

Prerequisites and prework

Is Machine Learning Crash Course right for you?

I have little or no machine learning background.

I have some background in machine learning, but I'd like a more current and complete understanding.

I have practical experience applying machine learning concepts to work with data and build models.

I am looking for tutorials on how to use ML APIs like Keras.

Please read through the following [Prework](#) (#prework) and [Prerequisites](#) (#prerequisites) sections before beginning Machine Learning Crash Course, to ensure you are prepared to complete all the modules.

Prework

Before beginning Machine Learning Crash Course, do the following:

1. If you're new to machine learning, take [Introduction to Machine Learning](#) (/machine-learning/intro-to-ml). This short self-study course introduces fundamental machine learning concepts.
2. If you are new to [NumPy](#) (<https://numpy.org>), do the [NumPy Ultraquick Tutorial](#) (https://colab.research.google.com/github/google/eng-edu/blob/main/ml/cc/exercises/numpy_ultraquick_tutorial.ipynb?utm_source=mlcc&utm_campaign=colab-external&utm_medium=referral&utm_content=mlcc-prework&hl=en)

Colab exercise, which provides all the NumPy information you need for this course.

3. If you are new to pandas (<https://pandas.pydata.org/>), do the pandas UltraQuick Tutorial (https://colab.research.google.com/github/google/eng-edu/blob/main/ml/cc/exercises/pandas_dataframe_ultraquick_tutorial.ipynb?utm_source=mlcc&utm_campaign=colab-external&utm_medium=referral&utm_content=mlcc-prework&hl=en)

Colab exercise, which provides all the pandas information you need for this course.

Programming exercises run directly in your browser (no setup required!) using the Colaboratory (<https://colab.research.google.com>) platform. Colaboratory is supported on most major browsers, and is most thoroughly tested on desktop versions of Chrome and Firefox.

Prerequisites

Machine Learning Crash Course does not presume or require any prior knowledge in machine learning. However, to understand the concepts presented and complete the exercises, we recommend that students meet the following prerequisites:

- You must be comfortable with variables, linear equations, graphs of functions, histograms, and statistical means.
- You should be a good programmer. Ideally, you should have some experience programming in Python (<https://www.python.org/>) because the programming exercises are in Python. However, experienced programmers without Python experience can usually complete the programming exercises anyway.

The following sections provide links to additional background material that is helpful.

Algebra

- variables (<https://www.khanacademy.org/math/algebra/x2f8bb11595b61c86:foundation-algebra/x2f8bb11595b61c86:intro-variables/v/what-is-a-variable>)
- coefficients (<https://www.khanacademy.org/math/cc-sixth-grade-math/cc-6th-equivalent-exp/cc-6th-parts-of-expressions/v/expression-terms-factors-and-coefficients>)
- and functions (<https://www.khanacademy.org/math/algebra-home/alg-functions>)

- linear equations (https://wikipedia.org/wiki/Linear_equation) such as $y = b + w_1 x_1 + w_2 x_2$
- logarithms (<https://wikipedia.org/wiki/Logarithm>), and logarithmic equations such as $y = \ln(1 + e^z)$
- sigmoid function (https://wikipedia.org/wiki/Sigmoid_function)

Linear algebra

- tensor and tensor rank (<https://www.tensorflow.org/guide/tensor>)
- matrix multiplication (https://wikipedia.org/wiki/Matrix_multiplication)

Trigonometry

- tanh (<https://reference.wolfram.com/language/ref/Tanh.html>) (discussed as an activation function (https://developers.google.com/machine-learning/glossary#activation_function); no prior knowledge needed)

Statistics

- mean, median, outliers (<https://www.khanacademy.org/math/probability/data-distributions-a1/summarizing-center-distributions/v/mean-median-and-mode>), and standard deviation (https://wikipedia.org/wiki/Standard_deviation)
- ability to read a histogram (<https://wikipedia.org/wiki/Histogram>)

Calculus (*optional, for advanced topics*)

- concept of a derivative (<https://wikipedia.org/wiki/Derivative>) (you won't have to actually calculate derivatives)
- gradient (<https://www.khanacademy.org/math/multivariable-calculus/multivariable-derivatives/gradient-and-directional-derivatives/v/gradient>) or slope
- partial derivatives (https://wikipedia.org/wiki/Partial_derivative) (which are closely related to gradients)

- [chain rule](https://wikipedia.org/wiki/Chain_rule) (https://wikipedia.org/wiki/Chain_rule) (for a full understanding of the [backpropagation algorithm](https://developers.google.com/machine-learning/crash-course/backprop-scroll/) (<https://developers.google.com/machine-learning/crash-course/backprop-scroll/>) for training neural networks)

Python Programming

The following Python basics are covered in [The Python Tutorial](https://docs.python.org/3/tutorial/)

(<https://docs.python.org/3/tutorial/>):

- [defining and calling functions](https://docs.python.org/3/tutorial/controlflow.html#defining-functions) (<https://docs.python.org/3/tutorial/controlflow.html#defining-functions>), using positional and [keyword](https://docs.python.org/3/tutorial/controlflow.html#keyword-arguments) (<https://docs.python.org/3/tutorial/controlflow.html#keyword-arguments>) parameters
- [dictionaries](https://docs.python.org/3/tutorial/datastructures.html#dictionaries) (<https://docs.python.org/3/tutorial/datastructures.html#dictionaries>), [lists](https://docs.python.org/3/tutorial/introduction.html#lists) (<https://docs.python.org/3/tutorial/introduction.html#lists>), [sets](https://docs.python.org/3/tutorial/datastructures.html#sets) (<https://docs.python.org/3/tutorial/datastructures.html#sets>) (creating, accessing, and iterating)
- [for loops](https://docs.python.org/3/tutorial/controlflow.html#for-statements) (<https://docs.python.org/3/tutorial/controlflow.html#for-statements>), **for** loops with multiple iterator variables (e.g., `for a, b in [(1,2), (3,4)]`)
- [if/else conditional blocks](https://docs.python.org/3/tutorial/controlflow.html#if-statements) (<https://docs.python.org/3/tutorial/controlflow.html#if-statements>) and [conditional expressions](https://docs.python.org/2.5/whatsnew/pep-308.html) (<https://docs.python.org/2.5/whatsnew/pep-308.html>)
- [string formatting](https://docs.python.org/3/tutorial/inputoutput.html#old-string-formatting) (<https://docs.python.org/3/tutorial/inputoutput.html#old-string-formatting>) (e.g., `'%.2f' % 3.14`)
- variables, assignment, [basic data types](https://docs.python.org/3/tutorial/introduction.html#using-python-as-a-calculator) (<https://docs.python.org/3/tutorial/introduction.html#using-python-as-a-calculator>) (`int`, `float`, `bool`, `str`)

A few of the programming exercises use the following more advanced Python concept:

- [list comprehensions](https://docs.python.org/3/tutorial/datastructures.html#list-comprehensions) (<https://docs.python.org/3/tutorial/datastructures.html#list-comprehensions>)

Bash Terminal and Cloud Console

To run the programming exercises on your local machine or in a cloud console, you should be comfortable working on the command line:

- [Bash Reference Manual](https://tiswww.case.edu/php/chet/bash/bashref.html) (https://tiswww.case.edu/php/chet/bash/bashref.html)
- [Bash Cheatsheet](https://github.com/LeCoupa/awesome-cheatsheets/blob/master/languages/bash.sh)
(https://github.com/LeCoupa/awesome-cheatsheets/blob/master/languages/bash.sh)
- [Learn Shell](http://www.learnshell.org/) (http://www.learnshell.org/)

[Help Center](https://support.google.com/machinelearningeducation) (https://support.google.com/machinelearningeducation)

[Next](#)

[Exercises](/machine-learning/crash-course/exercises) (/machine-learning/crash-course/exercises) →

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (https://creativecommons.org/licenses/by/4.0/), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (https://www.apache.org/licenses/LICENSE-2.0). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (https://developers.google.com/site-policies). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2024-10-09 UTC.