How to Learn Machine Learning
The Self-Starter Way

Follow me on LinkedIn for more:
Steve Nouri
https://www.linkedin.com/in/stevenouri/

Hello, and welcome!

In this guide, we're going to reveal **how you can get a world-class machine learning education for free.**

You don't need a fancy Ph.D in math. You don't need to be the world's best programmer. And you certainly don't need to pay $16,000 for an expensive "bootcamp."

Whether your goal is to become a data scientist, use ML algorithms as a developer, or add cutting-edge skills to your business analysis toolbox, you can pick up applied machine learning skills much faster than you might think.

1. Are you a self-starter?

Do you like to learn with hands-on projects? Are you driven and self-motivated? Can you commit to goals and see them through? If so, you'll love studying machine learning. You'll get to solve interesting challenges, tinker with fascinating algorithms, and build an incredibly valuable career skill.

2. Are you tired of seeing expensive courses and bootcamps?

We are too... That's why we put together this guide of completely free resources anyone can use to learn machine learning. The truth is that most paid courses out there recycle the same content that's already available online for free. We'll pull back the curtains and reveal where to find them for yourself.

3. Do you want a single page on the internet that will always be up-to-date?

Machine learning is a rapidly evolving field. That makes it exciting to learn, but materials can become outdated quickly. We're going to update this page regularly with the best resources to learn machine learning.

We've got a lot of great stuff you'll like, so let's dive right in!

This is exciting stuff!

Table of Contents

Introduction to Machine Learning:

WTF is Machine Learning?

Machine Badass (NOT Machine Learning)

Machine learning is about teaching computers how to learn from data to make decisions or predictions. For true machine learning, the computer must be able to learn to identify patterns without being explicitly programmed to.

It sits at the intersection of statistics and computer science, yet it can wear many different masks. You may also hear it labeled several other names or buzz words:

Data Science, Big Data, Artificial Intelligence, Predictive Analytics, Computational Statistics, Data Mining, Etc...

While machine learning does heavily overlap with those fields, it shouldn't be crudely lumped together with them. For example, machine learning is **one** tool for data science (albeit an essential one). It's also **one** use of infrastructure that can handle big data.

Here are some examples:

- **Supervised Learning** - Your email provider kindly places that sketchy email from the "Nigerian prince with $50,000 to deposit into an overseas bank account" into the spam folder.

- **Unsupervised Learning** - Marketing firms "kindly" use hundreds of behavior and demographic indicators to segment customers into targeted offer groups.

- **Reinforcement Learning** - A computer and camera within a self-driving car interact with the road and other cars to learn how to navigate a city.

Don't worry if some of those terms mean nothing to you. After you complete this guide, you'll be able to apply each of those techniques yourself! (Self-driving car not included.)

Self-driving car: NOT included in this guide!

Why Learn Machine Learning?

Have you ever wanted to take over the world with robot raccoons?...

Or program your own personal butler like J.A.R.V.I.S. from Iron Man?!...

Or crack the stock market and become a billionaire overnight??!!...

Well, sorry to be a party pooper... but you probably won't be able to do that with machine learning (*yet*). But there are still awesome reasons to learn machine learning! Here are a few:

Massive Global Demand

The demand for machine learning is booming all over the world. Entry salaries start from $100k – $150k. Data scientists, software engineers, and business analysts all benefit by knowing machine learning.

Data is Power

Data is transforming everything we do. All organizations, from startups to tech giants to Fortune 500 corporations, are racing to harness their data. Big *and* small data will continue to reshape technology and business.

It's Fun as Hell!

OK, we may be a bit biased, but ML is really damn cool. It has a unique blend of discovery, engineering, and business application that makes it one-of-a-kind. You'll have a ton of fun with this rich and vibrant field.

The Self-Starter Way

The self-starter way of mastering ML is to learn by **"doing shit."** (not the technical term).

Traditionally, students will first spend months or even years on the theory and mathematics behind machine learning. They'll get frustrated by the arcane symbols and formulas or get discouraged by the sheer volume of textbooks and academic papers to read.

Unless you want to devote yourself to Ph.D research, that's way overkill. For most people, the self-starter approach is superior to the academic approach for 3 reasons:

1. **You'll have more fun**. By cycling between theory, practice, and projects, you'll arrive at real results faster. This is a huge boost in morale.

2. **You'll build practical skills the industry demands.** Businesses don't care if you can derive proofs. They care if you can turn their data into gold.

3. **You'll build your portfolio along the way.** With hands-on projects, you'll conveniently build a portfolio you can show employers.

In a nutshell, the self-starter way is faster and more practical.

However, it definitely puts more responsibility in *your own hands* to follow through. Hopefully this guide will help you stay on track!

Here are the 4 steps to learning machine through self-study:

0 **Prerequisites**
Build a foundation of statistics, programming, and a bit of math.

1 **Sponge Mode**
Immerse yourself in the essential theory behind ML.

2 **Targeted Practice**
Use ML packages to practice the 9 essential topics.

3 **Machine Learning Projects**
Dive deeper into interesting domains with larger projects.

Free Self-Study Machine Learning Course:

Step 0: Prerequisites

Machine learning can appear intimidating without a gentle introduction to its prerequisites. You don't need to be a professional mathematician or veteran programmer to learn machine learning, but you do need to have the core skills in those domains.

The good news is that once you fulfill the prerequisites, the rest will be fairly easy. In fact, almost all of ML is about applying concepts from statistics and computer science to data.

Task: Make sure you are caught up to speed for at least programming and statistics.

Python for Data Science

You can't use machine learning unless you know how to program. Luckily, we have a free guide: How to Learn Python for Data Science, The Self-Starter Way

Statistics for Data Science

Understanding statistics, especially Bayesian probability, is essential for many machine learning algorithms. We have a free guide for you: How to Learn Statistics for Data Science, The Self-Starter Way

Math for Data Science

Original algorithm research requires a foundation in linear algebra and multivariable calculus. We have a free guide: How to Learn Math for Data Science, The Self-Starter Way

Step 1: Sponge Mode

Sponge mode is all about soaking in as much theory and knowledge as possible to give yourself a strong foundation.

Pictured: Spongebob (NOT Sponge Mode)

Now, some people may be wondering: *"If I don't plan to perform original research, why would I need to learn the theory when I can just use existing ML packages?"*

This is a reasonable question!

However, **learning the fundamentals is important for anyone who plans to apply machine learning in their work.** Here are 5 super practical reasons for learning ML theory. They span the entire modeling process:

1. **Planning and data collection.** Data collection can be an expensive and time consuming process. *What types of data do I need to collect? How much data do I need (hint: it's different depending on the model)? Is this challenge feasible?*

2. **Data assumptions and preprocessing.** Different algorithms have different assumptions about the input data. *How should I preprocess my data? Should I normalize it? Is my model robust to missing data? How about outliers?*

3. **Interpreting model results.** The notion that ML is a "black box" is simply false. Yes, not all results are directly interpretable, but you need to be able to diagnose your models to improve them. *How can I tell if my model is overfit or underfit? How do I explain these results to business stakeholders? How much room for improvement is left?*

4. **Improving and tuning your models.** You'll rarely reach the best model on your first try. You need to understand the nuances of different tuning parameters and regularization methods. *If my model is overfit, how can I remedy it? Should I spend more time on feature-engineering or on data collection? Can I ensemble my models?*

5. **Driving to business value.** ML is never done in a vacuum. If you don't truly understand the tools in your arsenal, you can't maximize their effectiveness. *Which outcome metrics are most important to optimize? Are there other algorithms that work better here? When is ML not the answer?*

Here's the great news... you *don't* need to have all the answers to these questions right from the start. In fact, the approach we recommend is to learn just enough theory to get started and not go astray. Then, you can build mastery over time by alternating between theory and practice.

1.1 Best Free Machine Learning Courses

These next two free courses are world-class (from Harvard and Stanford) resources for Sponge Mode.

Task: Complete at least one of the courses below.

Harvard's Data Science Course

End-to-end data science course. While there's less emphasis on ML than in Andrew Ng's course, you'll get more practice with the entire data science workflow from

Stanford's Machine Learning Course

This is the famous course taught by Andrew Ng, and it's the gold standard when it comes to learning machine learning theory. These videos really clear up the core concepts

data collection to analysis. (Course Homepage | Lecture Videos and Slides | Homework Assignments)

behind ML. *If you only have time for 1 course, we recommend this one.* (Course Videos)

1.2 Keys to Success

Here are a few keys to success for this step:

A.) Pay attention to the big picture and always ask "why."

Every time you're introduced to a new concept, ask "why." Why use a decision tree instead of regression in some cases? Why regularize parameters? Why split your dataset? When you understand *why* each tool is used, you'll become a true machine learning *practitioner*. For example, by the end of this step, you should know when to preprocess your data, when to use supervised vs. unsupervised algorithms, and methods for preventing model overfitting.

B.) Accept that you will not remember everything.

Don't stress about taking insane notes or reviewing everything 3 times. Accept that you'll need to cycle back and review concepts as you encounter them in the wild.

C.) Keep moving and don't be discouraged.

Try to avoid dwelling on any topic for too long. Some concepts can't be explained easily, even by the best professors. Your confusion will clear up once you start applying them in practice.

D.) Videos are more effective than textbooks.

From our experience, textbooks can be great reference tools, but they often omit the vital color commentary surrounding key concepts. We strongly recommend video lectures during Sponge Mode.
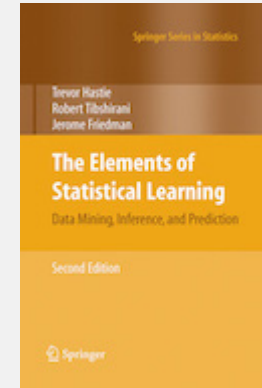
1.3 Free Reference Textbooks

Next, we have free (legal) PDFs of 2 classic textbooks in the industry.

Task: Download the free PDFs for your future reference.
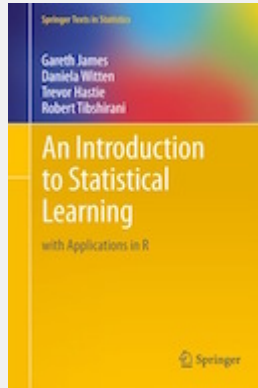
An Introduction to Statistical Learning

Gentler introduction than Elements of Statistical Learning. Recommended for everyone. (PDF)

Elements of Statistical Learning

Rigorous treatment of ML theory and mathematics. Recommended for ML researchers. (PDF)

Back to Table of Contents

Step 2: Targeted Practice

After Sponge Mode, you've probably already gotten a healthy dose of practice.

Now it's time to take that practice to the next level.

Step 2: Targeted Practice is all about using specific, deliberate exercises to hone your skills. The goal of this step is threefold:

1. **Practice the entire machine learning workflow:** Data collection, cleaning, and preprocessing. Model building, tuning, and evaluation.

2. **Practice on real datasets:** You'll start to build intuition around which types of models are appropriate for which types challenges.

3. **Deep dive on individual topics:** For example, in Step 1, you learned about clustering algorithms. In Step 2, you'll apply different types of clustering algorithms on datasets to see which perform the best.

After this step, you'll be ready to tackle bigger projects without feeling overwhelmed.

2.1 - The 9 Essential Topics

Machine learning is a broad and rich field. There are applications for almost any industry. It's easy to get flustered by all there is to learn. Plus, it's also easy to get lost in the weeds of individual models and lose sight of the big picture.

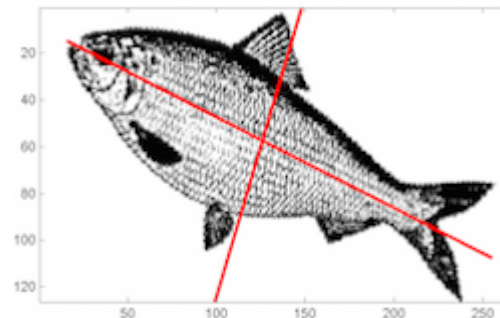Therefore, we've broken the essentials into the following 9 topics.

These are building block topics that collectively represent the simple value proposition of machine learning: **taking data and transforming it into something useful.**
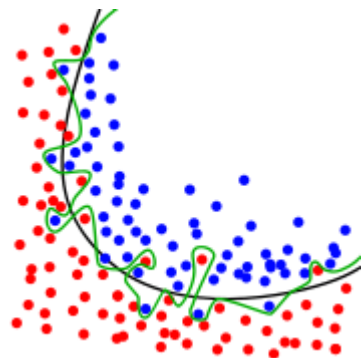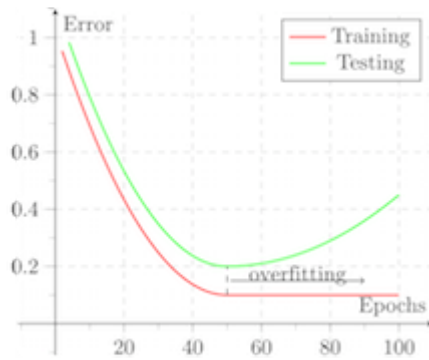
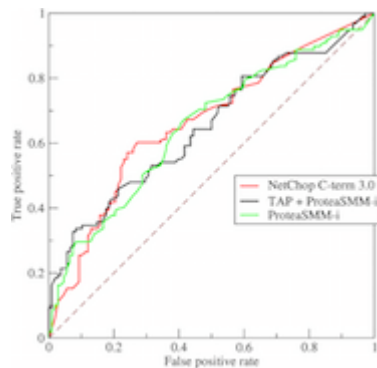

The Big Picture
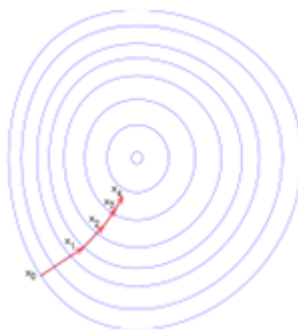
Optimization

Data Preprocessing

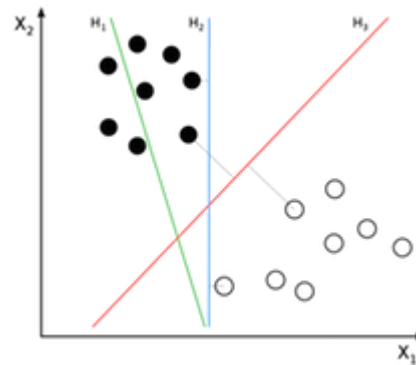Essential ML theory, such as the Bias-Variance tradeoff.

Algorithms for finding the best parameters for a model.

Dealing with missing data, skewed distributions, outliers, etc.

Sampling & Splitting

How to split your datasets to tune parameters and avoid overfitting.

Supervised Learning

Learning from labeled data using classification and regression models.
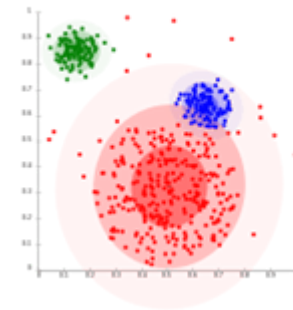
Unsupervised Learning

Learning from unlabeled data using factor and cluster analysis models.

Model Evaluation

Making decisions based on various performance metrics.

Ensemble Learning

Combining multiple models for better performance.
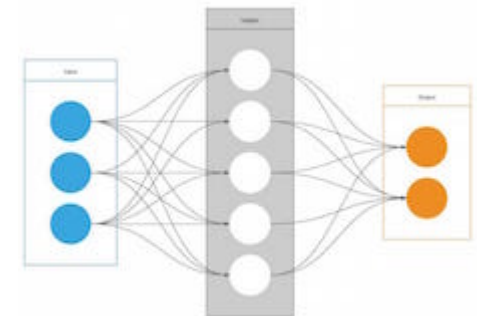
Business Applications

How machine learning can help different types of businesses.

2.2 - Tools of the Trade

For this step, we strongly recommend that you start with out-of-the-box algorithm implementations for two reasons.

First, this is how most ML is performed in the industry. Sure, there will be times when you'll need to research original algorithms or develop them from scratch, but prototyping always starts with existing libraries.

Second, you'll get the chance to practice the entire ML workflow without spending too much time on any one portion of it. This will give you an invaluable "big picture intuition."

Depending on your programming language of choice, you have 2 excellent options.

Task: Complete the Quickstart guide for one of the libraries below.

Python: Scikit-Learn

Scikit-learn, or sklearn, is the gold standard Python library for general purpose machine learning. It does almost everything, and it has implementations of all the common algorithms.

Scikit-Learn Tutorial, Wine Snob Edition

R: Caret

Caret is love. Caret is life. Caret is a library that provides a unified interface for many different model packages in R. It also includes functions for preprocessing, data splitting, and model evaluation, making it a complete end-to-end solution.

Quickstart Webinar

2.3 - Datasets for Practice

For this step, you'll need datasets to practice building and tuning models.

Again, the point of Step 2: Targeted Practice is to **take the theory that's floating around in your mind after Step 1: Sponge Mode and put it into code.**

Much of the art in data science and machine learning lies in dozens of micro-decisions you'll make to solve each problem. This is the perfect time to practice making those micro-decisions and evaluating the consequences of each.

Task: Pick 5-10 datasets from the options below. We recommend starting with the UCI Machine Learning Repository. For example, you can pick 3 datasets each for regression, classification, and clustering.

Task: For each dataset, try at least 3 different modeling approaches using Scikit-Learn or Caret. Think about the following questions:

- What types of preprocessing do you need to perform for each dataset?

- Do you need to reduce dimensions or perform feature selection? If so, what methods can you use?

- How should you sample or split your dataset?

- How do you know if your model is overfit?

- What types of performance metrics should you use?

- How do different tuning parameters affect your model results?

- Can you ensemble to get better results?

- (*For clustering*) Do your clusters appear intuitive?

We also have a curated list of some of our favorite datasets for practice and projects.

UCI Machine Learning Repo	**Kaggle**	**Data.gov**
This is an incredible collection of over 350 different datasets specifically curated for practicing machine learning. You can search by task (i.e. regression, classification, or clustering), industry, dataset size, and more. (Go to website)	Kaggle.com is most famous for hosting data science competitions, but the site also houses over 180 community datasets for fun topics ranging from Pokemon data to European Soccer matches. (Go to website)	If you're looking for social science or government-related datasets, look no further than Data.gov, a collection of the U.S. government's open data. You can search over 190,000 datasets. (Go to website)

Step 3: Machine Learning Projects

Alright, now comes the *really* fun part! Up to now, we've covered prerequisites, essential theory, and targeted practice. We're now ready to dive into some bigger projects.

The goal of this step is to **practice integrating machine learning techniques into complete, end-to-end analyses.**

Task: Complete the projects below. The order is up to you, but we ordered them by difficulty (easiest first).

3.1 - Titanic Survivor Prediction

The Titanic Survivor Prediction challenge is an incredibly popular project for practicing machine learning. In fact, it's the most popular competition on Kaggle.com.

We love this project as a starting point because there's a wealth of great tutorials out there. You can take a peek into the minds of more experienced data scientists and see how they approach data exploration, feature engineering, and model tuning.

The Titanic is sinking!

Python Tutorials

- Four-Part Tutorial by Kaggle - Detailed tutorial that starts from cleaning and exploring the data. We really like this tutorial because it teaches you how to properly preprocess and wrangle your data properly before using sklearn.

- Tutorial and iPython Notebooks by Pycon UK - Great tutorial that's presented in iPython Notebook. It has excellent appendices on cross-validation and visualization.

R Tutorials

- Binary Outcome Modeling Tutorial - Walks through a couple different models in R using the caret package. This tutorial nicely summarizes the predictive modeling process from end-to-end.

- An "Irresponsibly" Fast Tutorial - Bare bones tutorial that completely skips the theory. Useful as another perspective (and it shows random forests in action).

3.2 - Algorithm from Scratch

There's nothing that pushes your understanding quite like writing an algorithm from scratch. They say the devil's in the details, and here's where that really rings true.

We recommend starting with something simple, like logistic regression, decision trees, or k-nearest neighbors.

This project will also give you invaluable practice in translating math into code. This skill will be very handy when you eventually need to use the latest research from academia in your work.

If you get stuck, here are some tips:

- Wikipedia is a great resource for this project because it has pseudo-code for many common algorithms.

- For inspiration, try looking at the source code from existing ML packages.

- Break your algorithm into pieces. Write separate functions for sampling, gradient descent, etc.

- Start simple. Implement a decision tree before trying to write a random forest.

She's only a few years away from learning machine learning...

3.3 - Pick a Fun Project or Interesting Domain

You wouldn't be a self-starter if you didn't have curiosity and ideas. By now, you're probably itching to get started (or have already started) on some grand idea that you've been mulling over.

This is honestly the best part about learning machine learning. It's such a powerful tool that once you start to understand, so many ideas will come to you.

The good news is that if you've been following along, then you're more than ready to jump in. Go forth, and reap the fruits of your labor!

We'll also keep a list of project ideas here for inspiration:

Project Ideas

- 8 Fun Machine Learning Projects for Beginners

Great Job! (So Far...)

Congratulations on reaching the end of the self-study guide!

Here's some great news: If you've followed along and completed all the tasks, you're better at applied machine learning than 90% of the people out there claiming to be data scientists. You have an awesome skillset that employers will drool over.

Now, here's some better news: There's still much to learn! For example, deep learning, computer vision, and natural language processing are a few of the fascinating, cutting-edge subfields that await you.

The key to becoming the best data scientist or machine learning engineer you can be is to **never stop learning.** Welcome to the start of your journey in this dynamic, exciting field!

So great job! *So far...*

Bonus Goodies:

Top 10 Tips for Beginners

If you've chosen to seriously study machine learning, then congratulations! You have a fun and rewarding journey ahead of you.

Here are 10 tips that every beginner should know:

1. Set concrete goals or deadlines.

Machine learning is a rich field that's expanding every year. It can be easy to go down rabbit holes. Set concrete goals for yourself and keep moving.

2. Walk before you run.

You might be tempted to jump into some of the newest, cutting edge sub-fields in machine learning such as deep learning or NLP. Try to stay focused on the core concepts at the start. These advanced topics will be much easier to understand once you've mastered the core skills.

3. Alternate between practice and theory.

Practice and theory go hand-in-hand. You won't be able to master theory without applying it, yet you won't know what to do without the theory.

4. Write a few algorithms from scratch.

Once you've had some practice applying algorithms from existing packages, you'll want to write a few from scratch. This will take your understanding to the next level and allow you to customize them in the future.

5. Seek different perspectives.

The way a statistician explains an algorithm will be different from the way a computer scientist explains it. Seek different explanations of the same topic.

6. Tie each algorithm to value.

For each tool or algorithm you learn, try to think of ways it could be applied in business or technology. This is essential for learning how to "think" like a data scientist.

7. Don't believe the hype.

Machine learning is *not* what the movies portray as artificial intelligence. It's a powerful tool, but you should approach problems with rationality and an open mind. ML should just be one tool in your arsenal!

8. Ignore the show-offs.

Sometimes you'll see people online debating with lots of math and jargon. If you don't understand it, don't be discouraged. What matters is: Can you use ML to add value in some way? And the answer is yes, you absolutely can.

9. Think "inputs/outputs" and ask "why."

At times, you might find yourself lost in the weeds. When in doubt, take a step back and think about how data inputs and outputs piece together. Ask "why" at each part of the process.

10. Find fun projects that interest you!

Rome wasn't built in a day, and neither will your machine learning skills be. Pick topics that interest you, take your time, and have fun along the way.

More Resources

We'll be keeping this section updated with the best additional resources for learning machine learning, so keep this page bookmarked (links here open in a new tab).

Other posts you may like:

- 21 Must-Know Machine Learning Interview Questions & Answers

- 5 Tasty Python Web Scraping Libraries

- 5 Heroic Python NLP Libraries

- 5 Genius Python Deep Learning Libraries

Awesome Machine Learning TED Talks:

- Jeremy Howard: The wonderful and terrifying implications of computers that can learn

- Blaise Agüera y Arcas: How computers are learning to be creative

- Anthony Goldbloom: The jobs we'll lose to machines — and the ones we won't

Follow me on LinkedIn for more:
Steve Nouri
https://www.linkedin.com/in/stevenouri/