Course Logistics and Overview

Julia Kempe & David S. Rosenberg

CDS, NYU

January 29, 2019

Logistics

Logistics

- Class webpage: https://davidrosenberg.github.io/ml2019
  - Syllabus on the website
- Piazza: https://piazza.com/nyu/spring2019/dsga1003
  - **All class announcements via Piazza**
  - Ask all questions on Piazza
- Class Times
  - Tuesdays "Lecture": 5:20 - 7pm (Meyer 121)
  - Wednesdays "Lab": 6:45 - 7:35pm (Meyer 121)
  - **(Both are required.)**
- Office Hours:
  - Julia - Tuesdays 4-5pm, 60 5th Ave., 620 (Except weeks 5,7,11, possibly one more -> David)
  - Sreyas/Xintian (TAs): Wednesdays 7:45-8:45pm 60 5th Ave C-15
  - Graders: TBD (see course webpage)

Course Staff

- Instructors:
  - Julia Kempe (CDS Director, Professor for Computer Science and Mathematics, NYU Courant Institute)
  - David S. Rosenberg (CDS, Bloomberg) - 3-4 lectures and behind the scenes
- TAs:
  - Sreyras Mohan (CDS, PhD Data Science)
  - Xintian Han (CDS, PhD Data Science)
- Graders:
  - Sanyam Kapoor (Head Grader)
  - Aakash Kaku
  - Mingsi Long
  - Mihir Rana
  - Tingyan Xiang
  - Yi Zhou

Evaluation

- About 7 or 8 homeworks (40%)
- Two tests (60%)
  - Midterm Exam (30%) in Week 7 (March 6th)
  - Final Exam (30%) - Final Exam Period (Thursday May 16th 6-7:50pm - to be confirmed)
- These scores determine "class rank".
- Typical grade distribution: A (40%), A- (20%), B+ (20%), B (10%), B- (5%), <B- (5%)

Optional Homework Problems

- There will be a significant number of **optional homework problems**
- Grade-wise
  - Optional problems **do not** contribute to your homework grade.
  - They are a separate grade category
  - Primarily used to boost a borderline grade at the end of the term
  - **At most, increases final grade by half a letter (e.g. B+ to A-)**
  - In 2018, about 10% of people has letter grade increases from optional credit.
  - (To a lesser extent, Piazza and class participation can also help bump up a borderline grade.)
- It's primarily for highly motivated individuals (who have the time) to
  - Learn more concepts and practice more techniques
- High performance on optional homework is something we can mention in recommendation letters.

Lab Sessions

- Most led by TAs Sreyras and Xintian
- Most will be lecture format, some will be reviews
- Tomorrow: Guest lecture from Brett Bernstein (2017 TA, PhD student)

Homework (40%)

- First assignment out now – due week from Thursday 23:59pm
- Submit with Gradescope (details on website)
- Homeworks should be **submitted as a PDF document**.
- Late homework: Accepted up to 48 hours late with 20% penalty
- Collaboration is fine, but
  - Write up solutions and code on your own
  - List names of who you talked to about each problem
- When graders identify copying, we're obliged to tell the administration, which gets uncomfortable for everybody.

Prerequisites

- DS-GA 1001: Introduction to Data Science
- DS-GA 1002: Statistical and Mathematical Methods
- Math
  - Multivariate Calculus
  - Linear Algebra (see HW 1 for a question)
  - Probability Theory (see HW 1 for a question)
  - Statistics
  - [Preferred] Proof-based linear algebra or real analysis
- Python programming (numpy)

Course Overview and Goals

Syllabus (Tentative)

13 weeks of instruction + 1 week midterm exam

- 4-5 weeks: **Linear** methods for **binary classification** and **regression** (also **kernel methods)**
- 2 Weeks: Conditional **probability models**, **Bayesian** methods
- 1 Week: **Multiclass** and introduction to **structured prediction**
- 3-4 weeks: **Nonlinear** methods (**trees**, **ensemble** methods, and **neural networks**)
- 2 Weeks: **Unsupervised** learning: **clustering** and **matrix factorization**
- **More specific tentative Syllabus on the webpage.**

High Level Goals of the Class

- Learn fundamental building blocks of machine learning
- Goal is to start seeing
  - **fancy new method A "is just" familiar thing B + familiar thing C + tweak D**
  - SVM "**is just**" ERM with hinge loss with ℓ_2 regularization
  - Pegasos "**is just**" SVM with SGD with a particular step size rule
  - Random forest "**is just**" bagging with trees, with an interesting tweak on choosing splitting variables

Level of the Class

- We will learn how to build all ML algorithms **from scratch** – no ML libraries, just numpy.
- Once we have built it from scratch once, we can use the sklearn version.