Course Admin

EE-UY 4563/EL-GY 6123: INTRODUCTION TO MACHINE LEARNING

PROF. SUNDEEP RANGAN

People

- Professor: Sundeep Rangan, srangan@nyu.edu
  - 2 MetroTech Center 9.104
  - Office Hours: Thursdays, 2-4pm

- Head TAs:
  - Juntao Chen jc6412@nyu.edu
  - Amirhossein Khalilian-Gourtani akg404@nyu.edu
  - Office Hours: TBD
  - Ask for all questions regarding homeworks and labs

- There will be several other graders as well

Course Learning Objectives

- Formulate a task as a machine learning problem
  - Identify learning objectives, source of data, models, …
- Load, pre-process and extract features from common data sources
  - images, text, audio, …
- Mathematically describe simple models of the data
- Fit the models to data and use models for prediction and estimation
  - Use common tools
- Evaluate goodness of fit and refine models
- Evaluate the performance of methods using statistical techniques

Grad vs Undergrad

- Class is simultaneously offered at the graduate and undergraduate level
- Undergrad EE-UY/CSE-UY 4563: Intro to Machine Learning
  - Covers fundamental algorithms and some analysis
  - In depth coverage of software tools including python, Google Cloud, Tensorflow
  - Python-based lab exercises + mandatory project
- Grad EL 6123: Intro to Machine Learning
  - More algorithms and more mathematical analysis. Faster paced.
  - Software tools must be learned at home. Less coverage in class
  - Python-based lab exercises + optional project
- Lecture notes are mostly common with supplementary material for grad students indicated
- Many labs are common

Texts and Other Resources

- Undergrad: James, Witten, Hastie and Tibshirani, “An Introduction to Statistical Learning”,
  - http://www-bcf.usc.edu/~gareth/ISL/ISLR%20First%20Printing.pdf
  - Very clear explanation of concepts.
  - But examples are in R. And there is no review of probability
- Grad: Hastie, Tibshirani, Friedman, “Elements of Statistical Learning”
  - https://web.stanford.edu/~hastie/Papers/ESLII.pdf
  - More advanced text with more analysis
- Raschka, “Python Machine Learning”, 2015.
  - http://file.allitebooks.com/20151017/Python%20Machine%20Learning.pdf
  - Excellent examples of using Python
- Bishop, “Pattern Recognition and Machine Learning” (more advanced)
- Coursera course: Generally do not cover probability
- Undergrad probability

More Resources

- Entertaining and very good deep learning lectures by Siraj Raval
  - https://www.youtube.com/channel/UCWN3xxRkmTPmbKwht9FuE5A
- Universite de Paris labs:
  - https://github.com/m2dsupsdlclass/lectures-labs
  - Focus on deep learning
  - Similar format to this class
- Andrew Ng's machine learning class:
  - https://www.coursera.org/learn/machine-learning
  - A little less mathematical than this class
- Many, many others online…

Pre-Requisites

- Undergrad probability required for both UG and Grad version:
  - Basics of random variables, densities, Gaussian distributions, correlation, expectation, conditional densities, Bayes' theorem
  - Will provide a short review
  - NYU classes: Data analysis or Intro Probability are sufficient
- Undergraduate calculus and linear algebra
  - Vectors, matrices, partial derivatives, gradients.
  - Again, we will provide a brief review
- No machine learning experience is necessary
  - If you have ML experience, do NOT take this class.
  - Take Graduate probability (Fall) then Advanced machine learning (Spring)

Pre-Requisites Programming

- Python
  - All labs are in python, similar to object-oriented MATLAB, but many more libraries.
  - And free!
- What you need to know
  - You do not need to know python before class. But, we will go over it quickly.
  - You should have experience in some programming language (eg. MATLAB).
  - You should know or being willing to learn object oriented programming
- Resources:
  - Installing python and ipython notebook (make sure you install Version 3.6) http://jupyter-notebook-beginner-guide.readthedocs.io/en/latest/index.html
  - Python tutorial: https://docs.python.org/3/tutorial/
  - Numpy: http://cs231n.github.io/python-numpy-tutorial/

Grading: Undergraduate

- Midterm 1: 25%, Midterm 2: 25%, Labs, HW: 25%, Final project: 25%
- Labs: Simple python exercises
  - Given as jupyter notebook that you complete.
- Midterms
  - Each over approx. 3-4 weeks of material
  - Closed book with cheat sheet.
  - Follows homework and quiz problems + some very basic python questions
- Final project:
  - Use machine learning in some interesting way.
  - Must use data and python analysis.
  - Provide final report.

Grading: Graduate

- Midterm 35%, Final 35%, Labs / HW 30%
  - Optional project: Up to 20%
- Labs: Simple python exercises
  - Given as jupyter notebook that you complete.
- Midterms & final
  - Each over approx. 6-7 weeks
  - Open book but no electronic aids.
  - Follows homework and quiz problems + some very basic python questions
- Optional final project:
  - Use machine learning in some interesting way.
  - Must use data and python analysis.
  - Provide final report.

Machine Learning Project

- Perform an interesting machine learning task of your choice
- Many possible areas:
  - Machine vision, brain-computer interfaces, natural language processing, sentiment analysis, …
  - Anything that interests you
- Groups of 2 preferred
  - In NYU Classes, join a group “project1, project2, …”
  - Submit all material as that group
- Use real data
  - UCI ML repository
  - Google BigQuery data
- Write code
- Place all material in a github repo (including documentation) and submit only github repo

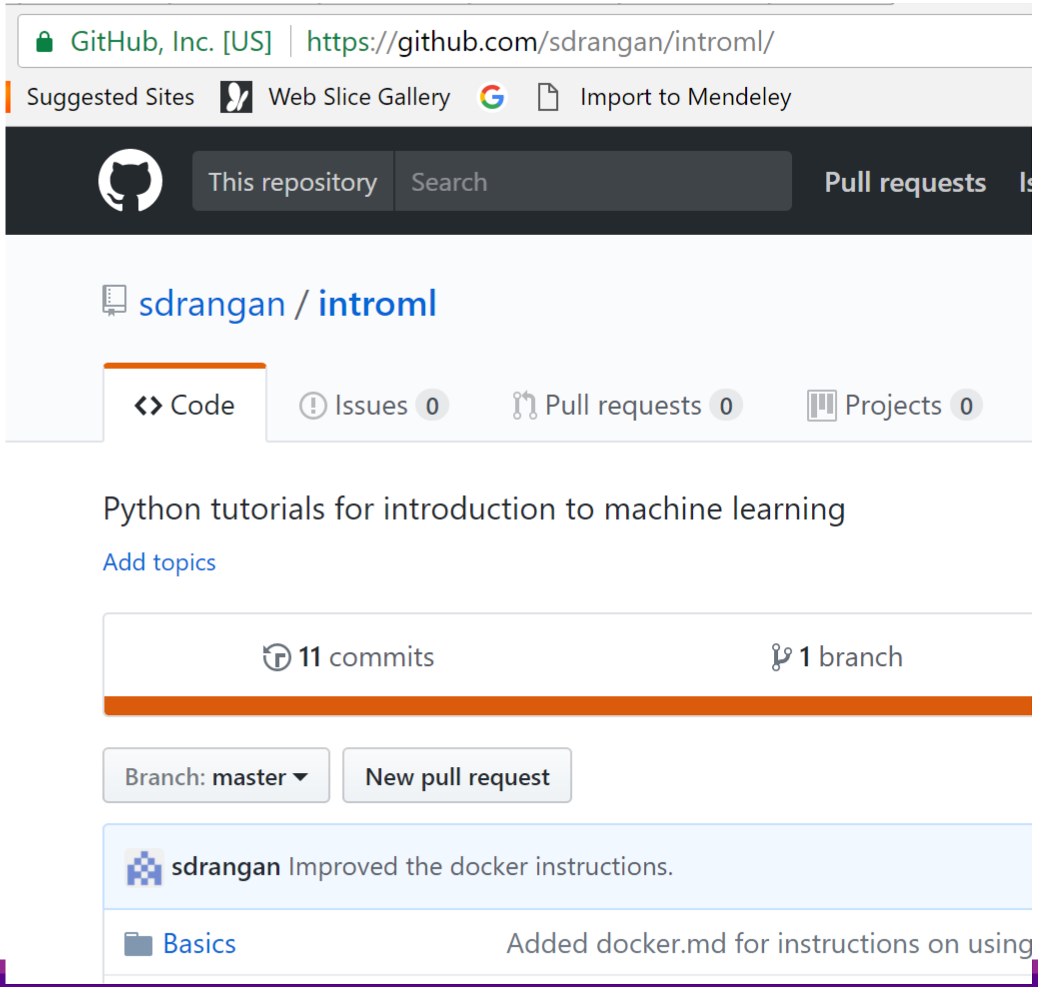
Project Grading

- Formulation
  - How well did you formulate the problem? Was it clear? Was that tied to the right objective?
- Approach
  - Does your approach properly solve your problem? Was that made clear?
- Evaluation and Interpretation
  - Did you comprehensively test the results? How well did you select / create the data?
  - Did you test against alternative approaches?
- Presentation
  - Were the ideas clear? Were all the details conveyed. Did you highlight the main points?
  - You can select a number of formats. Whatever makes sense. A github page
- Bonus
  - Given for particularly hard / novel research

Github

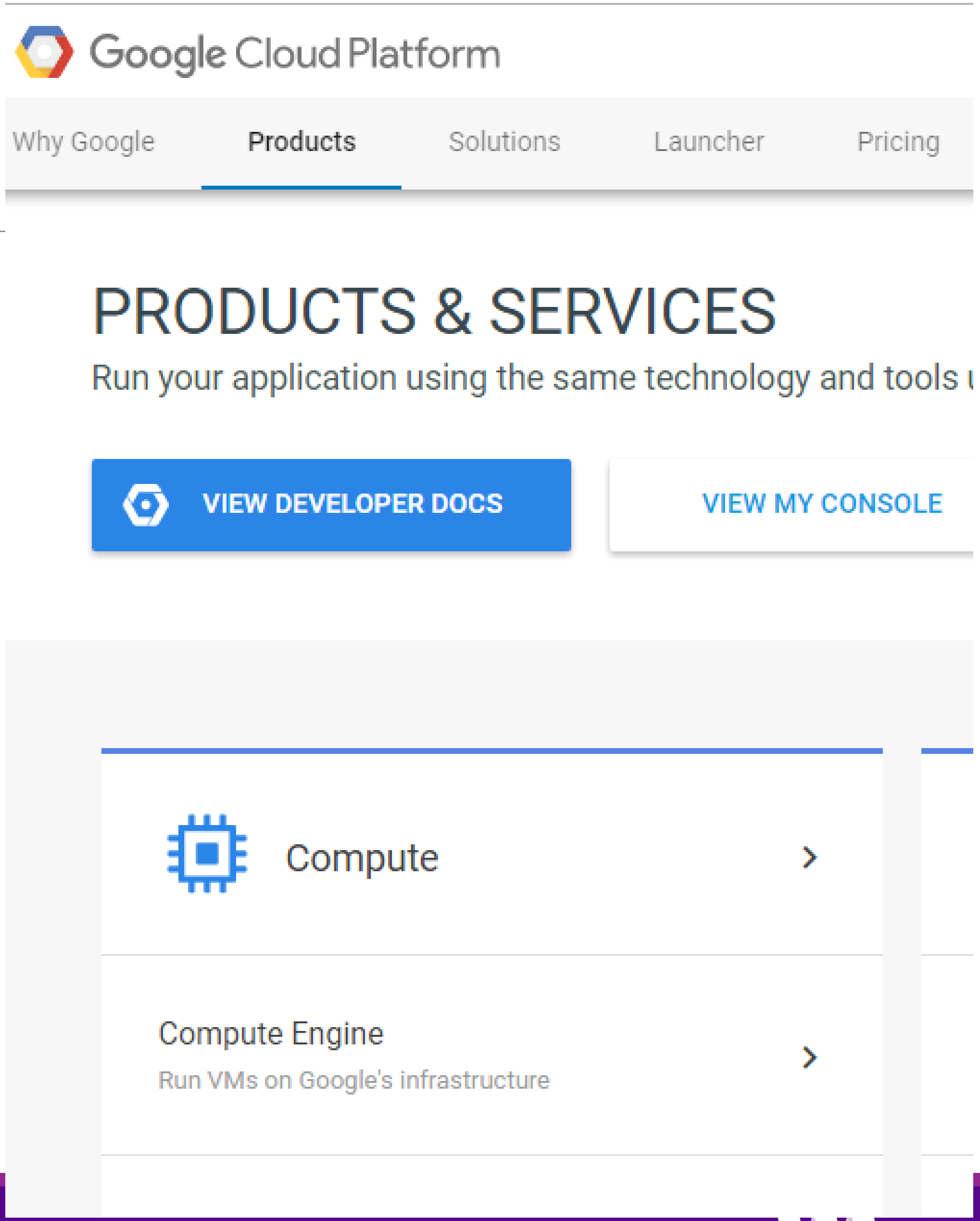
- Labs and demo posted on github
- https://github.com/sdrangan/introml/
- Also includes instruction for installing software

- Several tutorials of github on the web.
- Available on Windows, Mac and Unix.
- But, you can just clone the repo



Google Cloud Platform

- All labs in this class can be run on either:
  - Your own computer: Windows, MAC
  - Google Cloud Platform (GCP)
- GCP pros and cons:
  - Access to powerful machines / large storage for projects. Includes GPUs
  - Access to many services such as BigQuery
  - Can scale your computational resources
  - But, somewhat harder to sync editors / debuggers
- Getting started: https://cloud.google.com/
- Instructions on https://github.com/sdrangan/introml/tree/master/GCP



Other Software

- On your machine (local or GCP), you will need to install several pieces of software:
- Python with various packages
  - Make sure you get 3.6
  - Anaconda
  - Jupyter notebook
  - See notes in NYU Classes
- Tensorflow and Keras (needed only later in the class)
- Git hub
  - Guides: https://guides.github.com/
  - Available on Windows, Mac or Linux (including GCP instances)
  - All demos will be available on: https://github.com/sdrangan/introml.git