CS205L

Continuous Mathematical Methods
(emphasizing Machine Learning)

http://web.stanford.edu/class/cs205l/index.html

CS 205 Course History

- Mathematical Modeling of Continuous Systems (Tomasi)
    - ‘94-’95, ‘95-’96, ‘96-’97, ‘97-’98, ‘98-’99, ‘99-2000, 2000-’01 (7 years)
    - 2001-’02 not taught…
- New Emphasis on Robotics, Vision & Graphics (Fedkiw)
    - CS205: 2002-’03, ‘03-04, ‘04-’05, ‘05-’06, ‘06-’07 (5 years)
    - CS205A: 2007-’08, ‘08-’09, ‘09-’10, ‘10-’11, ‘11-’12, ‘12-’13 (6 years)
    - Justin Solomon (@MIT now) 2013-’14, ‘14-’15 (2 years)
    - Doug James 2015-’16, ‘16-’17, ‘17-’18 (3 years)
- New Emphasis on Machine Learning (Fedkiw)
    - CS205L: 2018-’19, ‘19-’20, ‘20-’21, ‘21-’22 (4th year)

GPU History

- 1970s Specialized hardware for graphics became popular for stand alone Arcade games
- 1993 Nvida was founded (Jensen Huang et al.)
- 1994 The term "GPU" was coined by SONY referring to a chip in the 1994 Playstation 1
- 1995 Toy Story (Pixar) – revolutionized 3D graphics for animated feature films
- 1996 Super Mario 64 (Nintendo 64) – revolutionized 3D graphics for games
- 1999 Star Wars: Episode 1, The Phantom Menace – revolutionized 3D movie VFX
- 2002 restructured CS205 to focus on graphics applications

- Now (like graphics in the past) machine learning is the popular computationally intensive area of computer science that is getting specialized hardware (e.g. Nvidia's ML/DL chips)
- So, restructured CS205 (once again) to CS205L

CS205L: Continuous Mathematical Methods with an Emphasis on Machine Learning

A survey of numerical approaches to the continuous mathematics used throughout computer science with an emphasis on machine and deep learning. Although motivated from the standpoint of machine learning, the course will focus on the underlying mathematical methods including computational linear algebra and optimization, as well as special topics such as automatic differentiation via backward propagation, momentum methods from ordinary differential equations, CNNs, RNNs, etc. Written homework assignments and (straightforward) quizzes focus on various concepts; additionally, students can opt in to a series of programming assignments geared towards neural network creation, training, and inference. (Replaces CS205A, and satisfies all similar requirements.) Prerequisites: Math 51; Math 104 or 113 or equivalent or comfort with the associated material. (Fedkiw)

CS205L

- Replaces CS205A, and satisfies all similar requirements.
- Prerequisites: Math 51; Math 104 or 113 or equivalent or <u>comfort</u> with the associated material.

Lectures

- The slides are broken into units, and each unit is broken into smaller chunks by changing the background color of the slides
- There is separate slide deck on notation (see the web site)
- A lot of the connections between continuous mathematics and machine/deep learning is still cutting edge research, so Ask Questions!

- Lectures are recorded, and attendance is optional
- Problem sets are based on the lectures, and related supplemental readings
- Quizzes are based on the lectures (and the questions are handed out in advance)

- Friday is an optional CA section on special topics/review (Covid may affect this)
  - Starts the second week of classes
  - Recorded, also optional, also ask questions…

Written Problem Sets (50% of grade)

- Written; No programming
- Assigned weekly on Thurs, covers Tues/Thurs lectures of **that** week (due 1 week later)
- Up to 4 people in discussion group (list names of group on HW)
- However, each person turns in their own unique-as-possible write-ups
  - if your write-up is not original enough, you may be asked to do extra problems for that assignment
- Each problem is assigned to a specific CA, and CAs will only address their problem(s)
  - However, CAs will address all *general* course material (and quiz questions)
- Each problem is graded on a coarse 0, ¼, ½, ¾, 1 point scale by the CA who created it
- Problems vary in difficulty: 1, 2, or 3 stars
- Choose 6 stars worth of problems (the maximum score is 6 points)

Weekly Quizzes (50% of grade)

- Each Thurs, a number of potential quiz questions will be released (along with the problem set for that week)
- The questions cover both the lectures (Tues and Thurs) of that week
- You may discuss questions with your HW discussion group, the CAs, etc.
- The following Monday afternoon (4 days later), you will take a one-on-one (no partners) oral quiz with one of the CAs (online this year)
- They will choose 1 question from the pre-determined list for you to answer
- If you need special accommodations for the time slot, reach out to the CAs

(Optional) Programming Assignments

- You may work with at most 1 partner, and may collaborate on code
- Weekly assignments, and some in-person grading (with CA: Jane - only)
- Intended to be practical, not difficult (probably *still* too easy this year)
- Goal: Create a network architecture, train a network to match data, and ascertain generalization error during prediction/inference

- If your Programming Assignment score is higher than your Problem Set score, 1/3 of your Problem Set grade is replaced with your Programming Assignment grade
- If your Programming Assignment score is higher than your Weekly Quiz score, 1/3 of your Weekly Quiz grade is replaced with your Programming Assignment grade
- You must be consistently doing the programming assignments, each week, in order for them to be included in your grade

(Tentative) Programming Assignments

- PA1: (Setup) Install Tensorflow.
- PA2: (Dataset Loading and Augmentation) Download a facial landmark dataset. Load the dataset and apply pre-processing, e.g. train/validation/test split and data augmentation.
- PA3: (Custom Layers and Models) Learn how to write custom Tensorflow layers/models and build a model for facial landmark detection. Write a loss function for heatmap regression.
- PA4: (Dataset Pre-processing and Network Training) Pre-process facial landmark detection data and train a heatmap regression network.
- PA5: (Pre-trained Network Training and Inference) Load pre-trained facial alignment network, train subset of weights, and run inference.
- PA6: (Tensorboard) Train on MNIST dataset and log results on Tensorboard.
- PA7: (Regularization) Add L1/L2 regularization and a dropout layer and compare results on MNIST dataset.
- PA8: (Optimizers) Implement and compare three popular optimizers: gradient descent, momentum, and Adam.
- PA9: (Ill-conditioning) Investigate challenges of ill-conditioning and methods to address the limitations of gradient descent.

Roadmap

- Part I – Linear Algebra (units 1-12) $Ac = b$

linearize

line search

- Part II – Optimization (units 13-20)
  - (units 13-16) Optimization -> Nonlinear Equations -> 1D roots/minima ← Theory
  - (units 17-18) Computing/Avoiding Derivatives
  - (unit 19) Hack 1.0: “I give up” $H = I$ and J is mostly 0 (descent methods)
  - (unit 20) Hack 2.0: ”It’s an ODE!?” (adaptive learning rate and momentum)

← Methods

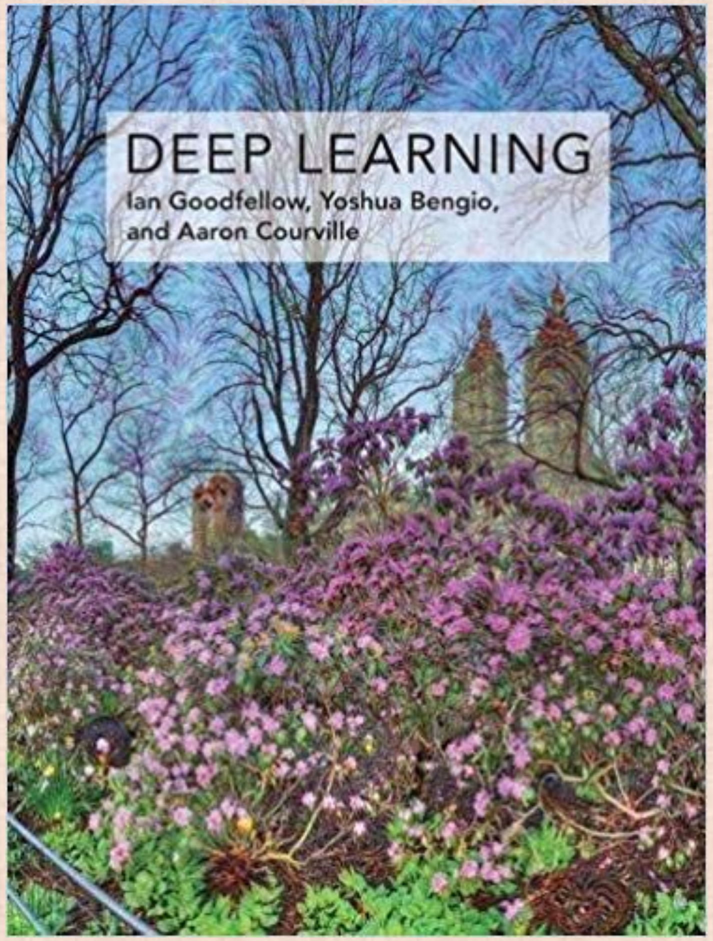
Solving Linear Systems

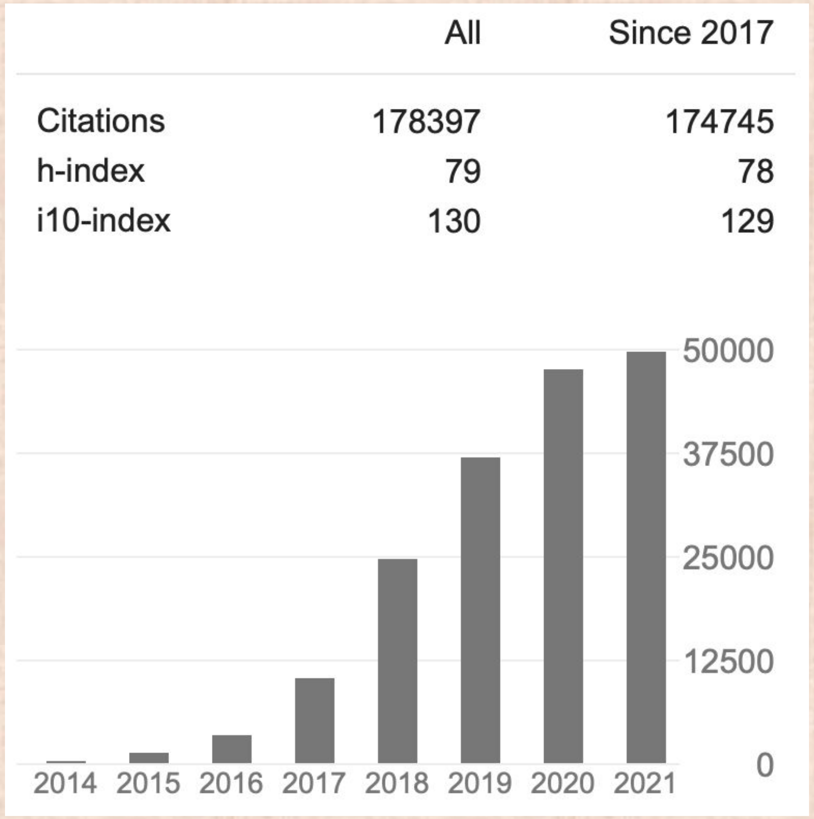
- Theory, all matrices: SVD (units 3, 9, 11)
- Square, full rank, dense:
  - LU factorization with pivoting (unit 2)
  - Symmetric: Cholesky factorization (unit 4), Symmetric approximation (unit 4)
- Square, full rank, sparse (iterative solvers) (unit 5):
  - SPD (sometimes SPSD): Conjugate Gradients
  - Nonsymmetric/Indefinite: GMRES, MINRES, BiCGSTAB (not steepest descent)
- Tall, full rank (least squares to minimize residual) (unit 8):
  - normal equations (units 9, 10), QR, Gram-Schmidt, Householder (unit 10)
- Any size/rank (minimum norm solution) (unit 11):
  - Pseudo-Inverse, PCA approximation, Power Method (unit 11)
  - Levenberg-Marquardt (iteration too), Column Space Geometric Approach (unit 12)

-----Original Message-----

From: Ian Goodfellow

Ron, I don't know if you remember me, but I took your class circa 2009. Writing numerically stable code was a big factor in my success as a DL researcher, especially for GANs.





	All	Since 2017
Citations	178397	174745
h-index	79	78
i10-index	130	129

Questions?