

Links to Resources

[Google Colab](#)

[Github](#)

[Machine Learning 2019](#)

[Section 2: Setting up Environment](#)

[Section 3: Regression](#)

[Section 4: Classification](#)

[Section 5: Support Vector Machine](#)

[Section 6: Tree](#)

[Section 7: Ensemble Models](#)

[Section 8: k-Nearest Neighbor](#)

[Section 9: Unsupervised Machine Learning - Dimensionality Reduction](#)

[Section 10: Unsupervised Machine Learning - Clustering](#)

[Deep Learning 2019](#)

[ANN](#)

[CNN](#)

[Appendix A1: Foundations of Deep Learning](#)

[Appendix A2: Foundations of CNN](#)

Copyright 2019 Anthony Ng.

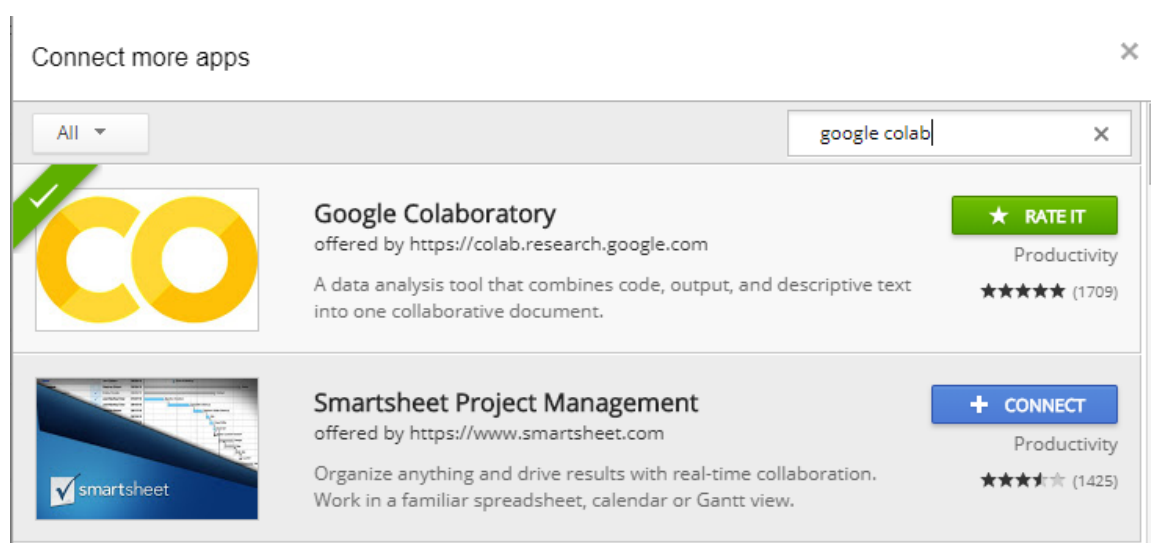
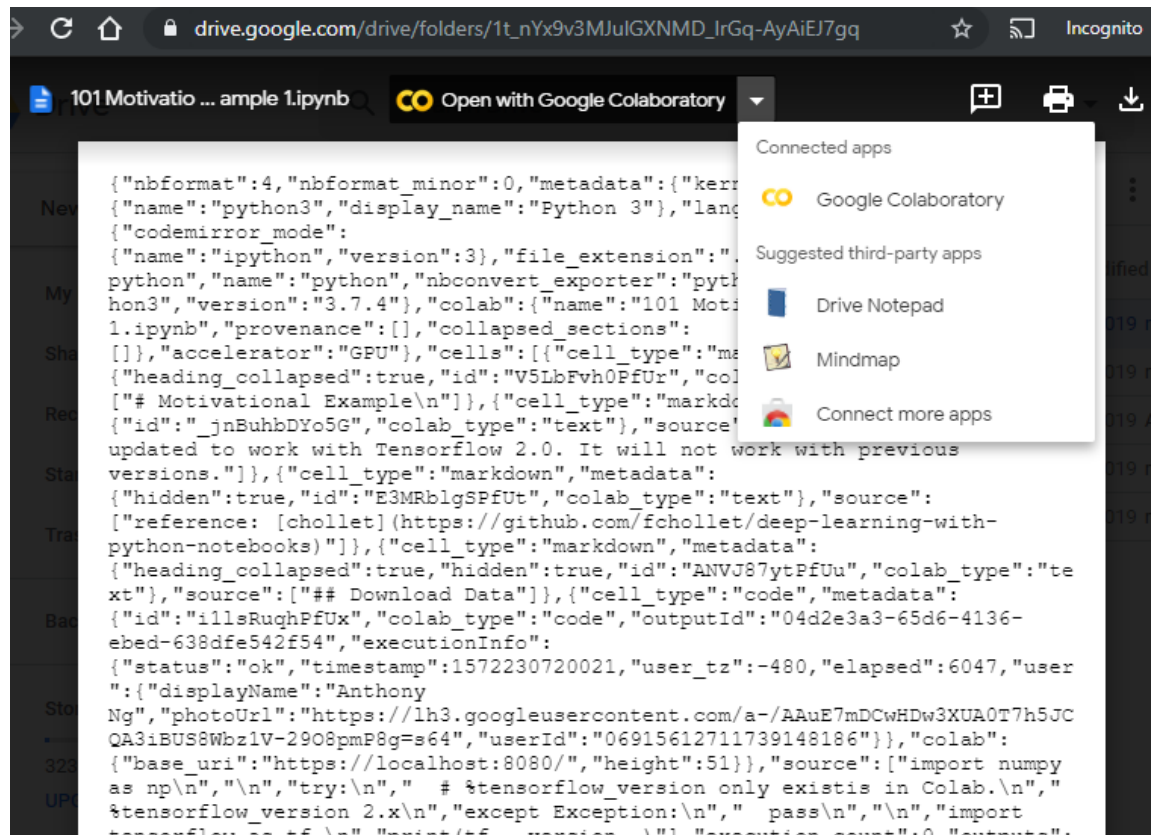
Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Google Colab

A note on opening Google Colab files, you may need to install the app.



Github

[Link to GitHub](#) for all the notebooks for Machine Learning section. These notebooks can be used for running on your local machine.

Machine Learning 2019

These are the links to the notebooks for Machine Learning section. These notebooks can be run via google colab. This is especially useful for training large and complex algorithms such as tree based models.

https://drive.google.com/drive/folders/1_iM2khZYFwQfxuHWC1kAdW4SrNgbKNow?usp=sharing

Section 2: Setting up Environment

[Installing Anaconda to your local Machine](#)

[Jupyter Notebook, lab and environment management](#)

[Hello World](#)

[Iris Project](#)

Section 3: Regression

[Boston Housing Price Prediction](#)

[Multiple Regression](#)

[Regularized Regression](#)

[Polynomial Regression](#)

[Nonlinear Relationships](#)

[Data Pre-Processing](#)

[Variance-Bias Tradeoff](#)

[Cross Validation](#)

Section 4: Classification

[Logistic Regression](#)

[Classification with MNIST](#)

Section 5: Support Vector Machine

[Support Vector Machine](#)

Section 6: Tree

[Decision Tree](#)

[Project HR](#)

Section 7: Ensemble Models

[Introduction](#)

[Bagging Machine Learning Algorithm](#)

[Random Forest and Extra Trees](#)

[AdaBoost and Gradient Boosting Machine](#)

[XGBoost Installation Instruction](#)

[XGBoost](#)

[Project HR](#)

[Ensemble of Ensembles 1](#)

[Ensemble of Ensembles 2](#)

Section 8: k-Nearest Neighbor

[Introduction](#)

[Project Cancer Detection](#)

Section 9: Unsupervised Machine Learning - Dimensionality Reduction

[Dimensionality Reduction](#)

[PCA - Linear](#)

[Project Wine](#)

[Kernel PCA](#)

[Linear Discriminant Analysis \(LDA\)](#)

[Project Abalone](#)

Section 10: Unsupervised Machine Learning - Clustering

[Clustering](#)

Deep Learning 2019

https://drive.google.com/open?id=1pxZW8fqSqOp5_yqUhyAL5Mt7dEV_M_GD

ANN

[Simple Function](#)

[Motivational Example 1 with MNIST dataset](#)

[Binary Classification 1 with MNIST dataset](#)

[Binary Classification 2 with IMDB dataset](#)

[Regression with Boston Housing dataset](#)

[ANN Resources - DOCX and PDF](#)

CNN

[CNN Motivational Example with MNIST dataset](#)

[CNN with dogs and cats \(subset\) dataset](#)

[Transfer Learning](#)

[CNN Resources - PDF and EXCEL](#)

Appendix A1: Foundations of Deep Learning

https://drive.google.com/open?id=1t_nYx9v3MJlGXNMD_IrGq-AyAiEJ7gg

[Motivational Example](#)

[Binary Classification Example 1](#)

[Binary Classification Example 2](#)

[Multi-Class Example](#)

[Regression Example](#)

Appendix A2: Foundations of CNN

<https://drive.google.com/open?id=1mD8PrNhJVihItJN-hOIJk4HqzZBxL3H>

[Cats and Dogs](#)

[Cats and Dogs - Loading Previously Trained Model](#)

[Cats and Dogs - Data Augmentation](#)

[Data Augmentation](#)

[Transfer Learning with VGG16](#)

[Features Extraction with VGG16](#)

[Transfer Learning with ResNet50](#)