

ML Roadmap

Machine Learning Road-Map

This is a list of resources to help new-comers to get started with Machine Learning. We divide it into various sections depending on amount of knowledge one has.

Pre-requisites

- **Calculus** : Basic Course for Calculus required for Machine Learning.
- **Linear Algebra**:
 - [Khan Academy](#) : Basic concepts of Linear Algebra.
 - [3Blue1Brown](#) : Good resource to "visualise" Linear Algebra.
- **Probability & Statistics**:
 - [Khan Academy](#) : Basic Probability & Statistics Concept.
 - [Machine Learning Mastery](#) : Probability from a Machine Learning prospective.
- **Python** :
 - Theory : [Udemy](#)
 - Coding Practice: [Hackerrank](#)
 - Anaconda Setup:
 - [Windows Setup](#) : Facilitates virtual environments to keep the libraries in one place
 - [Ubuntu Setup](#) : For Ubuntu (make sure your version matches), else google for the same.
 - [Cheat Sheet](#)
- **Python Libraries & Tools** :
 - **NumPy** : NumPy proficiency is a must for writing as well as understanding ML codes.
 - [Documentation](#)
 - [Tutorial](#)
 - [Practice \(Coding\)](#)
 - **Pandas** : It is a software library for data analysis and manipulation.
 - [Documentation](#)
 - [Tutorial](#) - Things you would wish you knew earlier.
 - [Practice \(Coding\)](#)

- **Matplotlib** : Being able to plot and visualise things is a must in ML, so be sure to master it!
 - [Documentation](#)
 - [Tutorial](#)
 - [Cheat sheet](#)
-

Some Blogs to Follow

These are some informative blogs in ML & AI, which you can refer to understand any particular concept or topic:

1. [Medium](#)
 2. [Towards Data Science](#)
 3. [Machine Learning Mastery](#)
 4. [Paperspace Blogs](#) : This is really an active site with really good & latest tutorials and concepts.
 5. [Distill](#) - This is a bit advanced blog, but understandable once you complete Intermediate Section. But rest assured, this blog is a treasure who are inclined towards development & research!
 6. [Christopher Olah's Blog](#) - A really nice & well explained blog.
 7. [Andrej Karpathy Blog](#)
 8. [The Batch](#) - [deeplearning.ai](#)'s Weekly magazine on advancements in Machine Learning & AI.
-

Books

There are tremendous number of books available on internet, but we have shortlisted those which are **sufficient** to understand any topic:

1. [Deep Learning](#) by Yoshua Bengio, Ian Goodfellow and Aaron Courville.
2. [Machine Learning - A Probabilistic Perspective](#) by Kevin Murphy (takes you from very basics of Probability to Modern State of Art methods in ML).
3. [Pattern Recognition And Machine Learning](#) by Christopher M. Bishop.

4. [Information Theory, Inference, and Learning Algorithms](#) by David J.C. MacKay (gives essence of Bayesian Inference and how it connects with Neural Networks).
 5. [Deep Learning Methods and Applications](#) Li Deng and Dong Yu.
-

Beginner

- [What is machine learning?](#)
- Try [this](#) out and wonder how could this work!
- [What's a neural network?](#)
- Play with [this](#) and try developing some intuition.
- [Basic Machine Learning](#) :
 - This is Andrej Ng's famous course on Machine Learning, where he covers mostly the Classical ML and also touches a bit of Deep Learning.
 - Unfortunately, the assignments in this course are made in Octave, find the same assignments in [python here](#).

Remember the best way to make sure that you master something is to do it in unfavourable circumstances.

- [Stat-Quest](#) : This is another nice channel where you can look at good explanation of many ML Algorithm.
- [Sci-Kit Learn Implementation](#) : Sci-Kit Learn is an essential python library to implement classical ML Algorithms, it contains almost all the needed algorithms, so make sure to get used to it!

[Multi-Layer Perceptron](#) - Acquainted with regression? Now dive in to find what a Multi Layer Perceptron can do that a simple perceptron (logistic regression) cannot!

[Neural Network Playground](#) - Just play with this framework. It is very helpful for visualising and building simple neural networks on the go and seeing the results. It will also give you a "feel" of how Neural Network learns and what is the effect of different parameters on its training.

- Use [this link](#) as a guide to various terminologies used in deep learning at length
- [Going deeper into the neural networks!](#) (Chap 1 and 2 have basics, 3, 4, 5 and 6 are for those who have good grip on basics)

- **Deep Learning :**

- [Deep Learning Specialisation](#) : This is the first course out of 5 in series of Deep Learning Specialisation available on Coursera. It contains basic Deep Learning Theory & Implementation. This is a must do course if you are interested in Deep Learning, do all the assignments (don't skip it) they are very essential to understand roots of DL.
- [Hyper-Parameter Tuning](#) : This is a bit implementation based short course, one should do if interested in real time implementation and better understanding of how Hyper-parameters play important role in performance of a Neural Network. You can manage to implement DL even without this course (it's upto you :)).
- [3Blue1Brown](#) : A very good series of videos to give intuition of what Neural Networks are and how they work.

- **Implementation** : One should choose between Tensorflow and Pytorch to implement their Deep Learning models. The thumb rule according to many resources is that - If you are into product deployment, real-time implementation, Go with Tensorflow. If you are into Research and want to try different models, Go with Pytorch.

- **Tensorflow** : A free and open source software library for dataflow and differential programming
 - [Udacity Course](#) : Course on Tensorflow.
 - [Tensorflow Specialisation](#) : It contains 4 courses, we prefer take the basic one so that you are familiar with the basics, then apply it according to your need, if you stumble across any problem, you can always refer to Stackoverflow and similar websites for solution. If you are working in a particular area though (eg. Computer Vision, NLP or Time Series Prediction) then it would be useful to take one of the other specialised courses.
 - [Tensorflow Deployment](#) : It contains 4 courses, all of these courses are deployment based, i.e, made for someone who is looking to deploy his/her model in real-time, say on an app or a website. It is very essential for developers, or someone who is looking to clear GSoC in Deep Learning.

- [Tensorflow Roadmap](#) : A Book with all the topic wise tutorial of Tensorflow.
One can use it as a shorthand resource if needed to refer to something.
- **Pytorch** :
 - [Udacity](#) : Course on Pytorch.
 - [Tutorials](#) : Official Pytorch tutorials.
- **Keras** :
 - [Introduction](#)
 - [Documentation](#)
 - [Cheat sheet](#)
- **keras Vs tf.keras**

It's completely fine if you don't understand in the very first time;) So, don't worry!

Intermediate

Suit yourselves!

The following courses are optional, but we encourage you to go through the descriptions and choose according to your taste:)

- [Statistical Machine Learning](#) : This is course based on pure Statistical Machine Learning. It is quite different from Deep Learning! Beauty of statistical ML is nothing is black-box here! Everything is in-front of you. Mathematics Enthusiasts (specially those who love Probability & Statistics) should try this out.
- [In-depth Deep Learning](#) : This is a bit in-depth course for Deep Learning given by pioneers of DL, this is preferred for someone interested in research sphere in DL. It also contains few research based tutorials.
- [Fast.ai Practical Deep Learning for Coders](#) : If you are looking for a course which explains all the theory with equal emphasis on the implementation, then this is a perfect course for you. In fact you can consider this as an Implementation course, with theory. Reason to add this course is that, the lecturer here explains everything with subtle clarity.

Convolutional Neural Networks, here you go!

Before proceeding further, make sure you have completed the Classical [Machine Learning](#) course by Andrew Ng atleast till week 5 and the [Neural networks and deep learning](#) course or its equivalent!

- Prior to diving into one of the most enthusiastic fields of machine learning i.e. Computer vision, having a sound knowledge about optimizers and regularization methods is extremely important. So if you haven't yet come across the [Improving Deep Neural Networks](#) course, follow the given link.
- [Convolutional Neural Networks](#) : This is a must do course if you are interested in CNNs, Image processing or Computer Vision (this is one of the really fascinating courses, where you will learn to do some "cool" stuffs too!).
- Didn't get how the back propagation worked? Follow [this](#) link for better understanding!
- For those willing to know the much deeper theory, [go](#) be an expert! For those preferring videos, click [here](#).

Some important topics/blogs

- [cs231n](#) : This video has various computer vision tasks for instance object detection, semantic segmentation, etc.
- Object detection [overview](#)
- [YOLO](#)

Now you would have got an overview of some advanced models like VGG and RESNET. If you are comfortable with the above topics you may start with :

- [Sequence Models](#) : Course on Sequence Models (Recurrent Neural Networks (RNNs), LSTMs, GRUs, etc). Initially this course was said to be essential for Natural Language Processing (NLP), but nowadays sequential models have found there role in every field of DL, be it images or 1D predictive data.