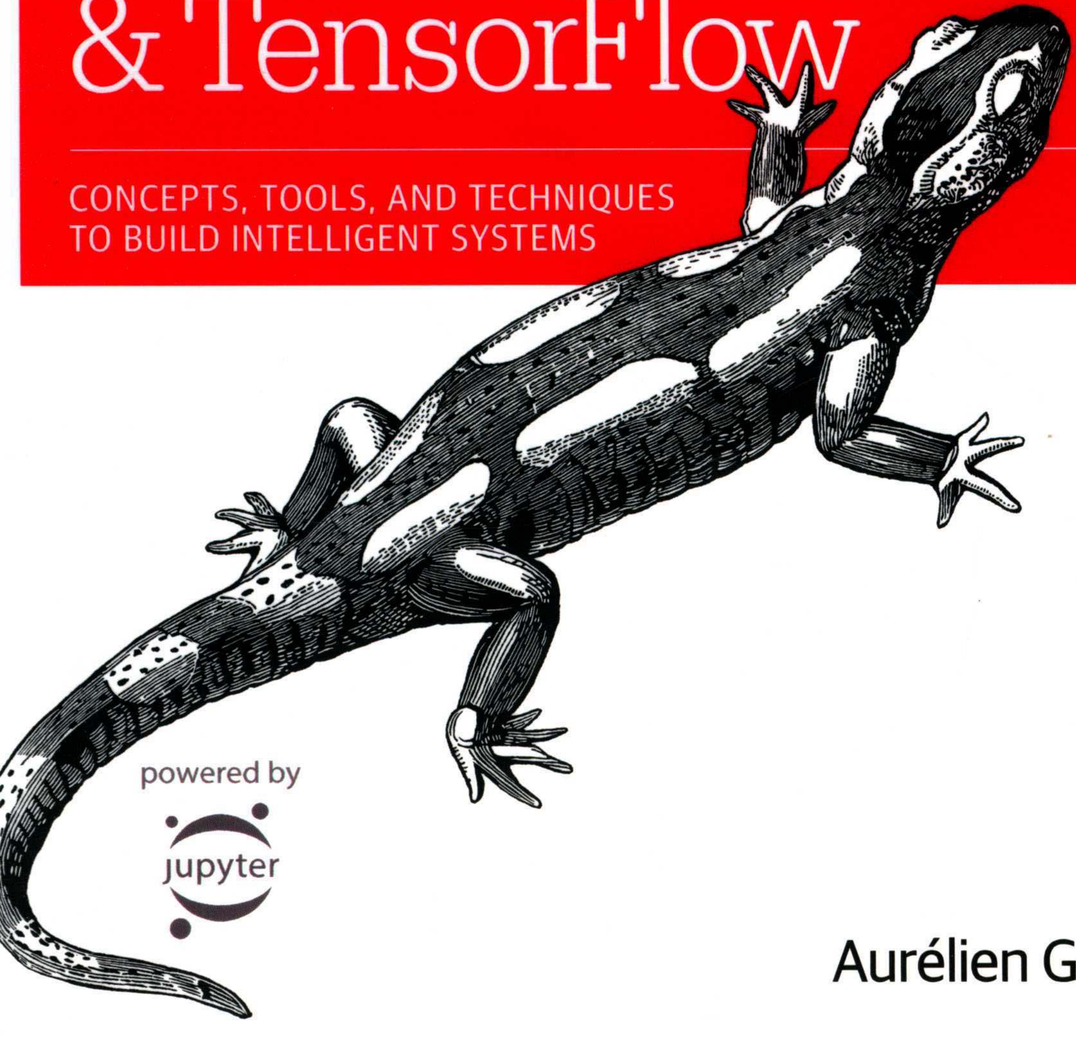
O'REILLY®

Hands-On Machine Learning with Scikit-Learn & TensorFlow

CONCEPTS, TOOLS, AND TECHNIQUES TO BUILD INTELLIGENT SYSTEMS



powered by jupyter

Aurélien Géron

Hands-On Machine Learning with Scikit-Learn and TensorFlow

Concepts, Tools, and Techniques to Build Intelligent Systems

Aurélien Géron

Beijing · Boston · Farnham · Sebastopol · Tokyo

O'REILLY®

Hands-On Machine Learning with Scikit-Learn and TensorFlow
by Aurélien Géron

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://oreilly.com/safari*). For more information, contact our corporate/institutional sales department: 800-998-9938 or *corporate@oreilly.com*.

Editor: Nicole Tache
Production Editor: Nicholas Adams
Copyeditor: Rachel Monaghan
Proofreader: Charles Roumeliotis

Indexer: Wendy Catalano
Interior Designer: David Futato
Cover Designer: Randy Comer
Illustrator: Rebecca Demarest

March 2017: First Edition

Revision History for the First Edition

2017-03-10: First Release
2017-06-09: Second Release

See *http://oreilly.com/catalog/errata.csp?isbn=9781491962299* for release details.

While the publisher and the author have used good faith efforts to ensure that the information and instructions contained in this work are accurate, the publisher and the author disclaim all responsibility for errors or omissions, including without limitation responsibility for damages resulting from the use of or reliance on this work. Use of the information and instructions contained in this work is at your own risk. If any code samples or other technology this work contains or describes is subject to open source licenses or the intellectual property rights of others, it is your responsibility to ensure that your use thereof complies with such licenses and/or rights

978-1-491-96229-9

[LSI]

Preface

The Machine Learning Tsunami

In 2006, Geoffrey Hinton et al. published a paper[1] showing how to train a deep neural network capable of recognizing handwritten digits with state-of-the-art precision (>98%). They branded this technique "Deep Learning." Training a deep neural net was widely considered impossible at the time,[2] and most researchers had abandoned the idea since the 1990s. This paper revived the interest of the scientific community and before long many new papers demonstrated that Deep Learning was not only possible, but capable of mind-blowing achievements that no other Machine Learning (ML) technique could hope to match (with the help of tremendous computing power and great amounts of data). This enthusiasm soon extended to many other areas of Machine Learning.

Fast-forward 10 years and Machine Learning has conquered the industry: it is now at the heart of much of the magic in today's high-tech products, ranking your web search results, powering your smartphone's speech recognition, and recommending videos, beating the world champion at the game of Go. Before you know it, it will be driving your car.

Machine Learning in Your Projects

So naturally you are excited about Machine Learning and you would love to join the party!

Perhaps you would like to give your homemade robot a brain of its own? Make it recognize faces? Or learn to walk around?

[1] Available on Hinton's home page at *http://www.cs.toronto.edu/~hinton/*.

[2] Despite the fact that Yann Lecun's deep convolutional neural networks had worked well for image recognition since the 1990s, although they were not as general purpose.

Or maybe your company has tons of data (user logs, financial data, production data, machine sensor data, hotline stats, HR reports, etc.), and more than likely you could unearth some hidden gems if you just knew where to look; for example:

- Segment customers and find the best marketing strategy for each group
- Recommend products for each client based on what similar clients bought
- Detect which transactions are likely to be fraudulent
- Predict next year's revenue
- And more (*https://www.kaggle.com/wiki/DataScienceUseCases*)

Whatever the reason, you have decided to learn Machine Learning and implement it in your projects. Great idea!

Objective and Approach

This book assumes that you know close to nothing about Machine Learning. Its goal is to give you the concepts, the intuitions, and the tools you need to actually implement programs capable of *learning from data*.

We will cover a large number of techniques, from the simplest and most commonly used (such as linear regression) to some of the Deep Learning techniques that regularly win competitions.

Rather than implementing our own toy versions of each algorithm, we will be using actual production-ready Python frameworks:

- Scikit-Learn (*http://scikit-learn.org/*) is very easy to use, yet it implements many Machine Learning algorithms efficiently, so it makes for a great entry point to learn Machine Learning.
- TensorFlow (*http://tensorflow.org/*) is a more complex library for distributed numerical computation using data flow graphs. It makes it possible to train and run very large neural networks efficiently by distributing the computations across potentially thousands of multi-GPU servers. TensorFlow was created at Google and supports many of their large-scale Machine Learning applications. It was open-sourced in November 2015.

The book favors a hands-on approach, growing an intuitive understanding of Machine Learning through concrete working examples and just a little bit of theory. While you can read this book without picking up your laptop, we highly recommend you experiment with the code examples available online as Jupyter notebooks at *https://github.com/ageron/handson-ml*.

Prerequisites

This book assumes that you have some Python programming experience and that you are familiar with Python's main scientific libraries, in particular NumPy (*http://numpy.org/*), Pandas (*http://pandas.pydata.org/*), and Matplotlib (*http://matplotlib.org/*).

Also, if you care about what's under the hood you should have a reasonable understanding of college-level math as well (calculus, linear algebra, probabilities, and statistics).

If you don't know Python yet, *http://learnpython.org/* is a great place to start. The official tutorial on python.org (*https://docs.python.org/3/tutorial/*) is also quite good.

If you have never used Jupyter, Chapter 2 will guide you through installation and the basics: it is a great tool to have in your toolbox.

If you are not familiar with Python's scientific libraries, the provided Jupyter notebooks include a few tutorials. There is also a quick math tutorial for linear algebra.

Roadmap

This book is organized in two parts. Part I, *The Fundamentals of Machine Learning*, covers the following topics:

- What is Machine Learning? What problems does it try to solve? What are the main categories and fundamental concepts of Machine Learning systems?
- The main steps in a typical Machine Learning project.
- Learning by fitting a model to data.
- Optimizing a cost function.
- Handling, cleaning, and preparing data.
- Selecting and engineering features.
- Selecting a model and tuning hyperparameters using cross-validation.
- The main challenges of Machine Learning, in particular underfitting and overfitting (the bias/variance tradeoff).
- Reducing the dimensionality of the training data to fight the curse of dimensionality.
- The most common learning algorithms: Linear and Polynomial Regression, Logistic Regression, k-Nearest Neighbors, Support Vector Machines, Decision Trees, Random Forests, and Ensemble methods.

Part II, *Neural Networks and Deep Learning*, covers the following topics:

- What are neural nets? What are they good for?
- Building and training neural nets using TensorFlow.
- The most important neural net architectures: feedforward neural nets, convolutional nets, recurrent nets, long short-term memory (LSTM) nets, and autoencoders.
- Techniques for training deep neural nets.
- Scaling neural networks for huge datasets.
- Reinforcement learning.

The first part is based mostly on Scikit-Learn while the second part uses TensorFlow.

> Don't jump into deep waters too hastily: while Deep Learning is no doubt one of the most exciting areas in Machine Learning, you should master the fundamentals first. Moreover, most problems can be solved quite well using simpler techniques such as Random Forests and Ensemble methods (discussed in Part I). Deep Learning is best suited for complex problems such as image recognition, speech recognition, or natural language processing, provided you have enough data, computing power, and patience.

Other Resources

Many resources are available to learn about Machine Learning. Andrew Ng's ML course on Coursera (*https://www.coursera.org/learn/machine-learning/*) and Geoffrey Hinton's course on neural networks and Deep Learning (*https://www.coursera.org/course/neuralnets*) are amazing, although they both require a significant time investment (think months).

There are also many interesting websites about Machine Learning, including of course Scikit-Learn's exceptional User Guide (*http://scikit-learn.org/stable/user_guide.html*). You may also enjoy Dataquest (*https://www.dataquest.io/*), which provides very nice interactive tutorials, and ML blogs such as those listed on Quora (*http://goo.gl/GwtU3A*). Finally, the Deep Learning website (*http://deeplearning.net/*) has a good list of resources to learn more.

Of course there are also many other introductory books about Machine Learning, in particular:

- Joel Grus, *Data Science from Scratch* (O'Reilly). This book presents the fundamentals of Machine Learning, and implements some of the main algorithms in pure Python (from scratch, as the name suggests).

- Stephen Marsland, *Machine Learning: An Algorithmic Perspective* (Chapman and Hall). This book is a great introduction to Machine Learning, covering a wide range of topics in depth, with code examples in Python (also from scratch, but using NumPy).
- Sebastian Raschka, *Python Machine Learning* (Packt Publishing). Also a great introduction to Machine Learning, this book leverages Python open source libraries (Pylearn 2 and Theano).
- Yaser S. Abu-Mostafa, Malik Magdon-Ismail, and Hsuan-Tien Lin, *Learning from Data* (AMLBook). A rather theoretical approach to ML, this book provides deep insights, in particular on the bias/variance tradeoff (see Chapter 4).
- Stuart Russell and Peter Norvig, *Artificial Intelligence: A Modern Approach, 3rd Edition* (Pearson). This is a great (and huge) book covering an incredible amount of topics, including Machine Learning. It helps put ML into perspective.

Finally, a great way to learn is to join ML competition websites such as Kaggle.com this will allow you to practice your skills on real-world problems, with help and insights from some of the best ML professionals out there.

Conventions Used in This Book

The following typographical conventions are used in this book:

Italic

Indicates new terms, URLs, email addresses, filenames, and file extensions.

`Constant width`

Used for program listings, as well as within paragraphs to refer to program elements such as variable or function names, databases, data types, environment variables, statements and keywords.

`Constant width bold`

Shows commands or other text that should be typed literally by the user.

`Constant width italic`

Shows text that should be replaced with user-supplied values or by values determined by context.

This element signifies a tip or suggestion.

This element signifies a general note.

This element indicates a warning or caution.

Using Code Examples

Supplemental material (code examples, exercises, etc.) is available for download at *https://github.com/ageron/handson-ml*.

This book is here to help you get your job done. In general, if example code is offered with this book, you may use it in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books does require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation does require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*Hands-On Machine Learning with Scikit-Learn and TensorFlow* by Aurélien Géron (O'Reilly). Copyright 2017 Aurélien Géron, 978-1-491-96229-9."

If you feel your use of code examples falls outside fair use or the permission given above, feel free to contact us at *permissions@oreilly.com*.

O'Reilly Safari

Safari (formerly Safari Books Online) is a membership-based training and reference platform for enterprise, government, educators, and individuals.

Members have access to thousands of books, training videos, Learning Paths, interactive tutorials, and curated playlists from over 250 publishers, including O'Reilly Media, Harvard Business Review, Prentice Hall Professional, Addison-Wesley Professional, Microsoft Press, Sams, Que, Peachpit Press, Adobe, Focal Press, Cisco Press, John Wiley & Sons, Syngress, Morgan Kaufmann, IBM Redbooks, Packt, Adobe Press, FT Press, Apress, Manning, New Riders, McGraw-Hill, Jones & Bartlett, and Course Technology, among others.

For more information, please visit *http://oreilly.com/safari*.

How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
800-998-9938 (in the United States or Canada)
707-829-0515 (international or local)
707-829-0104 (fax)

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at *http://bit.ly/hands-on-machine-learning-with-scikit-learn-and-tensorflow*.

To comment or ask technical questions about this book, send email to *bookquestions@oreilly.com*.

For more information about our books, courses, conferences, and news, see our website at *http://www.oreilly.com*.

Find us on Facebook: *http://facebook.com/oreilly*

Follow us on Twitter: *http://twitter.com/oreillymedia*

Watch us on YouTube: *http://www.youtube.com/oreillymedia*

Acknowledgments

I would like to thank my Google colleagues, in particular the YouTube video classification team, for teaching me so much about Machine Learning. I could never have started this project without them. Special thanks to my personal ML gurus: Clément Courbet, Julien Dubois, Mathias Kende, Daniel Kitachewsky, James Pack, Alexander Pak, Anosh Raj, Vitor Sessak, Wiktor Tomczak, Ingrid von Glehn, Rich Washington, and everyone at YouTube Paris.

I am incredibly grateful to all the amazing people who took time out of their busy lives to review my book in so much detail. Thanks to Pete Warden for answering all my TensorFlow questions, reviewing Part II, providing many interesting insights, and of course for being part of the core TensorFlow team. You should definitely check out his blog (*https://petewarden.com/*)! Many thanks to Lukas Biewald for his very thorough review of Part II: he left no stone unturned, tested all the code (and caught a few errors), made many great suggestions, and his enthusiasm was contagious. You should check out his blog (*https://lukasbiewald.com/*) and his cool robots (*https://goo.gl/Eu5u28*)! Thanks to Justin Francis, who also reviewed Part II very thoroughly, catching errors and providing great insights, in particular in Chapter 16. Check out his posts (*https://goo.gl/28ve8z*) on TensorFlow!

Huge thanks as well to David Andrzejewski, who reviewed Part I and provided incredibly useful feedback, identifying unclear sections and suggesting how to improve them. Check out his website (*http://www.david-andrzejewski.com/*)! Thanks to Grégoire Mesnil, who reviewed Part II and contributed very interesting practical advice on training neural networks. Thanks as well to Eddy Hung, Salim Sémaoune, Karim Matrah, Ingrid von Glehn, Iain Smears, and Vincent Guilbeau for reviewing Part I and making many useful suggestions. And I also wish to thank my father-in-law, Michel Tessier, former mathematics teacher and now a great translator of Anton Chekhov, for helping me iron out some of the mathematics and notations in this book and reviewing the linear algebra Jupyter notebook.

And of course, a gigantic "thank you" to my dear brother Sylvain, who reviewed every single chapter, tested every line of code, provided feedback on virtually every section, and encouraged me from the first line to the last. Love you, bro!

Many thanks as well to O'Reilly's fantastic staff, in particular Nicole Tache, who gave me insightful feedback, always cheerful, encouraging, and helpful. Thanks as well to Marie Beaugureau, Ben Lorica, Mike Loukides, and Laurel Ruma for believing in this project and helping me define its scope. Thanks to Matt Hacker and all of the Atlas team for answering all my technical questions regarding formatting, asciidoc, and LaTeX, and thanks to Rachel Monaghan, Nick Adams, and all of the production team for their final review and their hundreds of corrections.

Last but not least, I am infinitely grateful to my beloved wife, Emmanuelle, and to our three wonderful kids, Alexandre, Rémi, and Gabrielle, for encouraging me to work hard on this book, asking many questions (who said you can't teach neural networks to a seven-year-old?), and even bringing me cookies and coffee. What more can one dream of?

Table of Contents

Part II. Neural Networks and Deep Learning