Syllabus

CPSC 440/550: Advanced Machine Learning

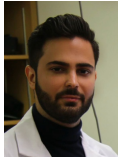
cs.ubc.ca/~dsuth/440/23w2

University of British Columbia, on unceded Musqueam land

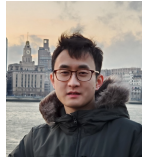
2023-24 Winter Term 2 (Jan–Apr 2024)

Hi!

- Danica Sutherland – `cs.ubc.ca/~dsuth` – ICICS X539 – she/her (🏳️‍⚧️)
  - Feel free to call me "Danica," or "Professor Sutherland" / "Dr. Sutherland" are fine
  - At UBC since 2021
  - Currently: one (shared) postdoc, five PhD/PhD-track students, one course MSc
  - Representation learning, kernel methods, statistical testing, learning theory
  - Grad course on statistical learning theory (CPSC 532D)
  - Previous places: TTI-Chicago (baby faculty), University College London (postdoc), Carnegie Mellon (PhD; Pittsburgh), Swarthmore (undergrad; Philadelphia)
- Teaching Assistants:


Peyman Gholami


Siddhesh Khandelwal


Nathaniel Xu


Yilin (Justin) Yang


Yuwei (Joey) Yin

Where to find stuff

- Course website: `cs.ubc.ca/~dsuth/440`
  - Schedule, slides, assignments, links to everything else
- Piazza
  - Discussion, announcements, etc
  - I mostly won't send any course-related emails
  - You'll get faster, better responses on Piazza than emailing me
  - (Even for things that only affect you: make a private post)
- Gradescope
  - Handing in assignments, seeing graded ones
- Canvas
  - Posting some files, links to other places
  - Should be able to access even if you're not registered yet: link from course page

Outline

What is this course?

- A rough goal: "what most grad students in 'core machine learning' should know"
- . . . that isn't already covered in 340/540
- . . . and isn't reinforcement learning (see 422, 532J, 533V)

Take 340/540 first!

- CPSC 340/540 (previously 532M) is the **first course**:
  - Introductory course on data mining and ML
  - Emphasis on implementing core ideas and various applications
  - Most useful and applicable techniques
- CPSC 440/550 (previously 540) is the **second course**:
  - Stuff that requires more time / background than the stuff in 340
  - Assumes a solid background in fundamental ML concepts
  - Assumes a (somewhat) stronger math/CS background than 340
- **Take 340/540 first**
  - 440/550 will seem very random if you haven't
  - Lack of foundation in the fundamentals likely to lead to big mistakes down the road

340/540 content

review

- Slides formatted like this will be "reminders of things you should already know"
  - There won't be a ton of them!
- I'll assume you're familiar with things like:
  - Vector calculus with matrix notation
    - e.g. turning $\arg\min_w \|\mathbf{X}w - \mathbf{y}\|$ into the "normal equations"
  - IID assumption, complexity vs. generalization trade-offs, cross-validation
  - Probabilistic classifiers, maximum likelihood, MAP estimation
  - Radial basis functions; basic idea of kernel methods
  - Basic definitions of deep/convolutional networks
  - Showing a function is convex; L1 versus L2 regularization
  - Stochastic gradient descent; softmax/cross-entropy loss.
  - Ensemble methods; PCA; collaborative filtering
- If you don't know one or two things here, you can probably catch up
- If you can't already implement most of the above in code, you might struggle

So, what's in *this* course?

- density estimation
- probabilistic / Bayesian points of view
- a smattering of more theory than in 340
- leading (roughly) to Transformers – but for NLP focus, see 436N/532V/503
- probabilistic graphical models
- mixture and latent variable models
- Monte Carlo, variational inference, MCMC
- leading (roughly) to image generative models (diffusion)

- There'll be a bunch of cool stuff left out, out of necessity
- Not a "deep learning course," but deep learning is integrated in throughout

Doesn't a lot of that sound like statistics?

"If you're analyzing data and proving theorems about it in [ESB], that's statistics. If you're doing it in [ICICS], that's machine learning."

— Larry Wasserman
(who said it with the equivalent buildings at CMU)

Statistical Science
2001, Vol. 16, No. 3, 199–231

Statistical Modeling: The Two Cultures

Leo Breiman

Prerequisites

- Undergrads don't have a choice: CPSC 340 and CPSC 320 are required
  - Exceptions need to go through CS advising; I don't control this process

- Grad students: you should know ∼all of the content in 340
- 320 is "Intermediate Algorithm Design and Analysis"; you should know
  - Dynamic programming
  - Graphs, as mathematical objects and as data structures
  - Be very comfortable with big-O asymptotic analysis

Auditing

- Auditing is a great option, esp. if you're not sure on background/commitment:
  - Transcript shows pass/fail instead of a grade
  - Usual UBC requirement is "everything but the final", but we need only **one** of
    - hand in one assignment (with passing grade)
    - write a two-page report on one technique from class
    - attend $>90\%$ of classes
- Please audit officially, or talk to me if for some reason you can't
- Strongly expect we'll have space for auditors
  - I'll sign forms next week, just in case

Lectures

- I'll post "handout" versions of slides before class
- I may or may not post marked-up versions after class; tbd

- Please ask questions: others probably have similar questions
  - I might sometimes deflect to later lectures, or to after class / Piazza

- Warning that this course will move fast and cover a lot of topics
  - I'll try to emphasize the big, important ideas and cover them carefully
  - But many topics will only be covered briefly

- Isn't it better to have deep knowledge of things we cover?
  - Mark Schmidt (who created this course) strongly thinks it's better to know many methods than five in detail. If you know the key idea of a method, you can know when to use it and look up details later
  - I don't know that I fully agree with Mark on the trade-off here
  - Level of detail may be a little in flux as we go

Lecture recordings

- Lectures will be recorded on Panopto (link from Canvas/Piazza)
- Please come to class anyway

- Intended to help you review things you didn't get the first time, or to help if you're sick and miss a class, etc.
- But it's *really easy* to fall behind remotely, and this course moves fast
- Many other instructors don't offer recordings for this reason

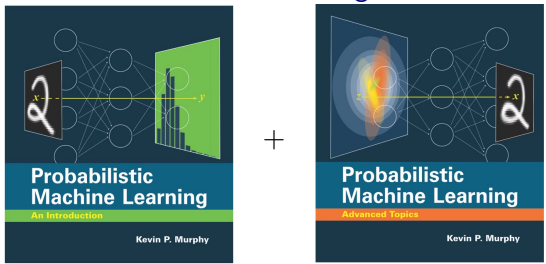
A warning up front

- This is the fourth time CPSC 440 has been offered; my second time
  - Each time has been prepared in a real rush, for a variety of logistical/personal reasons
- The course isn't as "put together" as you might like
  - Definitely much less mature than e.g. 340
- Trying some organizational changes this year

- It's also a hard course – harder than 340, which is definitely not easy
  - We cover a lot of material, and the assignments are long.
- I hope that if you care about the subject, and put in effort, it'll be rewarding
- But expect to have to put in real effort

Textbook

- Kevin Murphy's "Probabilistic Machine Learning" series covers most course topics



- Free PDFs from `probml.github.io`; can order hardcopy of volume 1
- **Not required reading**, it's a supplement to lectures
- I'll point to relevant sections on the course page as we go

Other supplementary sources

bonus!

- Good book for catching up on mathematical background:
  - Mathematics for Machine Learning (Deisenroth, Faisal, Ong) – `mml-book.com`
- All of Statistics (Wasserman)
- Elements of Statistical Learning (Hastie et al.)
- Pattern Recognition and Machine Learning (Bishop)
- Deep Learning (Goodfellow et al.)
- Probabilistic Graphical Models (Koller and Friedman)
- Bayesian Data Analysis (Gelman et al.)

Bonus slides

bonus!

- "Bonus slides" will look like this one
  - Some mention advanced variants of methods we cover
  - Some point to big topics we don't have time for
  - Some cover technical details I don't need you to know
- You don't have to learn the material from these slides
  - But you might find them interesting, or useful in the future

- There are also often "post-lecture" bonus slides at the end of the slides

Getting help

- Course-related questions should go on Piazza
  - Feel free to make private posts that you think are only relevant to you
  - If we disagree, we'll make it public without asking

- (Almost)-weekly tutorials, starting next week
  - Run by TAs; covering related material, mostly helping with assignments
  - Tuesdays 3-4, Wednesdays 12-1 and 5-6, Fridays 3-4
  - Not required; you can go to any tutorial section, regardless of registration
  - On the very off chance it's full, priority to those registered

- Office hours from me and the TAs, starting end of this week
  - Google calendar linked from Piazza+Canvas

Outline

40%: Assignments

- Four (probably) assignments:
  - Conceptual questions, math, and Python
  - Coding portions are Python only, no Julia/R/Matlab/...
  - Do in LaTeX and hand in on Gradescope (instructions soon)

- First assignment **due next Friday (the) at noon**
  - Released in next few days;
  - No late handin; do it alone
  - Will be a little shorter than subsequent assignments, but still a lot of work; start early
  - Intentionally due (a few hours) before add/drop deadline
    - If you realize you're in over your head, you can drop
  - Mostly on prerequisite material

- Later assignments: will allow working with a partner

Working together

- Again, do assignment 1 alone.
- Later assignments: encouraged to use a partner
- But *don't* say "you do question 1 and I'll do question 2"
  - If you hand something in with your name on it, you're promising that you were involved in and understand the full solution

- If you refer to anything other than class notes or the PML book, cite that source
  - Just a link/etc is fine, doesn't have to be bibtex
- General discussion with friends (not in your group) is okay
  - Cite them ("My friend Carlos suggested that...")
  - "General discussion" doesn't include them writing code for you, telling you exactly what math step to take, etc.
- Don't search specifically for answers to the questions we're asking
  - If you happen across one, then just say so and cite it
- **Do not use ChatGPT, Copilot, etc for the assignments**
  - Different policy for the project

Late assignments

- For assignments other than assignment 1, you have seven late days
- No penalty, no questions asked, no need to "declare" (just hand it in late)
  - If assignment is due Friday midnight and you hand it in the next Sunday afternoon, that's two late days from each member of your group
  - If Alice has 4 late days and Bob has 1, Alice+Bob group can hand in 4 days late
    - If you go negative, no penalty, but then no more late days

- No other extensions except in *very* exceptional circumstances
  - Official academic concessions process for medical issues, death in the family, etc.
  - **Not** "I had a lot of other assignments" or "there was a conference deadline"
  - You can submit more than once; do what you can by the deadline if I haven't already approved an extension
  - The assignments will be out for a long window; don't wait to the last minute

20%: Quizzes

- Similar to the "very short answer" questions from past years (or 340/540)
- In the CBTF (Computer-Based Testing Facility): `cbtf.cs.ubc.ca`
- Self-scheduled over a few-day period roughly every other week
- 50-minute window, intended to take much less time than that
- Questions will be somewhat randomized
- Do not discuss with others until after the quiz period is over
- First time I'm doing this (or any ML course here)
- Reserve the right to decide it's not working and convert future quizzes to more assignment questions and/or final weight
- If you have a *very good reason* why you can't do them in person, talk to me

40%: Final and/or project

- 440 students: 40% of course grade is maximum of final exam and course project
  - You can choose to do only one or the other
- 550 students: 20% of course grade is final exam, 20% is course project

Final exam

- Scheduled by UBC, date currently unknown
  - Unless you're sure you won't take it, don't make plans to travel before April 27

- In-person, handwritten, traditional exam
- Closed book, with three pages of double-sided reference sheets

- Mix of "conceptual" questions and more "technical" questions
- All lecture material, except "bonus slides," is fair game
- (I'll of course try not to depend too much on some minor point)
- More details later in the term

Course project

- More details later in the term
- Smaller scope than projects in most classes
- Groups of 2-3
- Short proposal due sometime in second half of the term
- Due on the last day of exams
- Any programming language, Copilot/etc is fine
- Goal is to produce something roughly like a workshop paper
- Some overlap with your own research is fine

Summary

- Hard class, sequel to 340/540, covers cool stuff

- 40% assignments
  - Probably four assignments
  - A1 alone, no late days
  - A2+ with a partner (or alone), 7 total late days over the term
- 20% quizzes
  - Self-scheduled, about every other week, short conceptual questions
- 40%: `(max if student.in_440 else mean)`(final, project)
  - In-person, handwritten final
  - Small research-type project due last day of exams (small proposal earlier)

UBC provides resources to support student learning and to maintain healthy lifestyles but recognizes that sometimes crises arise and so there are additional resources to access including those for survivors of sexual violence. UBC values respect for the person and ideas of all members of the academic community. Harassment and discrimination are not tolerated nor is suppression of academic freedom. UBC provides appropriate accommodation for students with disabilities and for religious and cultural observances. UBC values academic honesty and students are expected to acknowledge the ideas generated by others and to uphold the highest academic standards in all of their actions. Details of the policies and how to access support are available here: `senate.ubc.ca/policies-resources-support-student-success`

Quick course history

bonus!

- Started out as Mark Schmidt's grad intro to ML (CPSC 540), independent of 340
- Gradually, 340 added on pre-reqs and covered more material
- Advanced optimization material split from 540 into "CPSC 5XX" (later: 532M)
- Slowly became more of a sequel to 340
- 2020-21: cross-listed as 440, revamped to modernize and be more "undergrad-friendly"
- 2022w2: I took over, changed Julia $\rightarrow$ Python but otherwise similar
- 2023w2: mild further changes to keep "modernizing"