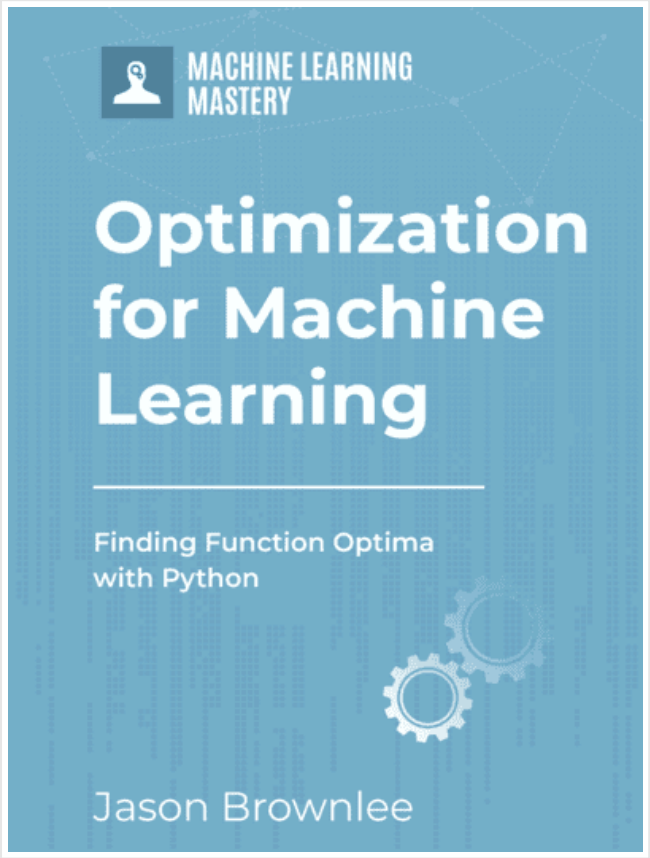
Navigation

Optimization for Machine Learning

Finding Function Optima with Python



$37 USD

Optimization happens everywhere. Machine learning is one example of such and gradient descent is probably the most famous algorithm for performing optimization. Optimization means to find the best value of some function or model. That can be the maximum or the minimum according to some metric.

Using clear explanations, standard Python libraries, and step-by-step tutorial lessons, you will learn how to find the optimum point to numerical functions confidently using modern optimization algorithms.

About this Ebook:

Read on all devices: English PDF format EBook, no DRM.
Tons of tutorials: 30 step-by-step lessons, 412 pages.
Foundations: functional optimization, how to choose an algorithm, and more.
Many algorithms: Nelder-Mead, BFGS, simulated annealing, RMSProp, and more.
Working code: 104 Python (.py) code files included.

Clear, Complete End-to-End Examples.

Convinced?

Jump Straight to the Packages

…so What is Function Optimization?

Function optimization is to find the maximum or minimum value of a function. The function may have any structure as long as it produces numerical values.

If we got a function as a blackbox how can we find its maximum or minimum? We can check out every single possible input to see which one will give the best output, or we can assume the function might behave in certain manner, such as output is continuous to its input, and exploit this by, for example, hill-climbing. There are vast amount of optimization algorithms, each was proposed together with certain assumptions or heuristics. The most suitable one for a particular function may not fit another.

…so Why do we care about Function Optimization?

In applied machine learning, we can construct a function that is a blackbox model with a predefined set of training and test data. The input to this function are the model's hyperparameters and the output is the evaluation score. So we are looking for what hyperparameters can produce the best score. Or, the function can be the model itself, and we are looking for what weight parameters to produce the lowest error rate for that given dataset.

These are just a few examples of how function optimization is related to machine learning. In fact, when the computer busy working on training the machine learning model, it is the optimization algorithm in action. When human is involved, to decide on what kind of model to use and how to configure or set up the model, we are also doing an optimization at the higher level without noticing it.

Nevertheless, there are 2 main reasons to we need to learn function optimization, they are:

To Use

The ability to apply function optimization freely allows us to go to a new level in various stages of machine learning. We may find better models by hyperparameter tuning. We may also produce better input data by feature selection in preprocessing stage.

Optimization algorithms **allow us to use machine learning to its potential**.

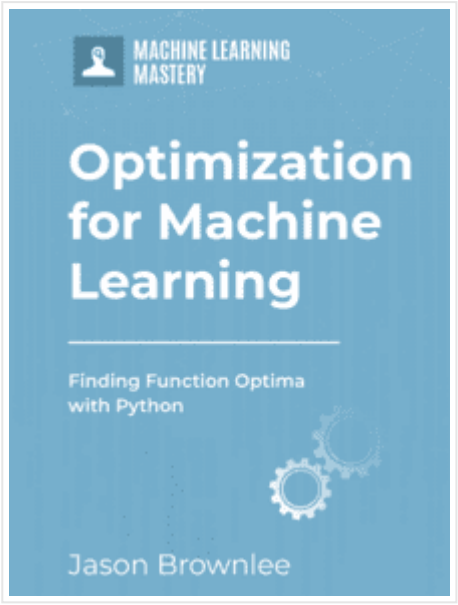
To Understand

Machine learning is doing a lot of optimization behind the scenes. When new algorithm invented or new technique proposed, it is inevitable to explain them in terms of optimization. Hence we also need to understand them from optimization perspectives.

Knowledgable in optimization algorithms **allow us to communicate the action of machine learning better**.

The best illustration of these is from the history of development in neural network models. When we started with gradient descent and later we have Adam algorithm to use, it is only possible to understand the reason for this progression if you understand function optimization.

Introducing My New EBook:
"*Optimization for Machine Learning*"



Welcome to the EBook: **Optimization for Machine Learning**.

I designed this book to teach machine learning practitioners, like you, step-by-step how to use the most common function optimization algorithms with examples in Python.

This book was carefully designed to help you bring a wide variety of the **proven and powerful optimization algorithms** to your next project.

The tutorials were designed to teach you these techniques the fastest and most effective way that I know how: to learn by doing. With executable code that you can run to develop the intuitions required, and that you can copy-and-paste into your project and **immediately get a result**.

Function optimization is important to machine learning, and I believe that if it is taught at the right level for practitioners, it can be a **fascinating, fun, directly applicable**, and immeasurably useful toolbox of techniques.

I hope that you agree.

Convinced?
Click to jump straight to the packages.

…so, is this book right for YOU?
Who Is This Book For?

Let's make sure you are in the right place.

This book is for developers that may know some applied machine learning. Maybe you know how to work through a predictive modeling problem end-to-end, or at least most of the main steps, with popular tools.

This guide was written in the top-down and results-first machine learning style that you're used to from Machine Learning Mastery.

The lessons in this book do assume a few things about you, such as:

You need to know:

You know your way around basic Python for programming.

You do NOT need to be:

You do not need to be a math wiz! You do not need to be a master programmer!

You may know some basic NumPy for array manipulation.
You may know some basic scikit-learn for modeling.

You do not need to be a machine learning expert!

About Your Outcomes
…so what will YOU know after reading this book?

After reading and working through this book, you will know:

The intuition behind optimizing a function without knowing the details of how the function computed its value.
How the nature of an objective function affects the applicability of an optimization algorithm.
The trade-offs made in applying different optimization algorithm.
The difference between local and global optimization, and between deterministic and stochastic optimization.
Different techniques to monitor the progress of optimization algorithm in action.

How use SciPy to optimize your own objective function using various optimization algorithms.
How to develop and apply heuristic optimization algorithms such as hill climbing and simulated annealing.
How to develop and apply evolutionary optimization such as genetic algorithms.
How to implement gradient descent and its many variations and apply it on your own objective function.
How the optimization algorithms can help developing a machine learning solution in various aspects.

But, *what if…?*

Do you have some doubts? Let me see if I can help.

What if I Am New to Machine Learning?

This book does not assume you have a background in machine learning.

That being said, I do recommend that you learn how to work through a predictive modeling problem first. It will give you the context for the challenge of optimization.

What if I Am Just a Developer?

Perfect. I wrote this book for you.

What if My Math is Really Poor?

What if I Am Not a Python Programmer?

You can handle this book if you are a programmer in another language, even if you are not experienced in Python.

Everything is demonstrated with a small code example that you can run directly.

All code is provided for you to play with, modify, and learn from.

The book even has an appendix to show you how to set up Python on your workstation.

Perfect. This book is for you.

No complicated math used, only the concept of functions and some basic arithmetic.

What if I Am Working Through another Machine Learning Course?

Excellent!

This book is not a substitute for an undergraduate course in machine learning or a textbook for such a course, although it is a great complement to such materials.

What Exactly Is In This Book?
…see the table of contents

This book was designed around major function optimization techniques that are directly relevant to real-world problems.

There are a lot of things you could learn about function optimization, from theory to abstract concepts to APIs. My goal is to take you straight to developing an intuition for the elements you must understand with laser-focused tutorials.

The tutorials were designed to teach you how to get results with function optimization algorithms. As such, the tutorials give you the tools to both rapidly understand and apply each technique or operation. There is a mixture of both tutorial lessons and projects to both introduce the methods and give plenty of examples and opportunities to practice using them.

Each of the tutorials is designed to take you about one hour to read through and complete, excluding the extensions and further reading.

You can choose to work through the lessons one per day, one per week, or at your own pace. I think momentum is critically important, and this book is intended to be read and used, not to sit idle.

I would recommend picking a schedule and sticking to it.

The tutorials are divided into six parts; they are:

Part 1: Foundation: Discover the nature of function optimization, why they are important to machine learning and how to develop an intuition for what is being optimized.
Part 2: Background: Discover the background required for understanding the process the outcome of function optimization, including the broad categories of optimization algorithms, and how to visualize the progress.
Part 3: Local Optimization: Discover various local optimization techniques, and the difference in the requirements of several algorithms.
Part 4: Global Optimization: Discover several global optimization algorithms that would not be trapped by “local optima” but rather to have potential to look for the optimal solution in the entire function domain.

Part 5: Gradient Descent: Discover gradient descent, the most famous optimization algorithm in machine learning, with its weakness and the many variations that aimed to alleviate it.

Part 6: Projects: Discover the way function optimization can be used in practice through examples.

Lessons Overview

Below is an overview of the 30 step-by-step tutorial lessons you will work through:

Each lesson was designed to be completed in about 30-to-60 minutes by an average developer.

Part 1: Foundation

- **Lesson 01**: What is Function Optimization
- **Lesson 02**: Optimization and Machine Learning
- **Lesson 03**: How to Choose an Optimization Algorithm

Part 2: Background

- **Lesson 04**: No Free Lunch Theorem for Machine Learning
- **Lesson 05**: Local Optimization vs. Global Optimization
- **Lesson 06**: Premature Convergence
- **Lesson 07**: Creating Visualization for Function Optimization
- **Lesson 08**: Stochastic Optimization Algorithms
- **Lesson 09**: Random Search and Grid Search

Part 3: Local Optimization

- **Lesson 10**: What is a Gradient in Machine Learning?
- **Lesson 11**: Univariate Function Optimization
- **Lesson 12**: Pattern Search: Nelder-Mead Optimization Algorithm
- **Lesson 13**: Second Order: The BFGS and L-BFGS-B Optimization Algorithms
- **Lesson 14**: Least Square: Curve Fitting with SciPy
- **Lesson 15**: Stochastic Hill Climbing

- **Lesson 16**: Iterated Local Search

Part 4: Global Optimization

- **Lesson 17**: Simple Genetic Algorithm from Scratch
- **Lesson 18**: Evolution Strategies
- **Lesson 19**: Differential Evolution
- **Lesson 20**: Simulated Annealing from Scratch

Part 5: Gradient Descent

- **Lesson 21**: Gradient Descent Optimization from Scratch
- **Lesson 22**: Gradient Descent with Momentum
- **Lesson 23**: Gradient Descent with AdaGrad
- **Lesson 24**: Gradient Descent with RMSProp
- **Lesson 25**: Gradient Descent with Adadelta
- **Lesson 26**: Adam Optimization Algorithm

Part 6: Projects

- **Lesson 27**: Use Optimization Algorithms to Manually Fit Regression Models
- **Lesson 28**: Optimize Neural Network Models
- **Lesson 29**: Feature Selection using Stochastic Optimization
- **Lesson 30**: Manually Optimize Machine Learning Model Hyperparameters

Appendix

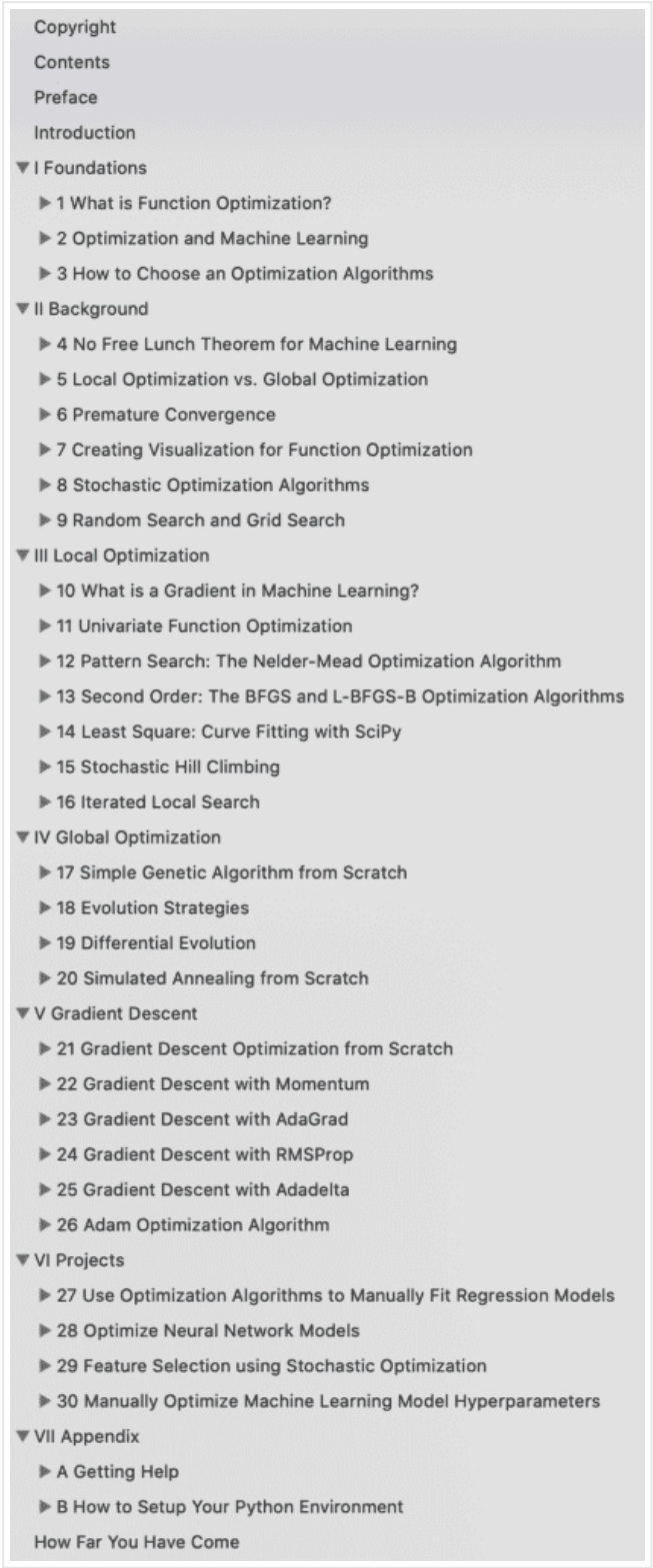
- **Appendix A**: Getting help
- **Appendix B**: How to Setup Your Python Environment

You can see that each part targets a specific learning outcome, and so does each tutorial within each part. This acts as a filter to ensure you are only focused on the things you need to know to get to a specific result and do not get bogged down in the math or near-infinite number of digressions.

The tutorials were not designed to teach you everything there is to know about each of the

EBook Table of Contents

The screenshot below was taken from the PDF Ebook. It provides you a full overview of the table of contents from the book.



theories or techniques. They were designed to give you an understanding of how they work, how to use them, and how to interpret the results the fastest way I know how: to **learn by doing**.

Here's Everything You'll Get… in "*Optimization for Machine Learning*"

A digital download that contains everything you need, including:

- Clear descriptions to help you understand optimization algorithms for applied machine learning.
- Step-by-step Python tutorials to show you exactly how to apply each technique and algorithm.
- End-to-end self-contained examples that give you everything you need in each tutorial without assuming prior knowledge.
- Python source code recipes for every example in the book so that you can run the tutorial code in seconds.
- Digital Ebook in PDF format so that you can have the book open side-by-side with the code and see exactly how each example works.

Resources you need to go deeper, when you need to, including:

- The best sources of information on the Python ecosystem including the SciPy, NumPy, and scikit-learn libraries.
- The best places online where you can ask your challenging questions and actually get a response.
- The best books, and articles to learn more about each technique covered.

Foundations required for developing and using optimization algorithms, including:

Specific optimization algorithms that we can use from SciPy, including:

- Brent's method.
- Nelder-Mead algorithm.
- BGFS and L-BGFS-B algorithms.
- Curve fitting.

Techniques in implementing optimization algorithms, including:

- Hill climbing algorithm and its extension to use iterated local search.
- Genetic algorithm and its application to discrete and continuous functions.
- Evolution strategies and differential evolution.
- Simulated annealing.
- Gradient descent, and its variations including momentum, AdaGrad, RMSProp, Adadelta, and Adam.

Details on the technical aspects of gradient descent, including:

- Gradient descent as a first-order method that applicable to differentiable functions.
- What gradient tells about the optimal value of a function.
- How gradient descent depends on the initial point, step size, and stopping criteria.
- The problem of gradient descent applied to functions with noisy gradient.

The scope of optimization and its limitation in applied machine learning.
The choice between local optimization and global optimization.
The problem of premature convergence and how to address it.
Use exhaustive search as baseline for performance evaluation of other optimization algorithms.
Techniques of stochastic optimization.
How visualization can be used as a proof of optimization is being done.

The use of momentum to get out of local optimal in gradient descent.
How to make the step size in gradient descent adaptive to the curvature.
The use of movement estimation to further improve the adaptive step size in gradient descent.

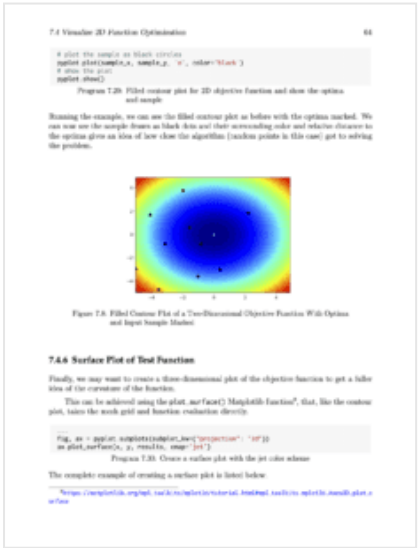
Experiences in applying optimization algorithms, including:

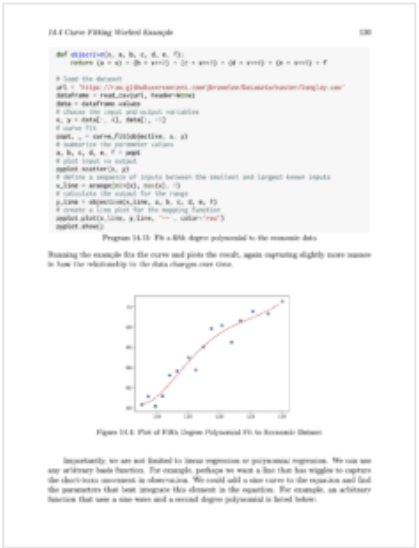
Implementing and fitting a regression model.
Implementing and fitting a multilayer perceptron model with various transfer functions.
Performing feature selection for a machine learning model.
Performing hyperparameter optimization with bounds constraints for a machine learning model.

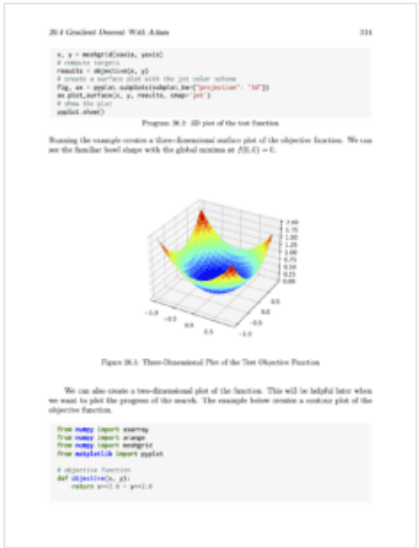
What More Do You Need?

Take a Sneak Peek Inside The EBook

Click an image to Enlarge.







BONUS: Optimization Algorithms With Python Code Recipes

….you also get 104 fully working Python scripts

Sample Code Recipes

Each recipe presented in the book is standalone, meaning that you can copy and paste it into your project and use it immediately.

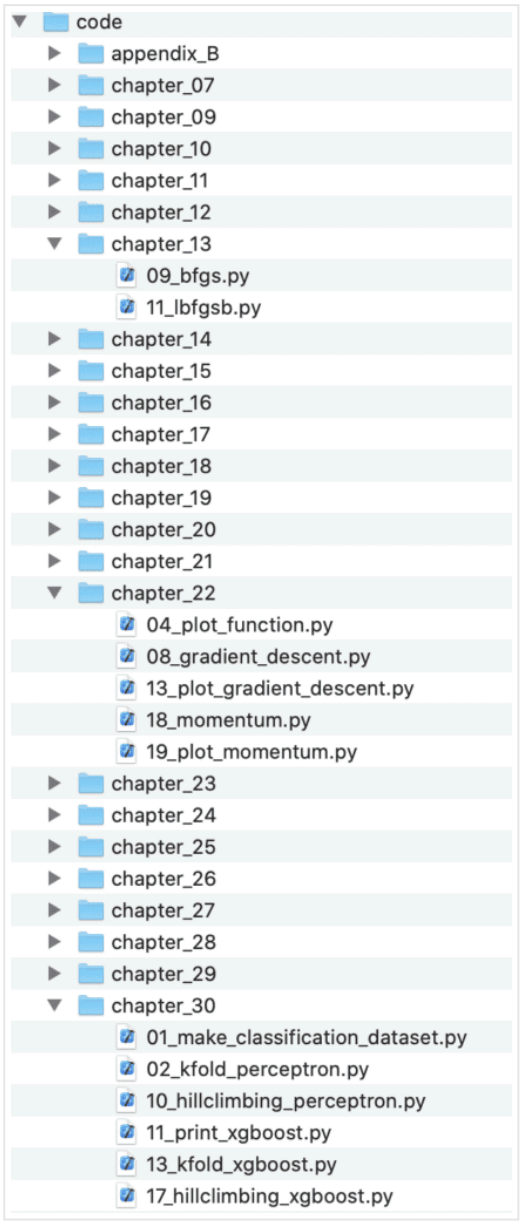
> You get one Python script (.py) for each example provided in the book.

This means that you can follow along and compare your answers to a known working implementation of each example in the provided Python files.

This helps a lot to speed up your progress when working through the details of a specific task, such as:

> Plotting a function to visually see its shape and the optimization progress.
> Random search and grid search.
> Nelder-Mead, BFGS and L-BFGS-B algorithms.
> Hill-climbing algorithms and its different variations.
> Genetic algorithms and the evolution strategies.
> Simulated annealing.
> Gradient descent, and its variations including momentum, AdaGrad, RMSProp, Adadelta, and Adam.
> Implementing regression from scratch.
> Training a neural network model from scratch.
> Tuning hyperparameters from scratch.

The provided code was developed in a text editor and is intended to be run on the command line. No special IDE or notebooks are required.



All code examples were designed and tested with Python 3.6+.

All code examples will run on modest and modern computer hardware and were executed on a CPU.

Python Technical Details

This section provides some technical details about the code provided with the book.

Python Version: You can use Python 3.6 or higher.
SciPy: You will use NumPy, SciPy, and scikit-learn APIs.
OS: You can use Windows, Linux, or MacOS.
Hardware: A standard modern workstation will do.
Editor: You can use a text editor and run the example from the command line.

Don’t have a Python environment?

No problem!

The appendix contains a step-by-step tutorial showing you exactly how to set up a Python machine learning environment.

About The Author

Hi, I'm Jason Brownlee. I run this site and I wrote and published this book.

I live in Australia with my wife and sons. I love to read books, write tutorials, and develop systems.

I have a computer science and software engineering background as well as Masters and PhD degrees in Artificial Intelligence with a focus on stochastic optimization.

I've written books on algorithms, won and ranked well in competitions, consulted for startups, and spent years in industry. (Yes, I have spend a long time building and maintaining REAL operational systems!)

I get a lot of satisfaction helping developers get started and get really good at applied machine learning.

I teach an unconventional top-down and results-first approach to machine learning where we start by working through tutorials and problems, then later wade into theory as we need it.

I'm here to help if you ever have any questions. I want you to be awesome at machine learning.

Download Your Sample Chapter

Do you want to take a closer look at the book? Download a free sample chapter PDF.

Enter your email address and your sample chapter will be sent to your inbox.

Click Here to Download Your Sample Chapter

You're Not Alone in Choosing Machine Learning Mastery

Trusted by Over 53,938 Practitioners

...including employees from companies like:

...students and faculty from universities like:

and many thousands more...

Absolutely No Risk with...

100% Money Back Guarantee

Plus, as you should expect of any great product on the market, every Machine Learning Mastery Ebook
comes with the surest sign of confidence: my gold-standard 100% money-back guarantee.

100% Money-Back Guarantee

If you're not happy with your purchase of any of the Machine Learning Mastery Ebooks, just email me within 90 days of buying, and ***I'll give you your money back*** ASAP.

No waiting. No questions asked. No risk.

…it's time to take the next step.

Bring Modern Optimization Algorithms to

16. Deep Learning for Time Series Forecasting
17. Generative Adversarial Networks with Python
18. Better Deep Learning
19. LSTM Networks with Python
20. XGBoost with Python
21. Ensemble Learning Algorithms with Python
22. Calculus for Machine Learning
23. Python for Machine Learning
24. Building Transformer Models with Attention
25. Deep Learning with PyTorch
26. Maximizing Productivity with ChatGPT
27. Machine Learning in OpenCV
28. The Beginner's Guide to Data Science
29. Mastering Digital Art with Stable Diffusion

(*includes all bonus source code*)

BUY NOW
FOR $597

That's $1,083.00 of Value!

(You save a massive $486.00)

All prices are in US Dollars (USD).

(1) Click the button. (2) Enter your details. (3) Download immediately.

Secure Payment Processing With SSL Encryption

Are you a Student, Teacher or Retiree?

Contact me about a discount.

Do you have any Questions?

See the FAQ.

What Are Skills in Machine Learning Worth?

Your boss asks you:

Hey, can you build a predictive model for this?

Imagine you had the skills and confidence to say:
"*YES!*"
...and follow through.

I have been there. It feels great!

How much is that worth to you?

The industry is demanding skills in machine learning.
The market wants people that can deliver results, not write academic papers.

Business knows what these skills are worth and are paying sky-high starting salaries.

A Data Scientists Salary Begins at:
$100,000 to **$150,000**.
A Machine Learning Engineers Salary is Even Higher.

What Are Your Alternatives?

You made it this far.
You're ready to take action.

But, what are your alternatives? What options are there?

(1) A Theoretical Textbook for $100+
...it's boring, math-heavy and you'll probably never finish it.

(2) An On-site Boot Camp for $10,000+
...it's full of young kids, you must travel and it can take months.

(3) A Higher Degree for $100,000+
...it's expensive, takes years, and you'll be an academic.

OR...

For the **Hands-On Skills** You Get...
And the **Speed of Results** You See...
And the **Low Price** You Pay...

Machine Learning Mastery Ebooks are Amazing Value!

And they work. That's why I offer the money-back guarantee.

You're A Professional

The field moves quickly, *...how long can you wait?*

You think you have all the time in the world, but...

- New methods are devised and algorithms change.
- New books get released and prices increase.

Bottom-up is Slow and Frustrating, *...don't you want a faster way?*

Can you really go on another day, week or month...

- Scraping ideas and code from incomplete posts.

- New graduates come along and jobs get filled.

Right Now is the Best Time to make your start.

- Skimming theory and insight from short videos.
- Parsing Greek letters from academic textbooks.

Targeted Training is your Shortest Path to a result.

Professionals Stay On Top Of Their Field
Get The Training You Need!

You don't want to fall behind or miss the opportunity.

Frequently Asked Questions

Customer Questions (78)

+ **Can I be a reseller for your books?**
+ **Can I exchange a book in a bundle?**
+ **Can I get a customized bundle of books (a la carte)?**
+ **Can I get a hard copy of your book?**
+ **Can I get a purchase order?**
+ **Can I get a refund for one book in a bundle?**
+ **Can I get a refund?**
+ **Can I get a sample of the book?**
+ **Can I get a tax invoice for my purchase?**
+ **Can I get an evaluation copy of a book?**
+ **Can I get Kindle or ePub versions of the books?**
+ **Can I get the ISBN for your book?**
+ **Can I get your books for free?**
+ **Can I have a discount?**

- Can I have an RFI / RFP / RFT / RFQ?
- Can I pay via debit card?
- Can I pay via WeChat Pay or Alipay?
- Can I print the PDF for my personal use?
- Can I see the table of contents?
- Can I send you a cheque, money order, western union, etc?
- Can I upgrade to the super bundle later?
- Can I use your material to teach my lecture or course?
- Can I white-label your books or content?
- Can your books be purchased elsewhere online or offline?
- Do your books provide exercises or assignments?
- Do I get a certificate of completion?
- Do I get new books for free if I buy the super bundle?
- Do I need to be a good programmer?
- Do you cover the theory and derivations?
- Do you have any sales, deals, or coupons?
- Do you have videos?
- Do you offer a guarantee?
- Do you ship to my country?
- Do you support TensorFlow 2?
- Does the LSTM book cover multivariate time series?
- How are the mini-courses different from the books?
- How are the two algorithm books different?
- How are your books different from the blog?
- How are your books different to other books on machine learning?

- How do I access the code provided with the book?
- How do I buy a book?
- How do I convert my currency to US dollars?
- How do I download my purchase?
- How do I get access to the bonuses?
- How do I use a discount coupon?
- How long do books take to ship?
- How long will a book take me to finish?
- How long will it take to get the book?
- In what order should I read your books?
- Is there a license for libraries?
- Is there a team or company license?
- Is there any digital rights management (DRM)?
- Is there errata or a change log for the books?
- What about delivery?
- What book should I start with?
- What books and bundles do you sell?
- What books are you writing next?
- What books have I already purchased?
- What currencies do you support?
- What if my download link expires?
- What is the difference between the LSTM and Deep Learning books?
- What is the difference between the LSTM and Deep Learning for Time Series books?
- What is the difference between the LSTM and the NLP books?
- What is your business or corporate tax number (e.g. ABN, ACN, VAT, etc.)

- What operating systems are supported in the books?
- What programming language is used in “Master Machine Learning Algorithms”?
- What software do you use to write your books?
- What version of Python is used?
- Where is my purchase?
- Why are some of the book chapters also on the blog?
- Why are your books so expensive?
- Why aren’t your books on Amazon?
- Why doesn’t my payment work?
- Why is there an additional small charge on my order?
- Why not give all of your books away for free?
- Will I get free updates to the books?
- Will you help me if I have questions about the book?
- Will your books get me a job?

Do you have another question?

Please contact me.