**Machine Learning Resources**

[Machine Learning is Fun!](https://medium.com/@ageitgey/machine-learning-is-fun-80ea3ec3c471" \t "_blank) (medium.com/@ageitgey)

Machine Learning Crash Course: [Part I](https://ml.berkeley.edu/blog/2016/11/06/tutorial-1/), [Part II](https://ml.berkeley.edu/blog/2016/12/24/tutorial-2/), [Part III](https://ml.berkeley.edu/blog/2017/02/04/tutorial-3/) (Machine Learning at Berkeley)

[An Introduction to Machine Learning Theory and Its Applications: A Visual Tutorial with Examples](https://www.toptal.com/machine-learning/machine-learning-theory-an-introductory-primer) (toptal.com)

[A Gentle Guide to Machine Learning](https://monkeylearn.com/blog/a-gentle-guide-to-machine-learning/) (monkeylearn.com)

[Which machine learning algorithm should I use?](https://blogs.sas.com/content/subconsciousmusings/2017/04/12/machine-learning-algorithm-use/) (sas.com)

**Activation and Loss Functions**

[Sigmoid neurons](http://neuralnetworksanddeeplearning.com/chap1.html#sigmoid_neurons) (neuralnetworksanddeeplearning.com)

[What is the role of the activation function in a neural network?](https://www.quora.com/What-is-the-role-of-the-activation-function-in-a-neural-network) (quora.com)

[Comprehensive list of activation functions in neural networks with pros/cons](https://stats.stackexchange.com/questions/115258/comprehensive-list-of-activation-functions-in-neural-networks-with-pros-cons) (stats.stackexchange.com)

[Activation functions and it’s types-Which is better?](https://medium.com/towards-data-science/activation-functions-and-its-types-which-is-better-a9a5310cc8f) (medium.com)

[Making Sense of Logarithmic Loss](http://www.exegetic.biz/blog/2015/12/making-sense-logarithmic-loss/) (exegetic.biz)

[Loss Functions](http://cs231n.github.io/neural-networks-2/#losses) (Stanford CS231n)

[L1 vs. L2 Loss function](http://rishy.github.io/ml/2015/07/28/l1-vs-l2-loss/) (rishy.github.io)

[The cross-entropy cost function](http://neuralnetworksanddeeplearning.com/chap3.html#the_cross-entropy_cost_function) (neuralnetworksanddeeplearning.com)

**Bias**

[Role of Bias in Neural Networks](https://stackoverflow.com/questions/2480650/role-of-bias-in-neural-networks/2499936#2499936) (stackoverflow.com)

[Bias Nodes in Neural Networks](http://makeyourownneuralnetwork.blogspot.com/2016/06/bias-nodes-in-neural-networks.html) (makeyourownneuralnetwork.blogspot.com)

[What is bias in artificial neural network?](https://www.quora.com/What-is-bias-in-artificial-neural-network) (quora.com)

**Perceptron**

[Perceptrons](http://neuralnetworksanddeeplearning.com/chap1.html#perceptrons) (neuralnetworksanddeeplearning.com)

[The Perception](http://natureofcode.com/book/chapter-10-neural-networks/#chapter10_figure3) (natureofcode.com)

[Single-layer Neural Networks (Perceptrons)](http://computing.dcu.ie/%7Ehumphrys/Notes/Neural/single.neural.html) (dcu.ie)

[From Perceptrons to Deep Networks](https://www.toptal.com/machine-learning/an-introduction-to-deep-learning-from-perceptrons-to-deep-networks) (toptal.com)

**Regression**

[Introduction to linear regression analysis](http://people.duke.edu/%7Ernau/regintro.htm) (duke.edu)

[Linear Regression](http://ufldl.stanford.edu/tutorial/supervised/LinearRegression/) (ufldl.stanford.edu)

[Linear Regression](http://ml-cheatsheet.readthedocs.io/en/latest/linear_regression.html) (readthedocs.io)

[Logistic Regression](http://ml-cheatsheet.readthedocs.io/en/latest/logistic_regression.html) (readthedocs.io)

[Simple Linear Regression Tutorial for Machine Learning](http://machinelearningmastery.com/simple-linear-regression-tutorial-for-machine-learning/) (machinelearningmastery.com)

[Logistic Regression Tutorial for Machine Learning](http://machinelearningmastery.com/logistic-regression-tutorial-for-machine-learning/) (machinelearningmastery.com)

[Softmax Regression](http://ufldl.stanford.edu/tutorial/supervised/SoftmaxRegression/) (ufldl.stanford.edu)

**Gradient Descent**

[Learning with gradient descent](http://neuralnetworksanddeeplearning.com/chap1.html#learning_with_gradient_descent) (neuralnetworksanddeeplearning.com)

[Gradient Descent](http://iamtrask.github.io/2015/07/27/python-network-part2/) (iamtrask.github.io)

[How to understand Gradient Descent algorithm](http://www.kdnuggets.com/2017/04/simple-understand-gradient-descent-algorithm.html) (kdnuggets.com)

[An overview of gradient descent optimization algorithms](http://sebastianruder.com/optimizing-gradient-descent/) (sebastianruder.com)

[Optimization: Stochastic Gradient Descent](http://cs231n.github.io/optimization-1/) (Stanford CS231n)

**Generative Learning**

[Generative Learning Algorithms](http://cs229.stanford.edu/notes/cs229-notes2.pdf) (Stanford CS229)

[A practical explanation of a Naive Bayes classifier](https://monkeylearn.com/blog/practical-explanation-naive-bayes-classifier/) (monkeylearn.com)

**Support Vector Machines**

[An introduction to Support Vector Machines (SVM)](https://monkeylearn.com/blog/introduction-to-support-vector-machines-svm/) (monkeylearn.com)

[Support Vector Machines](http://cs229.stanford.edu/notes/cs229-notes3.pdf) (Stanford CS229)

[Linear classification: Support Vector Machine, Softmax](http://cs231n.github.io/linear-classify/) (Stanford 231n)

**Backpropagation**

[Yes you should understand backprop](https://medium.com/@karpathy/yes-you-should-understand-backprop-e2f06eab496b) (medium.com/@karpathy)

[Can you give a visual explanation for the back propagation algorithm for neural networks?](https://github.com/rasbt/python-machine-learning-book/blob/master/faq/visual-backpropagation.md) (github.com/rasbt)

[How the backpropagation algorithm works](http://neuralnetworksanddeeplearning.com/chap2.html) (neuralnetworksanddeeplearning.com)

[Backpropagation Through Time and Vanishing Gradients](http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients/) (wildml.com)

[A Gentle Introduction to Backpropagation Through Time](http://machinelearningmastery.com/gentle-introduction-backpropagation-time/) (machinelearningmastery.com)

[Backpropagation, Intuitions](http://cs231n.github.io/optimization-2/) (Stanford CS231n)

**Deep Learning**

[Deep Learning in a Nutshell](http://nikhilbuduma.com/2014/12/29/deep-learning-in-a-nutshell/) (nikhilbuduma.com)

[A Tutorial on Deep Learning](http://ai.stanford.edu/%7Equocle/tutorial1.pdf) (Quoc V. Le)

[What is Deep Learning?](http://machinelearningmastery.com/what-is-deep-learning/) (machinelearningmastery.com)

[What’s the Difference Between Artificial Intelligence, Machine Learning, and Deep Learning?](https://blogs.nvidia.com/blog/2016/07/29/whats-difference-artificial-intelligence-machine-learning-deep-learning-ai/) (nvidia.com)

**Optimization and Dimensionality Reduction**

[Seven Techniques for Data Dimensionality Reduction](https://www.knime.org/blog/seven-techniques-for-data-dimensionality-reduction) (knime.org)

[Principal components analysis](http://cs229.stanford.edu/notes/cs229-notes10.pdf) (Stanford CS229)

[Dropout: A simple way to improve neural networks](http://videolectures.net/site/normal_dl/tag=741100/nips2012_hinton_networks_01.pdf) (Hinton @ NIPS 2012)

[How to train your Deep Neural Network](http://rishy.github.io/ml/2017/01/05/how-to-train-your-dnn/) (rishy.github.io)

**Long Short Term Memory (LSTM)**

[A Gentle Introduction to Long Short-Term Memory Networks by the Experts](http://machinelearningmastery.com/gentle-introduction-long-short-term-memory-networks-experts/) (machinelearningmastery.com)

[Understanding LSTM Networks](http://colah.github.io/posts/2015-08-Understanding-LSTMs/) (colah.github.io)

[Exploring LSTMs](http://blog.echen.me/2017/05/30/exploring-lstms/) (echen.me)

[Anyone Can Learn To Code an LSTM-RNN in Python](http://iamtrask.github.io/2015/11/15/anyone-can-code-lstm/) (iamtrask.github.io)

**Convolutional Neural Networks (CNNs)**

[Introducing convolutional networks](http://neuralnetworksanddeeplearning.com/chap6.html#introducing_convolutional_networks) (neuralnetworksanddeeplearning.com)

[Deep Learning and Convolutional Neural Networks](https://medium.com/@ageitgey/machine-learning-is-fun-part-3-deep-learning-and-convolutional-neural-networks-f40359318721) (medium.com/@ageitgey)

[Conv Nets: A Modular Perspective](http://colah.github.io/posts/2014-07-Conv-Nets-Modular/) (colah.github.io)

[Understanding Convolutions](http://colah.github.io/posts/2014-07-Understanding-Convolutions/) (colah.github.io)

**Recurrent Neural Nets (RNNs)**

[Recurrent Neural Networks Tutorial](http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/) (wildml.com)

[Attention and Augmented Recurrent Neural Networks](http://distill.pub/2016/augmented-rnns/) (distill.pub)

[The Unreasonable Effectiveness of Recurrent Neural Networks](http://karpathy.github.io/2015/05/21/rnn-effectiveness/) (karpathy.github.io)

[A Deep Dive into Recurrent Neural Nets](http://nikhilbuduma.com/2015/01/11/a-deep-dive-into-recurrent-neural-networks/) (nikhilbuduma.com)

**Reinforcement Learning**

[Simple Beginner’s guide to Reinforcement Learning & its implementation](https://www.analyticsvidhya.com/blog/2017/01/introduction-to-reinforcement-learning-implementation/) (analyticsvidhya.com)

[A Tutorial for Reinforcement Learning](https://web.mst.edu/%7Egosavia/tutorial.pdf) (mst.edu)

[Learning Reinforcement Learning](http://www.wildml.com/2016/10/learning-reinforcement-learning/) (wildml.com)

[Deep Reinforcement Learning: Pong from Pixels](http://karpathy.github.io/2016/05/31/rl/) (karpathy.github.io)

**Generative Adversarial Networks (GANs)**

[What’s a Generative Adversarial Network?](https://blogs.nvidia.com/blog/2017/05/17/generative-adversarial-network/) (nvidia.com)

[Abusing Generative Adversarial Networks to Make 8-bit Pixel Art](https://medium.com/@ageitgey/abusing-generative-adversarial-networks-to-make-8-bit-pixel-art-e45d9b96cee7) (medium.com/@ageitgey)

[An introduction to Generative Adversarial Networks (with code in TensorFlow)](http://blog.aylien.com/introduction-generative-adversarial-networks-code-tensorflow/) (aylien.com)

[Generative Adversarial Networks for Beginners](https://www.oreilly.com/learning/generative-adversarial-networks-for-beginners) (oreilly.com)

**Multi-task Learning**

[An Overview of Multi-Task Learning in Deep Neural Networks](http://sebastianruder.com/multi-task/index.html) (sebastianruder.com)

**NLP**

[A Primer on Neural Network Models for Natural Language Processing](http://u.cs.biu.ac.il/%7Eyogo/nnlp.pdf) (Yoav Goldberg)

[The Definitive Guide to Natural Language Processing](https://monkeylearn.com/blog/the-definitive-guide-to-natural-language-processing/) (monkeylearn.com)

[Introduction to Natural Language Processing](https://blog.algorithmia.com/introduction-natural-language-processing-nlp/) (algorithmia.com)

[Natural Language Processing Tutorial](http://www.vikparuchuri.com/blog/natural-language-processing-tutorial/) (vikparuchuri.com)

[Natural Language Processing (almost) from Scratch](https://arxiv.org/pdf/1103.0398.pdf) (arxiv.org)

**Deep Learning and NLP**

[Deep Learning applied to NLP](https://arxiv.org/pdf/1703.03091.pdf) (arxiv.org)

[Deep Learning for NLP (without Magic)](https://nlp.stanford.edu/courses/NAACL2013/NAACL2013-Socher-Manning-DeepLearning.pdf) (Richard Socher)

[Understanding Convolutional Neural Networks for NLP](http://www.wildml.com/2015/11/understanding-convolutional-neural-networks-for-nlp/) (wildml.com)

[Deep Learning, NLP, and Representations](http://colah.github.io/posts/2014-07-NLP-RNNs-Representations/) (colah.github.io)

[Embed, encode, attend, predict: The new deep learning formula for state-of-the-art NLP models](https://explosion.ai/blog/deep-learning-formula-nlp) (explosion.ai)

[Understanding Natural Language with Deep Neural Networks Using Torch](https://devblogs.nvidia.com/parallelforall/understanding-natural-language-deep-neural-networks-using-torch/) (nvidia.com)

[Deep Learning for NLP with Pytorch](http://pytorch.org/tutorials/beginner/deep_learning_nlp_tutorial.html) (pytorich.org)

**Word Vectors**

[Bag of Words Meets Bags of Popcorn](https://www.kaggle.com/c/word2vec-nlp-tutorial) (kaggle.com)

On word embeddings [Part I](http://sebastianruder.com/word-embeddings-1/index.html), [Part II](http://sebastianruder.com/word-embeddings-softmax/index.html), [Part III](http://sebastianruder.com/secret-word2vec/index.html) (sebastianruder.com)

[The amazing power of word vectors](https://blog.acolyer.org/2016/04/21/the-amazing-power-of-word-vectors/) (acolyer.org)

[word2vec Parameter Learning Explained](https://arxiv.org/pdf/1411.2738.pdf) (arxiv.org)

Word2Vec Tutorial — [The Skip-Gram Model](http://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/), [Negative Sampling](http://mccormickml.com/2017/01/11/word2vec-tutorial-part-2-negative-sampling/) (mccormickml.com)

**Encoder-Decoder**

[Attention and Memory in Deep Learning and NLP](http://www.wildml.com/2016/01/attention-and-memory-in-deep-learning-and-nlp/) (wildml.com)

[Sequence to Sequence Models](https://www.tensorflow.org/tutorials/seq2seq) (tensorflow.org)

[Sequence to Sequence Learning with Neural Networks](https://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf) (NIPS 2014)

[Machine Learning is Fun Part 5: Language Translation with Deep Learning and the Magic of Sequences](https://medium.com/@ageitgey/machine-learning-is-fun-part-5-language-translation-with-deep-learning-and-the-magic-of-sequences-2ace0acca0aa) (medium.com/@ageitgey)

[How to use an Encoder-Decoder LSTM to Echo Sequences of Random Integers](http://machinelearningmastery.com/how-to-use-an-encoder-decoder-lstm-to-echo-sequences-of-random-integers/) (machinelearningmastery.com)

[tf-seq2seq](https://google.github.io/seq2seq/) (google.github.io)

**Python**

[7 Steps to Mastering Machine Learning With Python](http://www.kdnuggets.com/2015/11/seven-steps-machine-learning-python.html) (kdnuggets.com)

[An example machine learning notebook](http://nbviewer.jupyter.org/github/rhiever/Data-Analysis-and-Machine-Learning-Projects/blob/master/example-data-science-notebook/Example%20Machine%20Learning%20Notebook.ipynb) (nbviewer.jupyter.org)

**Examples**

[How To Implement The Perceptron Algorithm From Scratch In Python](http://machinelearningmastery.com/implement-perceptron-algorithm-scratch-python/) (machinelearningmastery.com)

[Implementing a Neural Network from Scratch in Python](http://www.wildml.com/2015/09/implementing-a-neural-network-from-scratch/) (wildml.com)

[A Neural Network in 11 lines of Python](http://iamtrask.github.io/2015/07/12/basic-python-network/) (iamtrask.github.io)

[Implementing Your Own k-Nearest Neighbour Algorithm Using Python](http://www.kdnuggets.com/2016/01/implementing-your-own-knn-using-python.html) (kdnuggets.com)

[Demonstration of Memory with a Long Short-Term Memory Network in Python](http://machinelearningmastery.com/memory-in-a-long-short-term-memory-network/) (machinelearningmastery.com)

[How to Learn to Echo Random Integers with Long Short-Term Memory Recurrent Neural Networks](http://machinelearningmastery.com/learn-echo-random-integers-long-short-term-memory-recurrent-neural-networks/) (machinelearningmastery.com)

[How to Learn to Add Numbers with seq2seq Recurrent Neural Networks](http://machinelearningmastery.com/learn-add-numbers-seq2seq-recurrent-neural-networks/) (machinelearningmastery.com)

**Scipy and numpy**

[Scipy Lecture Notes](http://www.scipy-lectures.org/) (scipy-lectures.org)

[Python Numpy Tutorial](http://cs231n.github.io/python-numpy-tutorial/) (Stanford CS231n)

[An introduction to Numpy and Scipy](https://engineering.ucsb.edu/%7Eshell/che210d/numpy.pdf) (UCSB CHE210D)

[A Crash Course in Python for Scientists](http://nbviewer.jupyter.org/gist/rpmuller/5920182#ii.-numpy-and-scipy) (nbviewer.jupyter.org)

**scikit-learn**

[PyCon scikit-learn Tutorial Index](http://nbviewer.jupyter.org/github/jakevdp/sklearn_pycon2015/blob/master/notebooks/Index.ipynb) (nbviewer.jupyter.org)

[scikit-learn Classification Algorithms](https://github.com/mmmayo13/scikit-learn-classifiers/blob/master/sklearn-classifiers-tutorial.ipynb) (github.com/mmmayo13)

[scikit-learn Tutorials](http://scikit-learn.org/stable/tutorial/index.html) (scikit-learn.org)

[Abridged scikit-learn Tutorials](https://github.com/mmmayo13/scikit-learn-beginners-tutorials) (github.com/mmmayo13)

**Tensorflow**

[Tensorflow Tutorials](https://www.tensorflow.org/tutorials/) (tensorflow.org)

[Introduction to TensorFlow — CPU vs GPU](https://medium.com/@erikhallstrm/hello-world-tensorflow-649b15aed18c) (medium.com/@erikhallstrm)

[TensorFlow: A primer](https://blog.metaflow.fr/tensorflow-a-primer-4b3fa0978be3) (metaflow.fr)

[RNNs in Tensorflow](http://www.wildml.com/2016/08/rnns-in-tensorflow-a-practical-guide-and-undocumented-features/) (wildml.com)

[Implementing a CNN for Text Classification in TensorFlow](http://www.wildml.com/2015/12/implementing-a-cnn-for-text-classification-in-tensorflow/) (wildml.com)

[How to Run Text Summarization with TensorFlow](http://pavel.surmenok.com/2016/10/15/how-to-run-text-summarization-with-tensorflow/) (surmenok.com)

**PyTorch**

[PyTorch Tutorials](http://pytorch.org/tutorials/) (pytorch.org)

[A Gentle Intro to PyTorch](http://blog.gaurav.im/2017/04/24/a-gentle-intro-to-pytorch/) (gaurav.im)

[Tutorial: Deep Learning in PyTorch](https://iamtrask.github.io/2017/01/15/pytorch-tutorial/) (iamtrask.github.io)

[PyTorch Examples](https://github.com/jcjohnson/pytorch-examples) (github.com/jcjohnson)

[PyTorch Tutorial](https://github.com/MorvanZhou/PyTorch-Tutorial) (github.com/MorvanZhou)

[PyTorch Tutorial for Deep Learning Researchers](https://github.com/yunjey/pytorch-tutorial) (github.com/yunjey)

**Math**

[Math for Machine Learning](https://people.ucsc.edu/%7Epraman1/static/pub/math-for-ml.pdf) (ucsc.edu)

[Math for Machine Learning](http://www.umiacs.umd.edu/%7Ehal/courses/2013S_ML/math4ml.pdf) (UMIACS CMSC422)

**Linear algebra**

[An Intuitive Guide to Linear Algebra](https://betterexplained.com/articles/linear-algebra-guide/) (betterexplained.com)

[A Programmer’s Intuition for Matrix Multiplication](https://betterexplained.com/articles/matrix-multiplication/) (betterexplained.com)

[Understanding the Cross Product](https://betterexplained.com/articles/cross-product/) (betterexplained.com)

[Understanding the Dot Product](https://betterexplained.com/articles/vector-calculus-understanding-the-dot-product/) (betterexplained.com)

[Linear Algebra for Machine Learning](http://www.cedar.buffalo.edu/%7Esrihari/CSE574/Chap1/LinearAlgebra.pdf) (U. of Buffalo CSE574)

[Linear algebra cheat sheet for deep learning](https://medium.com/towards-data-science/linear-algebra-cheat-sheet-for-deep-learning-cd67aba4526c) (medium.com)

[Linear Algebra Review and Reference](http://cs229.stanford.edu/section/cs229-linalg.pdf) (Stanford CS229)

**Probability**

[Understanding Bayes Theorem With Ratios](https://betterexplained.com/articles/understanding-bayes-theorem-with-ratios/) (betterexplained.com)

[Review of Probability Theory](http://cs229.stanford.edu/section/cs229-prob.pdf) (Stanford CS229)

[Probability Theory Review for Machine Learning](https://see.stanford.edu/materials/aimlcs229/cs229-prob.pdf) (Stanford CS229)

[Probability Theory](http://www.cedar.buffalo.edu/%7Esrihari/CSE574/Chap1/Probability-Theory.pdf) (U. of Buffalo CSE574)

[Probability Theory for Machine Learning](http://www.cs.toronto.edu/%7Eurtasun/courses/CSC411_Fall16/tutorial1.pdf) (U. of Toronto CSC411)

**Calculus**

[How To Understand Derivatives: The Quotient Rule, Exponents, and Logarithms](https://betterexplained.com/articles/how-to-understand-derivatives-the-quotient-rule-exponents-and-logarithms/) (betterexplained.com)

[How To Understand Derivatives: The Product, Power & Chain Rules](https://betterexplained.com/articles/derivatives-product-power-chain/) (betterexplained.com)

[Vector Calculus: Understanding the Gradient](https://betterexplained.com/articles/vector-calculus-understanding-the-gradient/) (betterexplained.com)

[Differential Calculus](http://web.stanford.edu/class/cs224n/lecture_notes/cs224n-2017-review-differential-calculus.pdf) (Stanford CS224n)

[Calculus Overview](http://ml-cheatsheet.readthedocs.io/en/latest/calculus.html) (readthedocs.io)