



SECOND EDITION

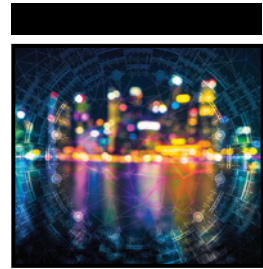
# Machine Learning

HANDS-ON FOR DEVELOPERS  
AND TECHNICAL PROFESSIONALS

JASON BELL

WILEY

# **Machine Learning**



---

# Machine Learning

Hands-On for Developers and Technical  
Professionals

Second Edition

Jason Bell

WILEY

Copyright © 2020 by John Wiley & Sons, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 978-1-119-64214-5

ISBN: 978-1-119-64225-1 (ebk)

ISBN: 978-1-119-64219-0 (ebk)

Manufactured in the United States of America

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

**Limit of Liability/Disclaimer of Warranty:** The publisher and the author make no representations or warranties with respect to the accuracy or completeness of the contents of this work and specifically disclaim all warranties, including without limitation warranties of fitness for a particular purpose. No warranty may be created or extended by sales or promotional materials. The advice and strategies contained herein may not be suitable for every situation. This work is sold with the understanding that the publisher is not engaged in rendering legal, accounting, or other professional services. If professional assistance is required, the services of a competent professional person should be sought. Neither the publisher nor the author shall be liable for damages arising herefrom. The fact that an organization or Web site is referred to in this work as a citation and/or a potential source of further information does not mean that the author or the publisher endorses the information the organization or website may provide or recommendations it may make. Further, readers should be aware that Internet websites listed in this work may have changed or disappeared between when this work was written and when it is read.

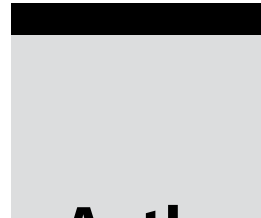
For general information on our other products and services please contact our Customer Care Department within the United States at (877) 762-2974, outside the United States at (317) 572-3993 or fax (317) 572-4002.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit [www.wiley.com](http://www.wiley.com).

**Library of Congress Control Number:** 2019956691

**Trademarks:** Wiley and the Wiley logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

*To all the developers who just wanted to get the code working without reading all the math stuff first.*



## About the Author

**Jason Bell** has worked in software development for more than 30 years. Currently he focuses on large-volume data solutions and helping retail and finance customers gain insight from data with machine learning. He is also an active committee member for several international technology conferences.



## About the Technical Editor

**Jacob Andresen** works as a senior software developer based in Copenhagen, Denmark. He has been working as a software developer and consultant in information retrieval systems and web applications since 2002.



# Acknowledgments

“Never again!” I think those were my final words after completing the first edition of this book. Five years later, and here we are again. When the call comes, you immediately think, “Well, it can’t be hard, can it?”

## To the Team

---

Jim Minatel, Devon Lewis, Janet Wehner, Pete Gaughan, and the rest of the team at Wiley, thank you for giving your blessing to this second edition and putting your faith in me to revise an awful lot of content. Apologies for the spelling mistakes and those *colour/color* occurrences. Many thanks to Jacob Andresen for giving a technical overview on the content of the book. His enthusiasm for the project was wonderful.

## Most Excellent Friends and Collaborators

---

Dearest friends and acquaintances, thank you: Jennifer Michael, Marie Bentall, Tim Brundle, Stephen Houston, Garrett Murphy, Clare Conway, Tom Spinks, Matt Johnston, Alan Edwards, Colin Mitchell, Simon Hewitt, Mary McKenna, Alan Thorburn, Colin McHale, Dan Lyons, Victoria McCallum, Andrew Bolster, Eoin McFadden, Catherine Muldoon, Amanda Paver, Ben Lorica, Alastair Croll, Mark Madsen, Ellen Friedman, Ted Dunning, Sophia DeMartini, Bruce Durling, Francine Bennett, Michelle Varron, Elise Huard, Antony Woods, John Stephenson, McCraigMcCraig of the Clan McCraig, everyone on the Clojurians Slack Channel, the Strata Data community, Carla Gaggini, Kiki Schirr, Wendy



Devolder, Brian O'Neill, Anthony O'Connor, Tom Gray, Deepa Mann-Kler, Alan Hook, Michelle Douglas, Pete Harwood, Jen Samuel, and Colin Masters. There are loads I've forgotten, I know. I'm sorry.

## And Finally

---

To my wife, Wendy, and my daughter, Clarissa, for absolutely everything and encouraging me to do these projects to the best of my nerdy ability. I couldn't have done it without you both.

To the rest of my family, Maggie, Fern, Andrew, Kerry, Ian and Margaret, William and Sylvia, thank you for all the support and kind words. William, if I need any more help, I'll call you.

## The Bios That Never Made It . . .

---

*"He has the boots and jacket that were the envy of many men."*

*"A dab hand at late-night YouTube videos of 80s pop stars."*

*"Jason Bell learned to play bass guitar on Saturday afternoons while pretending to work in a music shop."*

Thanks to everyone who reads this book. I hope it's helpful in your journey. It's an honor and privilege that you chose to read it. Now I believe it's time for a cup of tea.



# Contents

<b>Introduction</b>	<b>xxvii</b>
<b>Chapter 1 What Is Machine Learning?</b>	<b>1</b>
History of Machine Learning	1
Alan Turing	1
Arthur Samuel	2
Tom M. Mitchell	2
Summary Definition	3
Algorithm Types for Machine Learning	3
Supervised Learning	3
Unsupervised Learning	4
The Human Touch	4
Uses for Machine Learning	4
Software	4
Spam Detection	5
Voice Recognition	5
Stock Trading	5
Robotics	6
Medicine and Healthcare	6
Advertising	7
Retail and E-commerce	7
Gaming Analytics	9
The Internet of Things	10
Languages for Machine Learning	10
Python	10
R	11
Matlab	11
Scala	11
Ruby	11

Software Used in This Book	11
Checking the Java Version	12
Weka Toolkit	12
DeepLearning4j	13
Kafka	13
Spark and Hadoop	13
Text Editors and IDEs	13
Data Repositories	14
UC Irvine Machine Learning Repository	14
Kaggle	14
Summary	14
<b>Chapter 2 Planning for Machine Learning</b>	<b>15</b>
The Machine Learning Cycle	15
It All Starts with a Question	16
I Don't Have Data!	16
Starting Local	17
Transfer Learning	17
Competitions	17
One Solution Fits All?	18
Defining the Process	18
Planning	18
Developing	19
Testing	19
Reporting	19
Refining	19
Production	20
Avoiding Bias	20
Building a Data Team	20
Mathematics and Statistics	20
Programming	21
Graphic Design	21
Domain Knowledge	21
Data Processing	22
Using Your Computer	22
A Cluster of Machines	22
Cloud-Based Services	22
Data Storage	23
Physical Discs	23
Cloud-Based Storage	23
Data Privacy	23
Cultural Norms	24
Generational Expectations	24
The Anonymity of User Data	25
Don't Cross the "Creepy Line"	25
Data Quality and Cleaning	26
Presence Checks	26
Type Checks	27

	Length Checks	27
	Range Checks	28
	Format Checks	28
	The Britney Dilemma	28
	What's in a Country Name?	31
	Dates and Times	33
	Final Thoughts on Data Cleaning	33
	Thinking About Input Data	34
	Raw Text	34
	Comma-Separated Variables	34
	JSON	35
	YAML	37
	XML	37
	Spreadsheets	38
	Databases	39
	Images	39
	Thinking About Output Data	39
	Don't Be Afraid to Experiment	40
	Summary	40
<b>Chapter 3</b>	<b>Data Acquisition Techniques</b>	<b>43</b>
	Scraping Data	43
	Copy and Paste	44
	Google Sheets	46
	Using an API	47
	Acquiring Weather Data	48
	Using the Command Line	48
	Using Java	49
	Using Clojure	50
	Migrating Data	50
	Installing Embulk	51
	Using the Quick Run	51
	Installing Plugins	52
	Migrating Files to Database	53
	Bulk Converting CSV to JSON	55
	Summary	56
<b>Chapter 4</b>	<b>Statistics, Linear Regression, and Randomness</b>	<b>57</b>
	Working with a Basic Dataset	57
	Loading and Converting the Dataset	58
	Loading Data with Clojure	58
	Loading Data with Java	59
	Introducing Basic Statistics	59
	Minimum and Maximum Values	60
	Mathematical Notation	60
	Clojure	60
	Java	61
	Sum	61
	Mathematical Notation	61

Clojure	61
Java	61
Mean	62
Arithmetic Mean	62
Harmonic Mean	62
Geometric Mean	63
The Relationship Between the Three Averages	63
Clojure	63
Java	64
Mode	65
Clojure	65
Java	66
Median	66
Clojure	66
Java	66
Range	67
Clojure	67
Java	67
Interquartile Ranges	67
Clojure	68
Java	68
Variance	68
Clojure	68
Java	68
Standard Deviation	69
Clojure	69
Java	69
Using Simple Linear Regression	70
Using Your Spreadsheet	70
Using Excel	70
Loading the CSV Data	70
Creating a Scatter Plot	71
Showing the Trendline	72
Showing the Equation and R2 Value	72
Making a Prediction	73
Writing a Program	73
Embracing Randomness	75
Finding Pi with Random Numbers	76
Using Monte Carlo Pi in Clojure	77
Is the Dart Within the Circle?	77
Now Throw Lots of Darts!	78
Summary	80
<b>Chapter 5 Working with Decision Trees</b>	<b>81</b>
The Basics of Decision Trees	81
Uses for Decision Trees	81
Advantages of Decision Trees	82
Limitations of Decision Trees	82

Different Algorithm Types	82
ID3	83
C4.5	83
CHAID	83
MARS	84
How Decision Trees Work	84
Building a Decision Tree	85
Manually Walking Through an Example	85
Calculating Entropy	86
Information Gain	87
Rinse and Repeat	87
Decision Trees in Weka	88
The Requirement	88
Training Data	89
Relation	90
Attributes	90
Data	90
Using Weka to Create a Decision Tree	90
Creating Java Code from the Classification	94
Testing the Classifier Code	99
Thinking About Future Iterations	101
Summary	101
<b>Chapter 6 Clustering</b>	<b>103</b>
What Is Clustering?	103
Where Is Clustering Used?	104
The Internet	104
Business and Retail	104
Law Enforcement	105
Computing	105
Clustering Models	105
How the K-Means Works	106
Initialization	107
Assignments	107
Update	108
Calculating the Number of Clusters in a Dataset	108
The Rule of Thumb Method	108
The Elbow Method	109
The Cross-Validation Method	109
The Silhouette Method	109
K-Means Clustering with Weka	110
Preparing the Data	110
The Workbench Method	111
Loading Data	111
Clustering the Data	113
Visualizing the Data	115
The Command-Line Method	116
Converting CSV File to ARFF	116

	The First Run	117
	Refining the Optimum Clusters	118
	Name That Cluster	119
	The Coded Method	120
	Create the Project	120
	The Cluster Code	122
	Printing the Cluster Information	124
	Making Predictions	124
	The Final Code Listing	125
	Running the Program	127
	Further Development	128
	Summary	128
<b>Chapter 7</b>	<b>Association Rules Learning</b>	<b>129</b>
	Where Is Association Rules Learning Used?	129
	Web Usage Mining	130
	Beer and Diapers	130
	How Association Rules Learning Works	131
	Support	133
	Confidence	133
	Lift	134
	Conviction	134
	Defining the Process	134
	Algorithms	135
	Apriori	135
	FP-Growth	136
	Mining the Baskets—A Walk-Through	136
	The Raw Basket Data	136
	Using the Weka Application	137
	Inspecting the Results	141
	Summary	142
<b>Chapter 8</b>	<b>Support Vector Machines</b>	<b>143</b>
	What Is a Support Vector Machine?	143
	Where Are Support Vector Machines Used?	144
	The Basic Classification Principles	144
	Binary and Multiclass Classification	144
	Linear Classifiers	146
	Confidence	147
	Maximizing and Minimizing to Find the Line	147
	How Support Vector Machines Approach Classification	148
	Using Linear Classification	148
	Using Non-Linear Classification	150
	Using Support Vector Machines in Weka	151
	Installing LibSVM	151
	Weka LibSVM Installation	151
	A Classification Walk-Through	152
	Setting the Options	154
	Running the Classifier	156

	Dealing with Errors from LibSVM	158
	Saving the Model	158
	Implementing LibSVM with Java	158
	Converting .csv Data to .arff Format	158
	Setting Up the Project and Libraries	159
	Training and Predicting with the Existing Data	162
	Summary	164
<b>Chapter 9</b>	<b>Artificial Neural Networks</b>	<b>165</b>
	What Is a Neural Network?	165
	Artificial Neural Network Uses	166
	High-Frequency Trading	166
	Credit Applications	167
	Data Center Management	167
	Robotics	167
	Medical Monitoring	168
	Trusting the Black Box	168
	Breaking Down the Artificial Neural Network	169
	Perceptrons	169
	Activation Functions	170
	Multilayer Perceptrons	171
	Back Propagation	173
	Data Preparation for Artificial Neural Networks	174
	Artificial Neural Networks with Weka	175
	Generating a Dataset	175
	Loading the Data into Weka	177
	Configuring the Multilayer Perceptron	178
	Learning Rate	179
	Hidden Layers	179
	Training Time	179
	Training the Network	180
	Altering the Network	182
	Which Bit Is Which?	182
	Adding Nodes	182
	Connecting Nodes	182
	Removing Connections	182
	Removing Nodes	182
	Increasing the Test Data Size	183
	Implementing a Neural Network in Java	183
	Creating the Project	183
	Writing the Code	185
	Converting from CSV to Arff	188
	Running the Neural Network	188
	Developing Neural Networks with DeepLearning4J	189
	Modifying the Data	189
	Viewing Maven Dependencies	190
	Handling the Training Data	191
	Normalizing Data	191



	Building the Model	192
	Evaluating the Model	193
	Saving the Model	193
	Building and Executing the Program	194
	Summary	195
<b>Chapter 10</b>	<b>Machine Learning with Text Documents</b>	<b>197</b>
	Preparing Text for Analysis	198
	Apache Tika	198
	Downloading Tika	198
	Tika from the Command Line	199
	Tika Within an Application	202
	Cleaning the Text Data	203
	Convert Words to Lowercase	203
	Remove Punctuation	204
	Stopwords	205
	Stemming	206
	N-grams	206
	TF/IDF	207
	Loading the Documents	207
	Calculating the Term Frequency	208
	Calculating the Inverse Document Frequency	208
	Computing the TF/IDF Score	209
	Reviewing the Final Code Listing	209
	Word2Vec	211
	Loading the Raw Text Data	212
	Tokenizing the Strings	212
	Creating the Model	212
	Evaluating the Model	213
	Reviewing the Final Code	214
	Basic Sentiment Analysis	216
	Loading Positive and Negative Words	216
	Loading Sentences	217
	Calculating the Sentiment Score	217
	Reviewing the Final Code	218
	Performing a Test Run	220
	Further Development	220
	Summary	221
<b>Chapter 11</b>	<b>Machine Learning with Images</b>	<b>223</b>
	What Is an Image?	223
	Introducing Color Depth	224
	Images in Machine Learning	225
	Basic Classification with Neural Networks	226
	Basic Settings	226
	Loading the MNIST Images	226
	Model Configuration	227
	Model Training	228
	Model Evaluation	228

Convolutional Neural Networks	228
How CNNs Work	228
Feature Extraction	228
Activation Functions	230
Pooling	230
Classification	230
CNN Demonstration	231
Downloading the Image Data	231
Basic Setup	232
Handling the Training and Test Data	233
Image Preparation	233
CNN Model Configuration	234
Model Training	236
Model Evaluation	236
Saving the Model	237
Transfer Learning	237
Summary	238
<b>Chapter 12 Machine Learning Streaming with Kafka</b>	<b>239</b>
What You Will Learn in This Chapter	239
From Machine Learning to Machine Learning Engineer	240
From Batch Processing to Streaming Data Processing	241
What Is Kafka?	241
How Does It Work?	241
Fault Tolerance	243
Further Reading	243
Installing Kafka	243
Kafka as a Single-Node Cluster	244
Starting Zookeeper	244
Starting Kafka	245
Kafka as a Multinode Cluster	245
Starting the Multibroker Cluster	246
Topics Management	247
Creating Topics	248
Finding Out Information About Existing Topics	248
Deleting Topics	249
Sending Messages from the Command Line	249
Receiving Messages from the Command Line	250
Kafka Tool UI	250
Writing Your Own Producers and Consumers	251
Producers in Java	251
Properties	252
The Producer	253
Messages	253
The Final Code	253
Message Acknowledgments	254
Consumers in Java	255
Properties	255

Fetching Consumer Records	256
The Consumer Record	256
The Final Code	257
Building and Running the Applications	258
The Consumer Application	258
The Producer Application	259
The Streaming API	260
Streaming Word Counts	261
Building a Streaming Machine Learning System	262
Planning the System	263
What Topics Do We Require?	264
What Format Is the Data In?	264
Continuous Training	265
How to Install the Crontab Entries	265
Determining Which Models to Use for Predictions	266
Setting Up the Database	267
Determining Which Algorithms to Use	268
Decision Trees	268
Simple Linear Regression	271
Neural Network	274
Data Importing	275
Hidden Nodes	275
Model Configuration	276
Model Training	277
Evaluation	277
Saving the Model Results to the Database	277
Persisting the Model	277
The Final Code	278
Kafka Topics	281
Creating the Topics	281
Kafka Connect	283
Why Persist the Event Data?	283
Persisting Event Data	283
Persisting Training Data	284
Installing the Connector Configurations	284
The REST API Microservice	285
Processing Commands and Events	287
Finding Kafka Brokers	288
A Command or an Event?	289
Making Predictions	293
Prediction Streaming API	293
Prediction Functions	296
Predicting with Decision Tree Models	297
Predicting Linear Regression	298
Predicting the Neural Network Model	299
Running the Project	301
Run MySQL	301

	Run Zookeeper	301
	Run Kafka	301
	Create the Topics	301
	Run Kafka Connect	301
	Model Builds	302
	Run Events Streaming Application	302
	Run Prediction Streaming Application	302
	Start the API	302
	Send JSON Training Data	302
	Train a Model	302
	Make a Prediction	303
	Summary	303
<b>Chapter 13</b>	<b>Apache Spark</b>	<b>305</b>
	Spark: A Hadoop Replacement?	305
	Java, Scala, or Python?	306
	Downloading and Installing Spark	306
	A Quick Intro to Spark	306
	Starting the Shell	307
	Data Sources	307
	Testing Spark	308
	Load the Text File	308
	Make Some Quick Inspections	308
	Filter Text from the RDD	309
	Spark Monitor	309
	Comparing Hadoop MapReduce to Spark	310
	Writing Stand-Alone Programs with Spark	313
	Spark Programs in Java	313
	Using Maven to Build the Project	315
	Creating Packages in Maven	316
	Spark Program Summary	318
	Spark SQL	318
	Basic Concepts	318
	Wrapping Up SparkSQL	323
	Spark Streaming	323
	Basic Concepts	323
	Creating Your First Spark Stream	324
	Spark Streams from Kafka	326
	MLib: The Machine Learning Library	327
	Dependencies	328
	Decision Trees	328
	Clustering	330
	Association Rules with FP-Growth	332
	Summary	335
<b>Chapter 14</b>	<b>Machine Learning with R</b>	<b>337</b>
	Installing R	337
	macOS	337

Windows	338
Linux	338
Your First Run	338
Installing R-Studio	339
The R Basics	340
Variables and Vectors	340
Matrices	341
Lists	342
Data Frames	343
Installing Packages	344
Loading in Data	345
CSV Files	345
MySQL Queries	346
Creating Random Sample Data	346
Plotting Data	347
Bar Charts	347
Pie Charts	347
Dot Plots	348
Line Charts	349
Simple Statistics	350
Simple Linear Regression	350
Creating the Data	351
The Initial Graph	351
Regression with the Linear Model	351
Making a Prediction	352
Basic Sentiment Analysis	353
Using Functions to Load in Word Lists	353
Writing a Function to Score Sentiment	354
Testing the Function	354
Apriori Association Rules	355
Installing the arules Package	355
Gathering the Training Data	356
Importing the Transaction Data	356
Running the Apriori Algorithm	357
Inspecting the Results	358
Accessing R from Java	358
Installing the rJava Package	358
Creating Your First Java Code in R	359
Calling R from Java Programs	359
Setting Up an Eclipse Project	360
Creating the Java/R Class	361
Running the Example	361
Extending Your R Implementations	363
Connecting to Social Media with R	364
Summary	366
<b>Appendix A Kafka Quick Start</b>	<b>367</b>
Installing Kafka	367
Starting Zookeeper	367

---

Starting Kafka	368
Creating Topics	368
Listing Topics	369
Describing a Topic	369
Deleting Topics	369
Running a Console Producer	370
Running a Console Consumer	370
<b>Appendix B The Twitter API Developer Application Configuration</b>	<b>371</b>
<b>Appendix C Useful Unix Commands</b>	<b>375</b>
Using Sample Data	375
Showing the Contents: cat, more, and less	376
Example Command	376
Expected Output	376
Filtering Content: grep	377
Example Command for Finding Text	377
Example Output	377
Sorting Data: sort	378
Example Command for Basic Sorting	378
Example Output	378
Finding Unique Occurrences: uniq	380
Showing the Top of a File: head	381
Counting Words: wc	381
Locating Anything: find	382
Combining Commands and Redirecting Output	383
Picking a Text Editor	383
Colon Frenzy: Vi and Vim	383
Nano	384
Emacs	384
<b>Appendix D Further Reading</b>	<b>385</b>
Machine Learning	385
Statistics	386
Big Data and Data Science	386
Visualization	387
Making Decisions	387
Datasets	388
Blogs	388
Useful Websites	389
The Tools of the Trade	389
<b>Index</b>	<b>391</b>



# Introduction

Well, times have changed since writing the first edition of this book. Between 2014 and now there is more emphasis on data and what it can do for us but also how that power can be used against us. Hardware has gotten better, processing has gotten much faster, and the ability to classify, predict, and decide based on our data is extraordinary. At the same time, we've become much more aware of the risks of how data is used, the biases that can happen, and that a lot of black-box models don't always get things right.

Still, it's an exciting time to be involved. We still create more data than we can sensibly process. New ideas involving machine learning are being presented daily. The appetite for learning has grown rapidly, too.

Data mining and machine learning have been around a number of years already. When you look closely, the machine learning algorithms that are being applied aren't any different from what they were years ago; what is new is how they are applied at scale. When you look at the number of organizations that are creating the data, it's really, in my opinion, a minority. Google, Facebook, Twitter, Netflix, and a small handful of others are the ones getting the majority of mentions in the headlines with a mixture of algorithmic learning and tools that enable them to scale. So, the real question you should ask is, "How does all this apply to the rest of us?"

Data with large scale, near-instant processing, has come to the fore. The emphasis has moved from batch systems like Hadoop to more streaming-based systems like Kafka. I admit there will be times in this book when I look at the Big Data side of machine learning—it's a subject I can't ignore—but it's only a small factor in the overall picture of how to get insight from the available data. It is important to remember that I am talking about tools, and the key is figuring out which tools are right for the job you are trying to complete.

## Aims of This Book

---

This book is about machine learning and not about Big Data. It's about the various techniques used to gain insight from your data. By the end of the book, you will have seen how various methods of machine learning work, and you will also have had some practical explanations on how the code is put together, leaving you with a good idea of how you could apply the right machine learning techniques to your own problems.

There's no right or wrong way to use this book. You can start at the beginning and work your way through, or you can just dip in and out of the parts you need to know at the time you need to know them.

## "Hands-On" Means Hands-On

---

Many books on the subject of machine learning that I've read in the past have been very heavy on theory. That's not a bad thing. If you're looking for in-depth theory with really complex-looking equations, I applaud your rigor. Me? I'm more hands-on with my approach to learning and to projects. My philosophy is quite simple.

- Start with a question in mind.
- Find the theory I need to learn.
- Find lots of examples I can learn from.
- Put them to work in my own projects.

As a software developer, I like to see lots of examples. As a teacher, I like to get as much hands-on development time as possible but also get the message across to students as simply as possible. There's something about fingers on keys, coding away on your IDE, and getting things to work that's rather appealing, and it's something that I want to convey in the book.

Everyone has his or her own learning styles. I believe this book covers the most common methods, so everybody will benefit.

## "What About the Math?"

---

Like arguing that your favorite football team is better than another or trying to figure out whether Jimmy Page is a better guitarist than Jeff Beck (I prefer Beck), there are some things that will be debated forever and a day. One such debate is how much math you need to know before you can start doing machine learning.



Doing machine learning and learning the theory of machine learning are two very different subjects. To learn the theory, a good grounding in math is required. This book discusses a hands-on approach to machine learning. With the number of machine learning tools available for developers now, the emphasis is not so much on how these tools work but on how you can make these tools work for you. The hard work has been done, and those who did it deserve credit and applause.

## **“But You Need a PhD!”**

No, you don’t!

The long-running debate rages on about the level of knowledge you need before you can start doing analysis on data or claim that you are a data scientist. I believe that if you’d like to take a few years completing a degree and then pursuing the likes of a master’s degree and then a PhD, you should feel free to go that route. I’m a little more pragmatic about things and like to get reading and start doing.

Academia is great; and with the large number of online courses, papers, websites, and books on the subject of math, statistics, and data mining, there’s enough to keep the most eager of minds occupied. I dip in and out of these resources a lot, and it’s definitely a good way to keep up-to-date and investigate what’s emerging.

For me, though, there’s nothing like getting my hands dirty, grabbing some data, trying out some methods, and looking at the results. If you need to brush up on linear regression theory, then let me reassure you now, there’s plenty out there to read, and I’ll also cover that in this book.

Lastly, can one person ever be a data scientist? I think it’s more likely for a team of people to bring the various skills needed for machine learning into an organization. I talk about this more in Chapter 2.

So, while others in the office are arguing whether to bring some PhD brains in on a project, you can be coding up a decision tree to see whether it’s viable.

Over the last few years the job title data scientist has been joined by other titles like data engineer and machine learning engineer. All are valid and all focus on aspects of the data science pipeline. They all have their place.

## **What Will You Have Learned by the End?**

---

Assuming that you’re reading the book from start to finish, you’ll learn the common uses for machine learning, different methods of machine learning, and how to apply real-time and batch processing.

There's also nothing wrong with referencing a specific section that you want to learn. The chapters and examples were created in such a way that there's no dependency to learn one chapter over another.

The aim is to cover the common machine learning concepts in a practical manner. Using the existing free tools and libraries that are available to you, there's little stopping you from starting to gain insight from the existing data that you have.

## **Balancing Theory and Hands-on Learning**

---

There are many books on machine learning and data mining available, and finding the balance of theory and practical examples is hard. When planning this book, I stressed the importance of practical and easy-to-use examples, providing step-by-step instructions, so you can see how things are put together.

I'm not saying that the theory is light, because it's not. Understanding what you want to learn or, more importantly, how you want to learn will determine how you read this book.

You can think of the book split into three distinct sections. The first section covers the question, "What is machine learning?" and concentrates on planning for projects, data acquisition, and cleaning. For those wanting some refresher on the math and stats side of things, I've included a new chapter; it also covers linear regression and standard deviation.

The next section takes a closer look at some of the building-block algorithms used in machine learning projects. Clustering, decision trees, support vector machine, association rules learning, and neural networks provide both a background to how they work and code examples for you to work with. It's important to get the hands-on nature early on.

Lastly, I focus on the real-world tools used in enterprise; these are tools like Spark, Kafka, and R. Knowing how these frameworks and tools are put together will give you a grounding to know what to use when.

## **Source Code for This Book**

---

All the code that is explained in the chapters of the book has been saved on a GitHub repository for you to download and try. For this edition, I've also included the Maven dependency file so you can easily build the project you are working on.

The address for the repository is <https://github.com/jasebell/mlbook2nd-edition>. You can also find it on the Wiley website at [www.wiley.com/go/machinelearning2e](http://www.wiley.com/go/machinelearning2e).

The examples are in either Java, Clojure, or R. If you want to extend your knowledge into other languages, then a search around the GitHub site might lead you to some interesting examples.

Code has been separated by chapter; there's a folder in the repository for each of the chapters, and each has its own build file. The data is also within the repository in the data directory and has been split by each chapter.

## Using Git

---

Git is a version control system that is widely used in business and the open source software community. If you are working in teams, it becomes useful because you can create branches of the codebase to work on then merge the changes afterward.

The uses for Git in this book are limited, but you need it for “cloning” the repository of examples if you want to use them.

To clone the examples for this book, use the following commands:

```
$mkdir mlbookexamples
$cd mlbookexamples
$git clone https://github.com/jasebell/mlbook2ndedition.git
```

You see the progress of the cloning, and when it's finished, you'll be able to change directories to the newly downloaded folder and look at the code samples.

# What Is Machine Learning?

Let's start at the beginning, looking at what machine learning actually is, its history, and where it is used in industry. This chapter also describes some of the software used throughout the book so you can get everything installed and be ready to get working on the practical things.

## History of Machine Learning

---

So, what is the definition of machine learning? Over the last six decades, several pioneers of the industry have worked to steer us in the right direction.

### Alan Turing

In his 1950 paper, "Computing Machinery and Intelligence," Alan Turing asked, "Can machines think?" (For the full paper, see the link.)

[www.csee.umbc.edu/courses/471/papers/turing.pdf](http://www.csee.umbc.edu/courses/471/papers/turing.pdf)

The paper describes the "Imitation Game," which involves three participants—a human acting as a judge, another human, and a computer that is attempting to convince the judge that it is human. The judge would type into a terminal program to "talk" to the other two participants. Both the human and the

computer would respond, and the judge would decide which response came from the computer. If the judge couldn't consistently tell the difference between the human and computer responses, then the computer won the game.

The test continues today in the form of the Loebner Prize, an annual competition in artificial intelligence. The aim is simple enough: convince the judges that they are chatting to a human instead of a computer chat bot program.

## Arthur Samuel

In 1959, Arthur Samuel defined machine learning as a field of study that “gives computers the ability to learn without being explicitly programmed.” Samuel is credited with creating one of the first self-learning computer programs with his work at IBM. He focused on games as a way of getting the computer to learn things.

The game of choice for Samuel was checkers because it is a simple game but requires strategy from which the program could learn. With the use of alpha-beta evaluation pruning (eliminating nodes that do not need evaluating) and minimax strategies (minimizing the loss for the worst case), the program would discount moves and thus improve costly memory performance of the program.

Samuel is widely known for his work in artificial intelligence, but he was also noted for being one of the first programmers to use hash tables, and he certainly made a big impact at IBM.

## Tom M. Mitchell

Tom M. Mitchell is the chair of machine learning at Carnegie Mellon University. As author of the book *Machine Learning* (McGraw-Hill, 1997), his definition of machine learning is often quoted.

---

**A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with the experience E.**

---

The important thing here is that you now have a set of objects to define machine learning.

- Task (T), either one or more
- Experience (E)
- Performance (P)

So, with a computer running a set of tasks, the experience should be leading to performance increases.

## Summary Definition

*Machine learning* is a branch of artificial intelligence. Using computing, we design systems that can learn from data in a manner of being trained. The systems might learn and improve with experience and, with time, refine a model that can be used to predict outcomes of questions based on the previous learning.

## Algorithm Types for Machine Learning

---

There are a number of different algorithms that you can employ in machine learning. The required output is what decides which to use. As you work through the chapters, you'll see the different algorithm types being put to work. Machine learning algorithms characteristically fall into one of two learning types: supervised or unsupervised learning.

### Supervised Learning

*Supervised learning* refers to working with a set of labeled training data. For every example in the training data you have an input object and an output object. An example would be classifying Twitter data. (Twitter data is used a lot in the later chapters of the book.) Assume you have the following data from Twitter; these would be your input data objects:

```
Really loving the new St Vincent album!  
#fashion I'm selling my Louboutins! Who's interested? #louboutins  
I've got my Kafka cluster working on a load of data. #data
```

For your supervised learning classifier to know the outcome result of each tweet, you have to manually enter the answers; for clarity, I've added the resulting output object at the start of each line.

```
music    Really loving the new St Vincent album!  
clothing #fashion I'm selling my Louboutins! Who's interested?  
#louboutins  
bigdata  I've got my Kafka cluster working on a load of data. #data
```

Obviously, for the classifier to make any sense of the data when run properly, you have to work manually on a lot more input data. What you have, though, is a training set that can be used for the later classification of data.

There are issues with supervised learning that must be taken into account. The *bias-variance dilemma* is one of them: how the machine learning model performs accurately using different training sets. High-bias models contain restricted learning sets, whereas high-variance models learn with complexity