when possible and then uses system page files to manage the data at any size efficiently. In the last few years, SAS has made major changes in its architecture to take better advantage of the processing power and falling price per FLOP (floating point operations per second) of modern computing clusters.

PART TWO

Turning Data into Business Value

This second part of the book shifts from the storage of data, preparation of data, hardware considerations, and the software tools needed to perform data mining to the methodology, algorithms, and approaches that can be applied to your data mining activities. This includes a proven method for effective data mining in the sEMMA approach, discussion about the different types of predictive modeling target models, and understanding which methods and techniques are required to handle that data effectively. From my experience, most business environments use several people to perform each of the tasks. In larger organizations, the tasks might be split across many groups and organizationally only meet at the executive level of the organization. A quote from Shakespeare that I have always appreciated to introduce this topic is:

> *If you can look into the seeds of time, And say which grain will grow and which will not, Speak then to me, who neither beg nor fear Your favours nor your hate.*
>
> —***Macbeth***, Act 1, Scene 3

This verse shows an appreciation and understanding from hundreds of years ago that those people who can predict future behavior have a distinct advantage regardless of the venue. In sports, this is often call field presence: the talent some players have to effectively anticipate where the ball will be played and be there before the ball arrives. On Wall Street, fortunes are won by correctly anticipating the movement of the market in advance. In your business, correctly anticipating customer behavior or reducing future expenditures has a real impact. The goal in this section is to explain a methodology for predictive modeling. This process has been in place for over a decade and proven useful for thousands and thousands of users. I also discuss how this methodology applies in the big data era, which reduces or makes optional the need for data sampling.

Part Two discusses the types of target models, their characteristics, and information about their specific uses in business.

In addition, I discuss a number of predictive modeling techniques to help you understand the fundamental ideas behind these techniques, their origins, how they differ, and some of their drawbacks.

Finally, I present a set of methods that you might be less familiar with that address more modern methods for analysis or analysis on specific types of data.

CHAPTER 4

Predictive Modeling

I never guess. It is a capital mistake to theorize before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts.

—Sir Arthur Conan Doyle,
author of Sherlock Holmes stories

Predictive modeling is one of the most common data mining tasks. As the name implies, it is the process of taking historical data (the past), identifying patterns in the data that are seen though some methodology (the model), and then using the model to make predictions about what will happen in the future (scoring new data).

Data mining is a composite discipline that overlaps other branches of science. In Figure 4.1, we can see the contributions of many different fields in the development of the science of data mining. Because of the contributions of many disciplines, staying up to date on the progress being made in the field of data mining is a continuous educational challenge. In this section I discuss algorithms that come primarily from statistics and machine learning. These two groups largely live in different university departments (statistics and

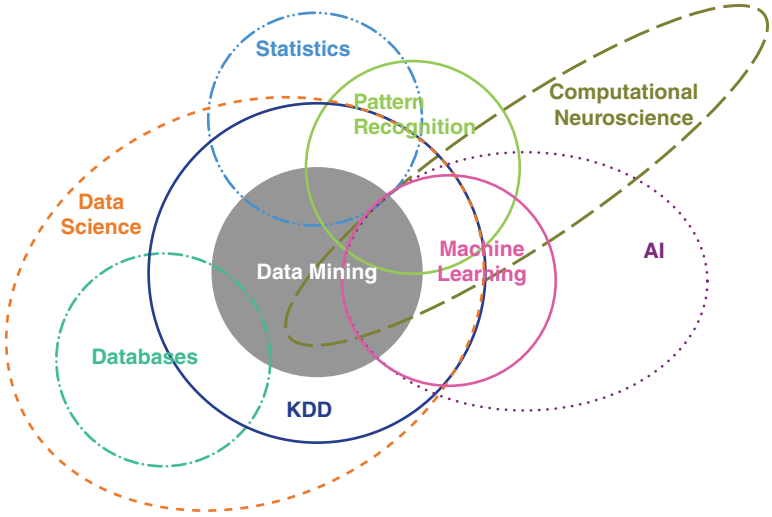

Figure 4.1 Multidisciplinary Nature of Data Mining
Source: SAS Enterprise Miner Training material from 1998.

computer science respectively) and in my opinion are feuding about the best way to prepare students for the field of data science. Statistics departments teach a great deal of theory but produce students with limited programming skills. Computer science departments produce great programmers with a solid understanding of how computer languages interact with computer hardware but have limited training on how to analyze data. This gap requires organizations to train new hires in these areas. There is a recent trend to give students a more complete education and better prepare them to contribute immediately in a data mining role. Until these programs build up their student bases, job applicants will likely know only half of the algorithms commonly used for modeling. For statisticians, that is regression, General Linear Models (GLMs), and decision trees. For the computer scientist, it is neural networks, support vector machines, and Bayesian methods.

Predictive modeling is not necessarily complicated or difficult; it is a fundamental task that we learn at an early age and hone as we grow and gain experience. In your daily life, you inherently do this many times a day. For instance, how long will your afternoon commute be?

My most variable commute has been living in the Washington, DC, area. The important factors in my commute time were these:

- The time I left the office
- The day of the week
- The season (was it summertime?)

Unless you work from home, your commute probably has similar key variables and perhaps a few more that were not important in my commute. Another prediction you make each day is what the weather will be. My children constantly want to wear shorts and T-shirts later into the fall season than social convention dictates. This year after the daily discussion about what the temperature might be during lunch, recess, PE, and the walk to and from school, my wife and I made a firm rule that after Veteran's Day (November 11), pants and long-sleeve shirts were required. Looking at average temperatures, it would have been reasonable to assume that the temperature would not rise to the point of "needing" to wear shorts, but looking at averages is sometimes not a good prediction technique. In the first week of December, while it was 30 degrees in Los Angeles, it was 75 degrees in Cary, North Carolina.

This short story illustrates a number of key points about predictive modeling:

- Sometimes models are wrong.
- The farther your time horizon, the more uncertainty there is.
- Averages (or averaging techniques) do not predict extreme values.

George Box was a pioneer and influential statistician (also the son-in-law of Sir R. A. Fischer) who taught at the University of Wisconsin. In 1987, he published a book titled *Empirical Model Building and Response Surfaces*. In it he makes two statements now famous in the statistical community related to the quality of predictive models. He says: "Remember that all models are wrong; the practical question is how wrong do they have to be to not be useful." Later in the same text he says: "Essentially, all models are wrong but some are useful." These points are important to keep in mind as you use increasing amounts of data to make more specific predictions. Logic and reason should not

be ignored because of a model result. Following a developed methodology and using clear processes will help reduce error and allow you to improve processes when things go wrong.

Building on the weather analogy, it is clear that for a stable process, predicting something immediate is more certain than farther out. The weather prediction for tomorrow is more likely to be accurate than the one for five or ten days into the future. The same is true for other stable processes like the stock market, real estate home prices, or gasoline. In statistical terms, this is referred to as a prediction interval, and it becomes larger the farther time horizon you predict for.

This chapter explains a methodology for building predictive models that has been used by thousands and thousands of data miners. After the methodology is developed, you are presented with information about the types of predictive models and their commonality and distinction.

A METHODOLOGY FOR BUILDING MODELS

The process of building models has been developed and refined by many practitioners over many years. Here is a simple, proven approach to building successful and profitable models.

1. **Prepare the data.** This step has to be completed before any meaningful exploration or analysis can take place.

   In larger organizations, this is done by a separate team and perhaps even a different business unit. Regardless of how your data is prepared, either through a formal request and accompanying specifications, as I did at the U.S. Census Bureau, or if you have permission to query directly from the enterprise data warehouse (EDW), this step is essential to a successful model.

   Investment in understanding the data preparation process within your organization often pays benefits in the long run. As you need access to increasingly larger and more granular data, knowledge of what data exists and how it can be combined to other data sources will provide insight that was

not possible just a few years ago. If your IT organization is not keeping more data for longer and at finer levels, then you are behind the trend and at risk for becoming irrelevant in your market.

2. **Perform exploratory data analysis.** This is the step where you understand your data and begin to gain intuition about relationships between variables. This exploration is best done with a domain expert, if you do not have that expertise. Fantastic data miners will discover relationships and trends through complicated exploration. They will, with great excitement, present these findings to a domain expert who will politely reply that those things have been known for some time. You cannot give a complete and thorough analysis and recommendation in today's business climate without domain expertise to complement analytical skill.

   The tools used for exploratory data analysis have changed in the past few years. In the late 1990s, this was done using mostly tabular output by means of SQL or a scripting language. There were correlation matrix and static graphical output to be pored over. The work was often slow, and programming skills were a requirement. In the last few years, the graphical tools have improved dramatically. There are now many commercial products from SAS, IBM, and SAP, as well as smaller vendors like QlikTech, and Tableau, to name a few, in the space of business visualization. These products are able to load data for visual exploration, generally through a browser-based interface, and provide a highly interactive experience in exploring data. This technology has been proven to work with billions of observations, assuming sufficient hardware resources are available.

   The exploration of data is never complete. There are always more ways to look at the data and interactions and relationships to consider, so the principle of sufficiency and the law of diminishing returns need to be observed. The law of diminishing returns comes from the field of economics and states that adding one more unit of effort (time in our case) will yield less increased value per unit for each successive

unit of effort you put into the task. In this case, the insight and knowledge you gain between hours 15 and 16 on data exploration is most likely less than the insight you gained between hours 2 and 3. The principle of sufficiency acknowledges the law of diminishing returns and sets a threshold on productivity loss. Stated in common language, this is: Know when to stop exploration. The software development methodology has moved to incorporate this idea through agile processes of learning a little, getting started, and continuous improvement.

3. **Build your first model.** The key for this step is to realize up front that the successful model-building process will involve many iterations. During some projects, the Thomas Edison quote that "I have not failed. I've just found 10,000 ways that won't work" will seem very apt. Until you build the first model, you are not able to accurately evaluate what the potential impact of the model will be. Build this model quickly with a method you are most comfortable with. (I frequently use a decision tree.) Building the first model helps to cement the criteria for success and set appropriate expectations for the people who will use the model predictions. Human nature is to be optimistic. We think products will sell better than they do, jobs will be completed without complications, and so on. Building the first model is a reality check for future performance and expectations. This first model is, by default, the champion model.
4. **Iteratively build models.** This phase is the where the majority of time should be spent. This step is a feedback loop where you will build a model (the challenger) and then compare it to the champion model using some objective criteria that defines the best model. If the challenger is better than the champion model, then evaluate if the challenger model satisfies the project objectives. If the project objectives are not met or the champion is not displaced then build another model. Often there is not a concrete model evaluation to determine when to stop but rather a time window that forces the project to end. Say you are contracted to provide a list of customers for a marketing

campaign. The campaign has a deadline for providing the customer list for next Tuesday, so model building will continue until that point in time.

sEMMA

sEMMA is a data mining methodology, created by SAS, that focuses on logical organization of the model development phase of data mining projects. It describes the process one must go through to capture insight and knowledge from their data. The acronym sEMMA—sample, explore, modify, model, assess—refers to the core process of conducting data mining. Beginning with a statistically representative sample of your data, sEMMA makes it easy to apply exploratory statistical and visualization techniques, select and transform the most significant predictive variables, model the variables to predict outcomes, and confirm a model's accuracy.

Before examining each stage of sEMMA, let us address a common misconception. A common misunderstanding is to refer to sEMMA as a data mining methodology. sEMMA is not a data mining methodology but rather a logical organization of the functional tool set of SAS Enterprise Miner for carrying out the core tasks of data mining. Enterprise Miner can be used as part of any iterative data mining methodology adopted by the client. Naturally, steps such as formulating a well-defined business or research problem and assembling quality representative data sources are critical to the overall success of any data mining project. sEMMA is focused on the model development aspects of data mining:

Sample (optional) your data by extracting a portion of a large data set. This must be big enough to contain the significant information, yet small enough to manipulate quickly. For optimal cost and performance, SAS Institute advocates a sampling strategy that applies a reliable, statistically representative sample of large full-detail data sources. Mining a representative sample instead of the whole volume reduces the processing time required to get crucial business information. If general patterns appear in the data as a whole, these will be traceable in a representative sample. If a niche is so tiny that it is not represented in a sample, and yet so

important that it influences the big picture, it can be discovered using summary methods. We also advocate creating partitioned data sets with the Data Partition node:

- Training—used for model fitting.
- Validation—used for assessment and to prevent overfitting.
- Test—used to obtain an honest assessment of how well a model generalizes.

Explore your data by searching for unanticipated trends and anomalies in order to gain understanding and ideas. Exploration helps refine the discovery process. If visual exploration does not reveal clear trends, you can explore the data through statistical techniques including factor analysis, correspondence analysis, and clustering. For example, in data mining for a direct mail campaign, clustering might reveal groups of customers with distinct ordering patterns. Knowing these patterns creates opportunities for personalized mailings or promotions.

Modify your data by creating, selecting, and transforming the variables to focus the model selection process. Based on your discoveries in the exploration phase, you may need to manipulate your data to include information, such as the grouping of customers and significant subgroups, or to introduce new variables. You may also need to look for outliers and reduce the number of variables to narrow them down to the most significant ones. You may also need to modify data when the "mined" data change either due to new data becoming available or newly discovered data errors. Because data mining is a dynamic, iterative process, you can update data mining methods or models when new information is available.

Model your data by allowing the software to search automatically for a combination of data that reliably predicts a desired outcome. Modeling techniques in data mining include neural networks, tree-based models, logistic models, and other statistical models—such as time series analysis, memory-based reasoning, and principal components. Each type of model has particular strengths and is appropriate within specific data mining situations depending on the data. For example, neural networks are very good at fitting highly complex nonlinear relationships.

Assess your data by evaluating the usefulness and reliability of the findings from the data mining process, and estimate how well it performs. A common means of assessing a model is to apply it to a portion of data set aside during the sampling stage. If the model is valid, it should work for this reserved sample as well as for the sample used to construct the model. Similarly, you can test the model against known data. For example, if you know which customers in a file had high retention rates and your model predicts retention, you can check to see whether the model selects these customers accurately. In addition, practical applications of the model, such as partial mailings in a direct mail campaign, help prove its validity.

By assessing the results gained from each stage of the sEMMA process, you can determine how to model new questions raised by the previous results and thus proceed back to the exploration phase for additional refinement of the data.

Once you have developed the champion model using the sEMMA-based mining approach, it then needs to be deployed to score new customer cases. Model deployment is the end result of data mining—the final phase in which the return on investment from the mining process is realized. Enterprise Miner automates the deployment phase by supplying scoring code in SAS, C, Java, and PMML. It not only captures the code for analytic models but also captures the code for preprocessing activities. You can seamlessly score your production data on a different machine and deploy the scoring code in batch or real time on the Web or directly in relational databases. This results in faster implementation and frees you to spend more time evaluating existing models and developing new ones.

sEMMA for the Big Data Era

How is sEMMA methodology impacted in the era of big data? The short answer is that largely it is not. sEMMA is a logical process that can be followed regardless of data size or complexity. However, the 's,' or sample, in sEMMA is less likely to be as critical with the more powerful systems available for mining the data. From my experience in working with big data, very large analytical databases can be addressed using sEMMA.

BINARY CLASSIFICATION

From my experience, binary classification is the most common type of predictive model. Key decision makers for corporations and other organizations must often make critical decisions quickly, requiring a system to arrive at a yes/no decision with confidence. These systems are not designed to do things part way or with some likelihood. They either do them or they do not. This is illustrated very well in the famous "go or no go for launch" checklist that each system's flight controller must answer. A launch can proceed only after the flight director has run through each system and received the "go for launch" answer. In the business world, many decisions are binary, such as should I extend you credit for this car purchase, or will you respond to this marketing campaign.

With predictive modeling of a binary target, the probability that an event will or will not occur is also very useful information. Let me provide an example to illustrate this point. With a binary target we are creating a black-and-white situation; will it rain tomorrow—yes or no? It will either rain or it will not, but you are very unlikely to see your local meteorologist (or even the National Weather Service) give a chance of rain at 0% or 100% because while it will either rain (100%) or not rain (0%), the estimate is implicitly the degree of confidence we have in our prediction. This confidence estimate is often much more useful than the binary prediction itself. Does your behavior differ if you see a 10% chance of rain versus a 40% chance of rain? Are you willing to leave your windows down or the top off of your convertible? Both of these predictions, because the percentage is less than 50, indicate that it is less likely to rain than not. However, from experience with having wet car seats for my commute home from a day when it was unlikely to rain (30%), I keep my windows up if there is *any* chance of rain because the downside risk is much greater than the potential upside.

An often-seen case in binary predictive modeling is when you are dealing with a "rare event." What qualifies as a rare event differs depending on the industry and domain, but it universally means that the event happens with extremely low probability and therefore might require some special treatment in the analysis.

Using an example from the sport of golf, let us examine the rare event more closely.[1] The most sought-after event in golf is the hole in one. This is where a golfer takes a tee shot during at least a 9-hole round and makes it in the hole in one shot. If you make a hole in one during some tournaments, you can often win a car, and custom dictates that regardless of when it happens, you buy a round of drinks for everyone in the clubhouse.[2] To be official, the shot must be witnessed by someone else. The United States Golf Association (USGA) keeps a registry of all holes in one and estimates the odds of hitting a hole in one to be 1 in 33,000. So while the odds of anyone hitting a hole in one is very small (.003%), some people, like Tiger Woods, not only have hit one but hit multiple holes in one. Tiger has 18 recorded holes in one but does not lead in this category. The most holes in one recorded goes to Norman Manley, with 59. To put quantitative terms around the odds of hitting a hole in one, we would say that no one is likely to hit a hole in one. Assuming 1 in 33,000 tee shots results in a hole in one and (after removing long holes) that there are 14 holes you could reasonably make a hole in one on, if you golfed a round every day of the year you would, on average, hit a hole in one every seven years.

Once we have determined the probability of the event occurring for each observation, they are then sorted in descending order from largest probability to smallest probability based on the likelihood the event will occur. Using the example above, if we were marketing commemorative hole-in-one plaques, we would apply the model to all golfers in our database sorting the list from highest probability to lowest probability of making a hole in one. We would then mail an offer letter to all the golfers from the top of the list down until our marketing campaign budget is exhausted or to those who exceeded a certain threshold. This will ensure that we send the offer for a commemorative plaque to the golfers, based on the model, most likely to hit a hole in one and who would be interested in our services.

[1] For full disclosure, I golfed a lot growing up but never hit a hole in one. I'm now in retirement since I haven't played a round of golf since my oldest child was born.

[2] Did you know that you can purchase hole-in-one insurance? Policies start at few hundred dollars depending on the prize offered during the tournament.

MULTILEVEL CLASSIFICATION

Multilevel or nominal classification is very similar to binary classification with the exception that there are now more than two levels. Nominal classification is an extension of binary classification. There are several examples where nominal classification is common, but for the most part this is the rarest of targets. An example of such a model can be seen in the cell phone industry when looking at customer churn. A company may not only be interested in the binary response of whether an account remains active or not. Instead, it may want to dive deeper and look at a nominal response of voluntary churn (customer chooses to cancel the contract), involuntary churn (e.g., contract was terminated due to delinquent payments), or an active customer.

In many cases, a nominal classification problem arises when an exception case is added to what could be a binary decision. This happens in the case of preventing credit card fraud, for example. When a credit card transaction is initiated, the card issuer has a very short window to accept or decline the transaction. While this could be thought of as a simple binary problem, accept or decline, there are some transactions in which the decision is not that straightforward and may fall into a gray area. A third level of response may be added to indicate that this transaction needs further review before a decision to accept or decline can be made.

The nominal classification problem poses some additional complications from a computational and also reporting perspective. Instead of just computing one probability of event (P) and then taking $1 - P$ to arrive at the probability of nonevent, you need to compute the probability of event #1 (P_1), the probability of event #2 (P_2), and so on until the last level which can be computed using $1-\sum_{i=1}^{n-1} P$. There is also a challenge in computing the misclassification rate. Since there are many choices, the report values must be calibrated to be easily interpreted by the report reader.

INTERVAL PREDICTION

The final type of prediction is interval prediction, which is used when the target level is continuous on the number line. Salary is an example of a prediction that employers and employees alike would like to make

accurately. The property and casualty insurance industry is an area with many interval prediction models. If you own a car in the United States, you are required to have insurance for it. To get that insurance, you likely requested a quote from several different insurance companies, and each one gave you a slightly different price. That price discrepancy is due to different predictive models and business factors that each insurance company uses. The business factors are the amount of exposure in certain geographic markets or an overall directive to try to gain market share in a certain market either geographic or economic strata.

Companies in the insurance industry generally utilize three different types of interval predictive models including claim frequency, severity, and pure premium. The insurance company will make predictions for each of these models based on its historical data and your specific information including the car make and model, your annual driving mileage, your past history with driving infractions, and so on.

ASSESSMENT OF PREDICTIVE MODELS

One consideration in the process of building predictive models will be to determine which model is best. A model is comprised of all the transformations, imputations, variable selection, variable binning, and so on manipulations that are applied to the data in addition to the chosen algorithm and its associated parameters. The large number of options and combinations makes a brute-force "try everything" method infeasible for any practical data mining problem. So the issue of model assessment plays an important role is selecting the best model. Model assessment, stated simply, is trying to find the best model for your application to the given data. The complexity comes from the term "best." Just like purchasing a washing machine, there are a number of aspects to consider and not everyone agrees on a single definition of "best." The common set of model assessment measures are listed and defined next. As an example, consider a local charity that is organizing a food drive. It has a donor mailing list of 125,000 people and data from past campaigns that it will use to train the model. We partition the data so that 100,000 people are in the training partition and 25,000 are in the validation partition. Both partitions have a response rate of 10%. The validation partition will be used for assessment, which is a best practice.

Table 4.1 Decision Matrix for Model Assessment

	Predicted Nonevent	Predicted Event
Nonevent	True negative	False positive
Event	False negative	True positive

Classification

There are a set of assessment measures that are based on the 2×2 decision matrix as shown in Table 4.1.

Classification is a popular method because it is easy to explain, it closely aligns with what most people associate at the "best" model, and it measures the model fit across all values. If the proportion of events and nonevents are not approximately equal, then the values need to be adjusted for making proper decisions. (See Table 4.2.)

Receiver Operating Characteristic

The receiver operating characteristics (ROC) are calculated for all points and displayed graphically for interpretation. The axis of the ROC plot are the Sensitivity and 1-Specificity, which were calculated from the classification rates.

Table 4.2 Formulas to Calculate Different Classification Measures

Measure	Formula
Classification rate (accuracy)	$\frac{\textit{true negative} + \textit{true positive}}{\textit{total observations}} \times 100$
Misclassification rate	$(1 - \frac{\textit{true negative} + \textit{true positive}}{\textit{total observations}}) \times 100$
Sensitivity (true positive rate)	$\frac{\textit{true positive}}{\textit{true positive} + \textit{false negative}} \times 100$
Specificity (true negative rate)	$\frac{\textit{true negative}}{\textit{false positive} + \textit{true negative}} \times 100$
1-Specificity (false positive rate)	$\frac{\textit{false positive}}{\textit{false positive} + \textit{true negative}} \times 100$

Lift

Lift is the ratio of percentage of correct responders to percentage of baseline response. To calculate lift, it must be accompanied by a percentile in the data. This is commonly referred to as a depth of file, and usually the first or second decile is chosen. For the food drive example, if we compute the lift at the first decile (10% of the data), the baseline (or random) model should have 2,500 responders to the campaign so that in the first decile there will be 250 responders (2,500×.1). Our model is good; it captures 300 responders in the first decile so the lift at the first decile is 1.2 (300/2500=12% captured response/10% baseline response). I prefer to use cumulative lift for my model assessment because it is monotonic and in practice campaigns will sort a list by the likelihood to respond and then market until a natural break is observed or the budget for the campaign is exhausted.

Gain

Gain is very similar to lift except 1 is subtracted from the value $\frac{\%\ of\ model\ events}{\%\ of\ baseline\ events} - 1$ for a given decile. For the food drive example, the gain would be 0.2 at the first decile.

Akaike's Information Criterion

Akaike's information criterion (AIC) is a statistical measure of the goodness of fit for a particular model. It maximizes the expression $-2(LL + k)$ where

k = number of estimated parameters (for linear regression the number of terms in the mode)

LL = maximized value of the log-likelihood function for the given model

The smaller the AIC, the better the model fits the data. Because of the k term, the smaller number of model parameters are favored. AIC values can be negative, but I do not remember ever encountering that in a real-world data mining problem.

Bayesian Information Criterion

The Bayesian information criterion (BIC) is a statistical measure similar to AIC but that maximizes the expression $-2LL + k \times \ln(n)$ where:

n = number of observations (or sample size)

k = number of estimated parameters (for linear regression the number of terms in the mode)

LL = maximized value of the log-likelihood function for the given model

For any two given models, the one with the lower BIC is better. This is because the number of terms in the model is smaller, the variables in the model better explain the variation of the target, or both.

Kolmogorov-Smirnov

The Kolmogorov-Smirnov (KS) statistic shows the point of maximum separation of the model sensitivity and the baseline on the ROC curve.

Model assessment is ideally done on a holdout partition that is representative of the data but was not used in the model-building phase. This holdout partition (often called a validation or test partition) is essential to measure how well your model will generalize to new incoming data.