Georgia Tech
College of Computing

CX4240 Spring 2026

Intro to Computing for Data Analysis

Bo Dai
School of CSE, Georgia Tech
bodai@cc.gatech.edu

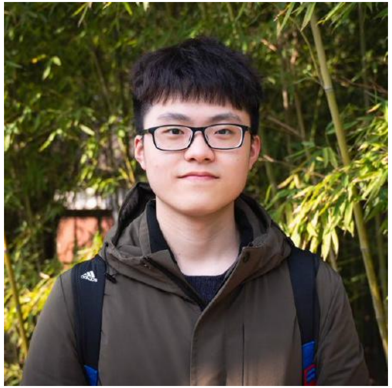
Bo Dai

Assistant Professor
School of Computational Science and Engineering

Address: Coda E1342A
Email: bodai@cc.gatech.edu
Homepage: https://bo-dai.github.io

Research: Reinforcement Learning, Generative AI
https://bo-dai.github.io/CX4240-spring2026/

Teaching Assistant



Chenxiao Gao

Address: Coda East Wing
Email: cgao@gatech.edu

Changhao Li

Address: Coda East Wing
Email: cli911@gatech.edu

Logistics

Time: Monday/Wednesday 3:30-04:45 pm

Location: [Molecular Sciences and Engr G011]

Discussion & HW submission: Ed Discussion & Canvas

Course Website https://bo-dai.github.io/CX4240-spring2026/

Office Hour:

- Instructor: TBD
- TA: TBD

What to cover in this course?

"This course introduces techniques for computational data analysis, with an emphasis on machine learning algorithms and their applications to real-world data."

-----------Quote from previous syllabus

What is Machine Learning?

Machine learning (**ML**) is an umbrella term for solving problems for which development of algorithms by human programmers would be cost-prohibitive, and instead the problems are solved by helping machines 'discover' their 'own' algorithms,[1] without needing to be explicitly told what to do by any human-developed algorithms.

-- Wikipedia

Machine learning is a branch of artificial intelligence (AI) that focuses on developing computer systems that can learn and adapt without explicit programming. Instead of following rigid rules, these systems learn from data and improve their performance over time.

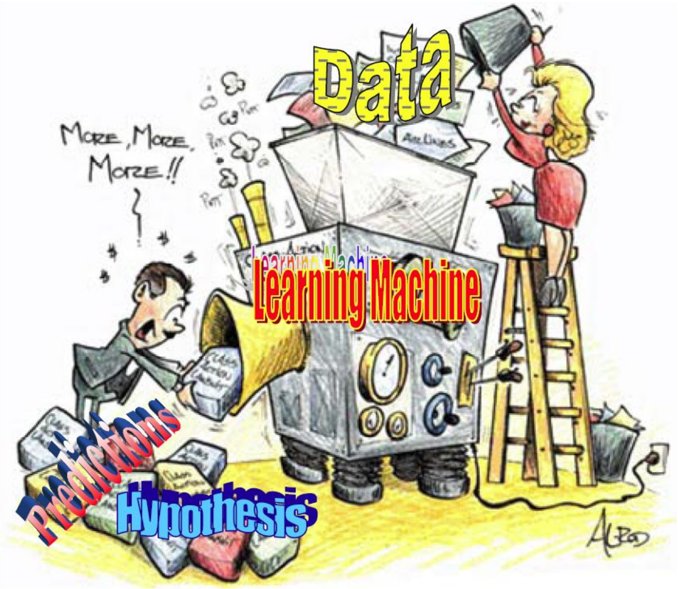
-- Gemini

Machine learning (ML) is a subset of artificial intelligence (AI) focused on the development of algorithms and statistical models that allow computers to learn from and make predictions or decisions based on data. Instead of being explicitly programmed to perform specific tasks, machine learning systems use data to identify patterns and improve their performance over time.

-- ChatGPT

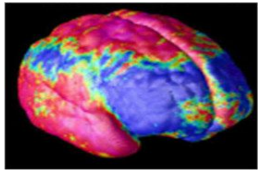
Personal Opinion

- Machine Learning is a subfield of AI
- Machine Learning focuses on a special type of algorithm design
    - These algorithms consume data, generates a model for prediction and decision



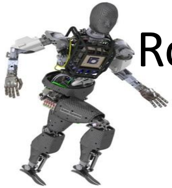
Machine Learning Applications

Brain

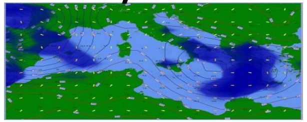


Galaxy

Self-driving car



Robot



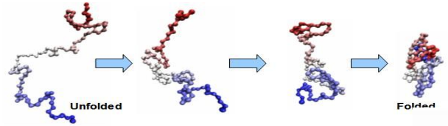
Weather

Finance

Genome

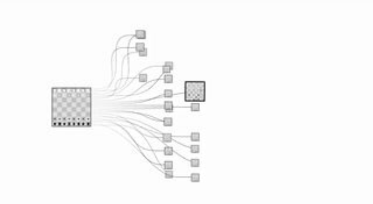
Music

Sustainability



Protein

Game



Chatbot

Syllabus

Cover a number of most commonly used machine learning algorithms in sufficient amount of details on their mechanisms.

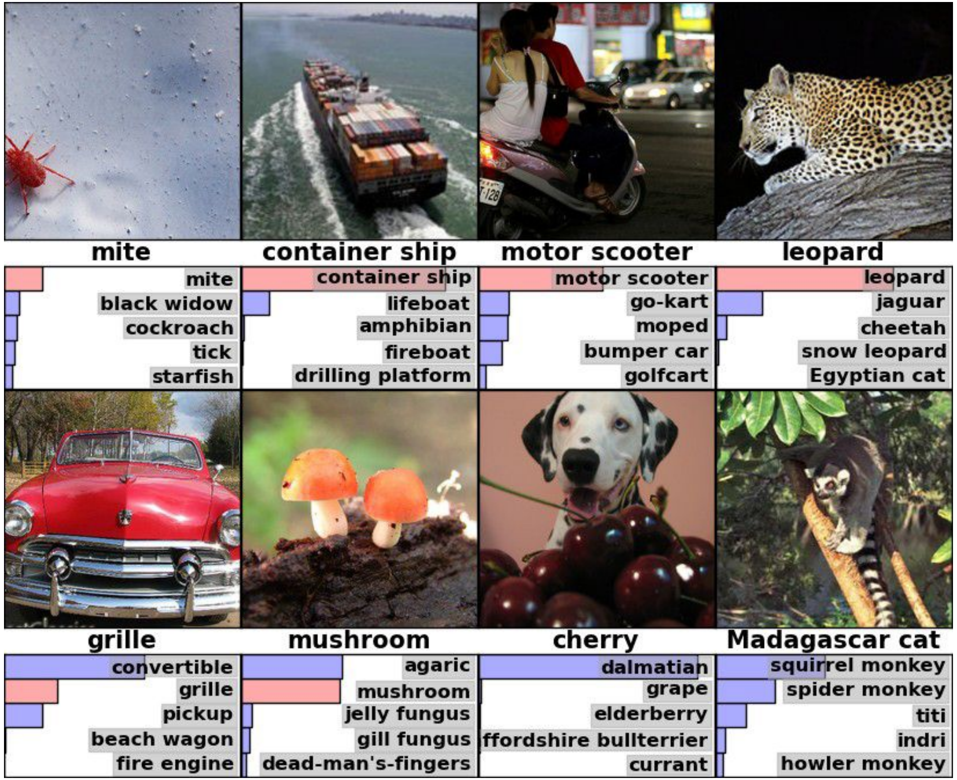
Organization

- *Background knowledge*
- *Supervised learning*
  - Learning with labels, focusing on predictive performance
- *Unsupervised learning*
  - Learning without labels or without optimizing for predictive task
- *Advanced Topics*
  - Foundation Models Training

Syllabus: Supervised Learning

- Learning with labels, focusing on predictive performance (limited data)
  - Linear Models
    - Classification: Naïve Bayes classifier vs. Logistic regression
    - Regressions: Linear Ridge regression
  - Nonlinear Linear Models
    - Neural Network: CNN, RNN

Image Classification

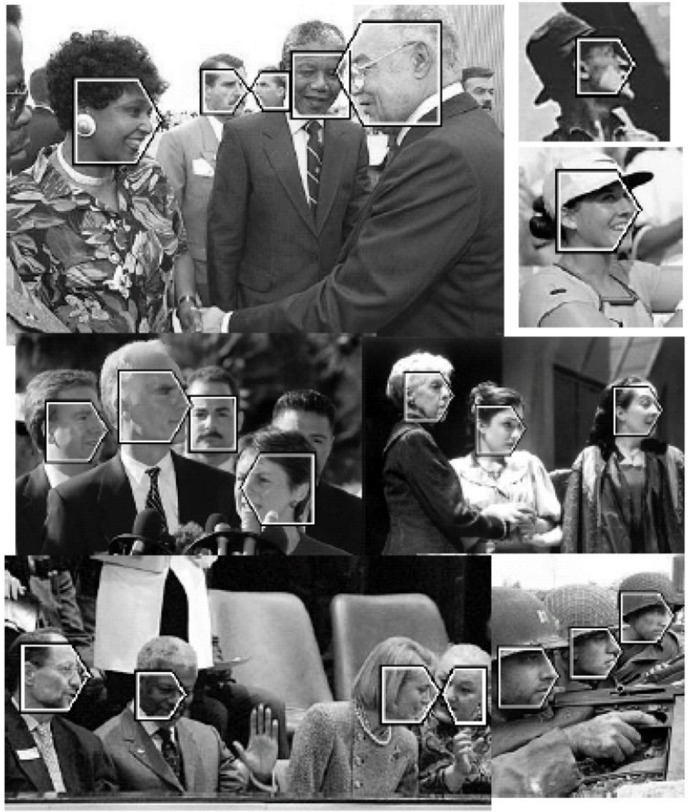


What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

Face Detection



What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

Weather Prediction

Predict

Numeric values:
40 F
Wind: NE at 14 km/h
Humidity: 83%

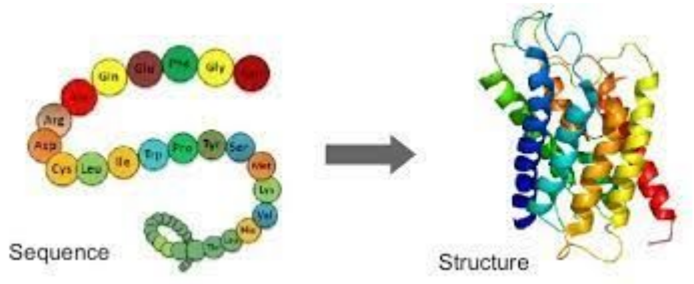
Predict

What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

Protein Prediction



What are the desired outcomes?

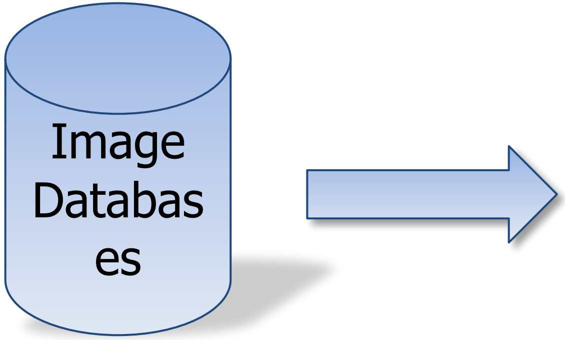
What are the inputs (data)?

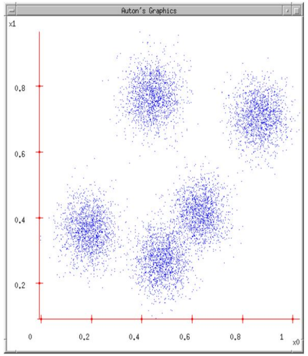
What are the learning paradigms?

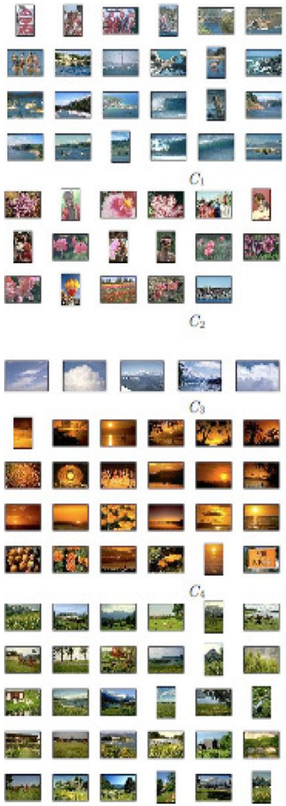
Syllabus: Unsupervised Learning

- Learning without labels or without optimizing for predictive task (almost unlimited data)
  - Clustering
    - K-means vs. Gaussian Mixture Models
  - Generative Models
    - Gaussian Mixture Models vs. Variational AutoEncoder
  - Dimension Reduction and Representation Learning
    - Principal Component Analysis vs. Neural Contrastive Representation

Organizing Images





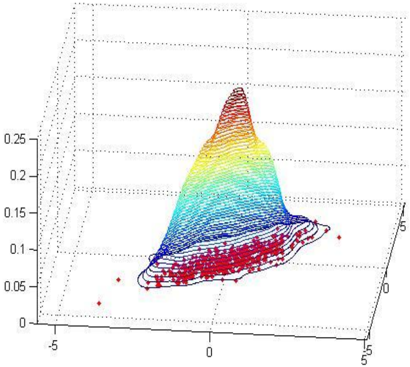


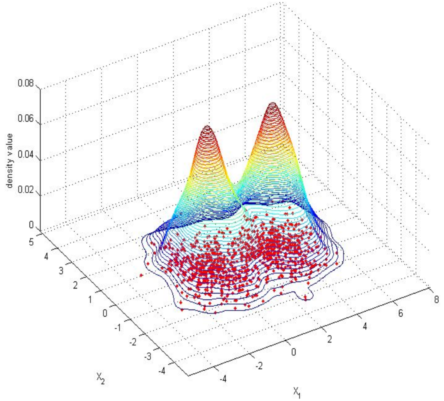
What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

Generative Models



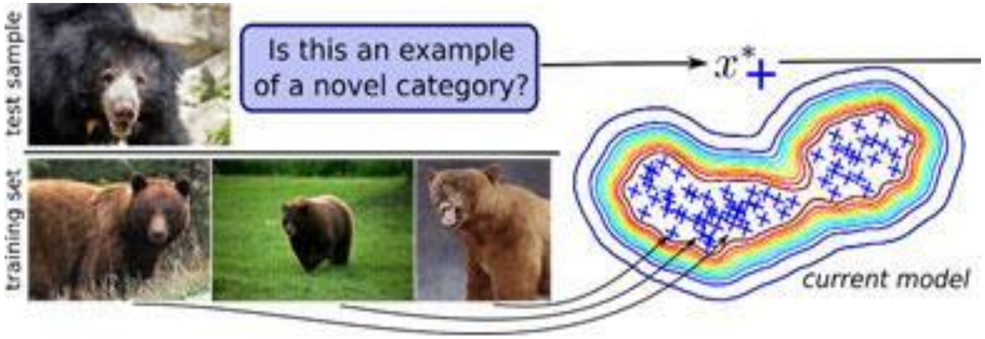


What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

Novelty/Abnormality Detection

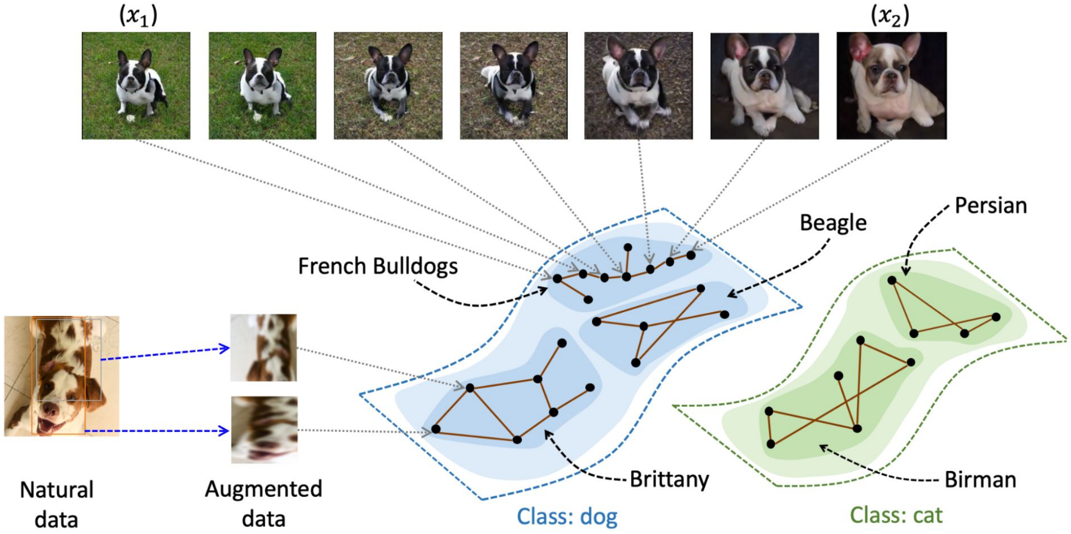


What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

Representation Learning

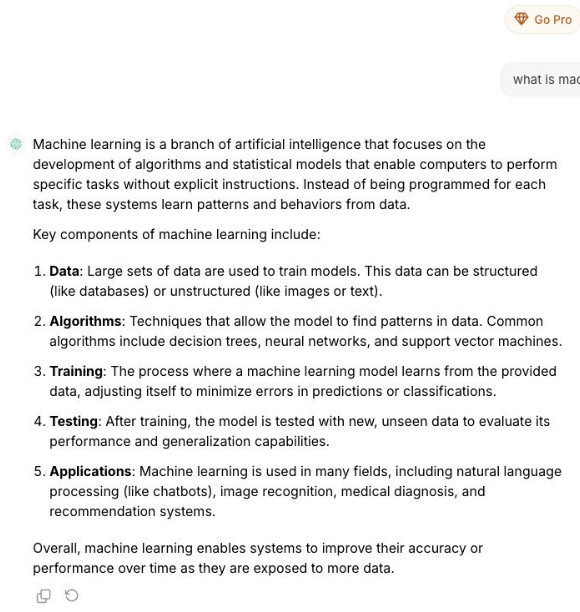


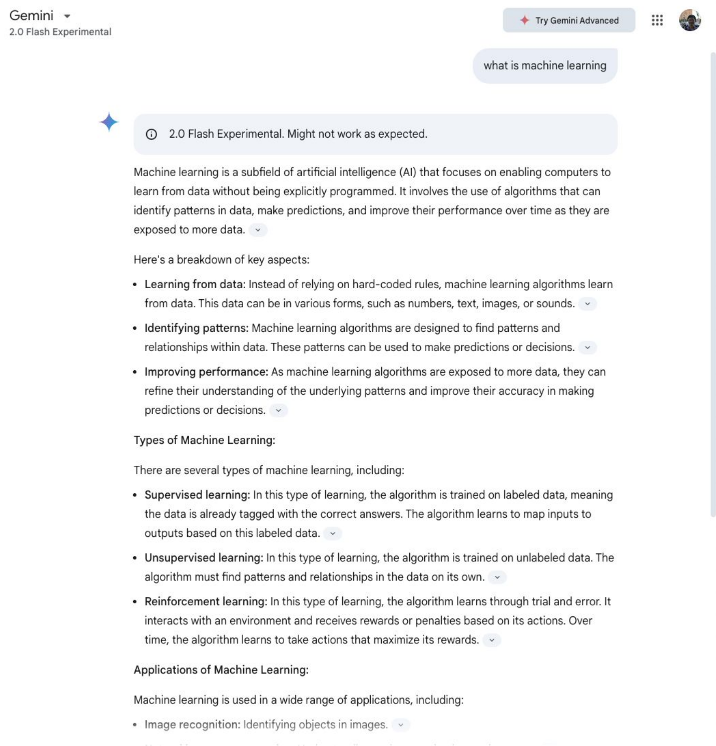
What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

Syllabus: Large Language Models

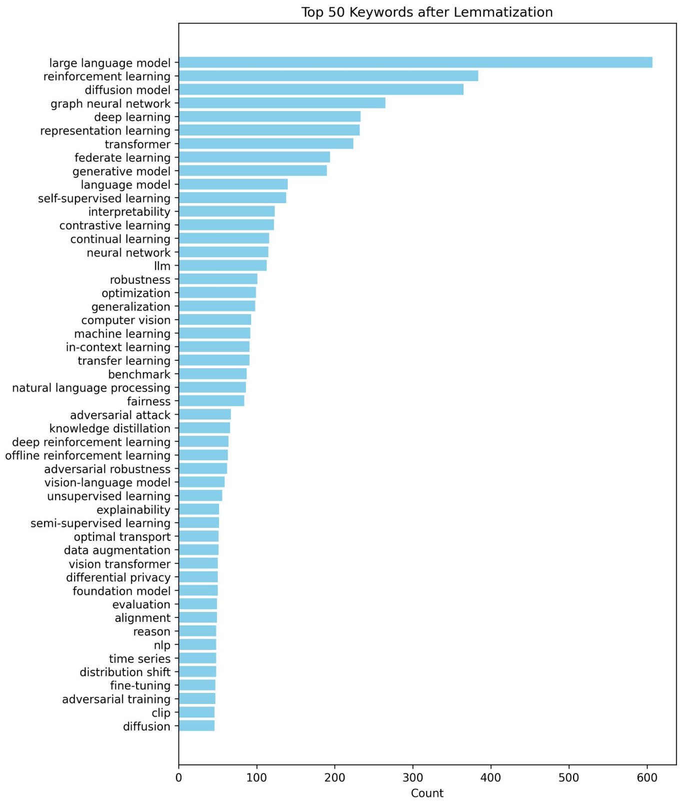


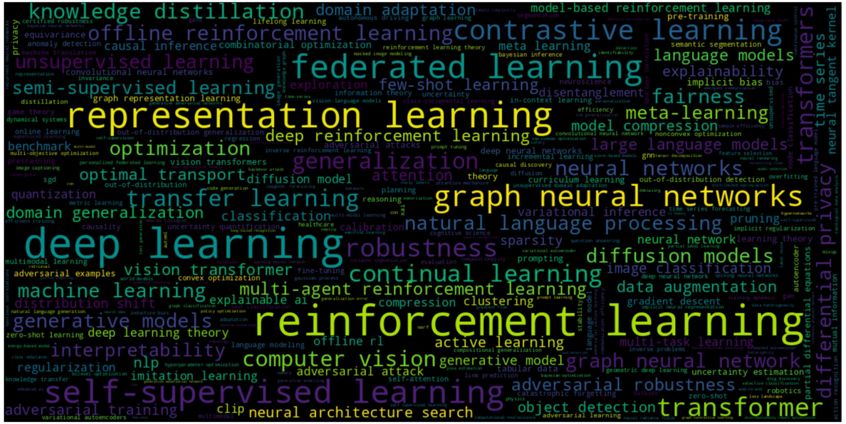


Syllabus: Large Language Models

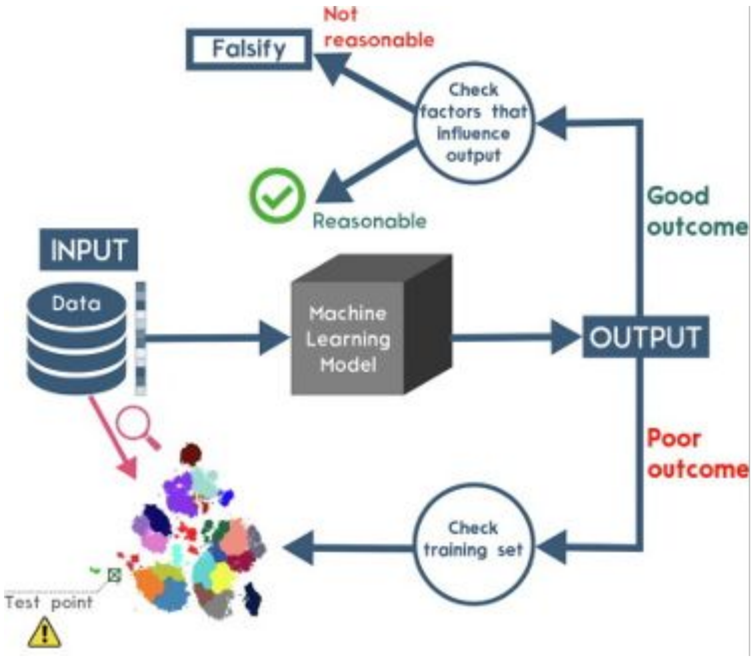
- Language Model Architecture
  - Attention, Transformer
- Supervised Fine-Tuning
- Reinforcement Learning

Modern Topics in Machine Learning

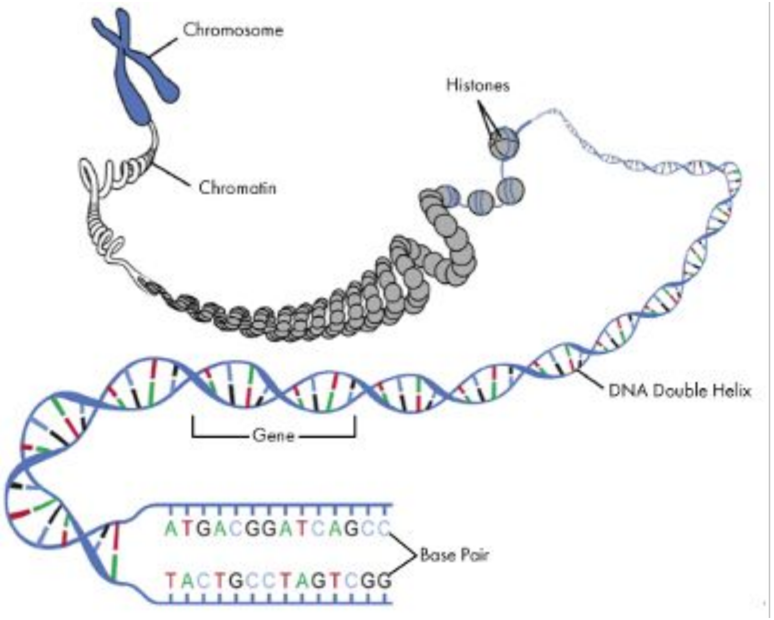


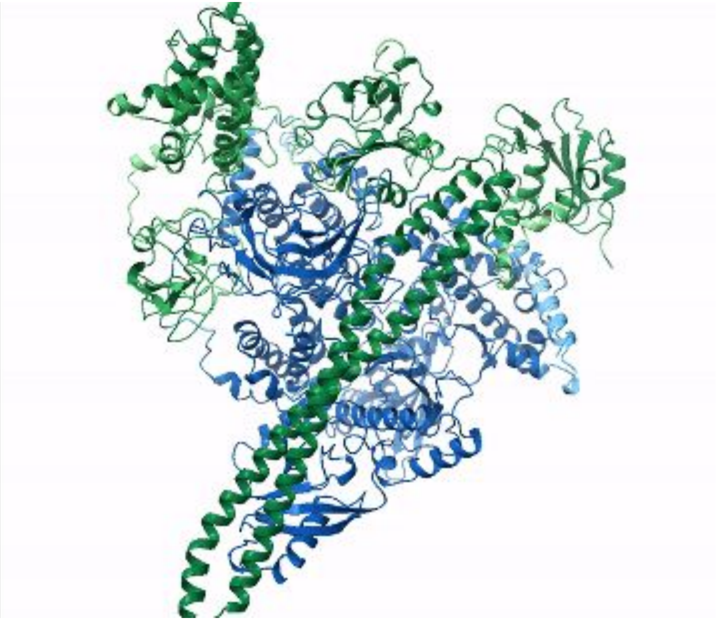


Industrial Engineering

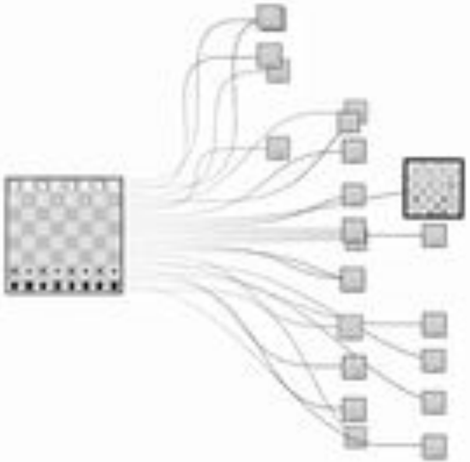


Bioinformatics





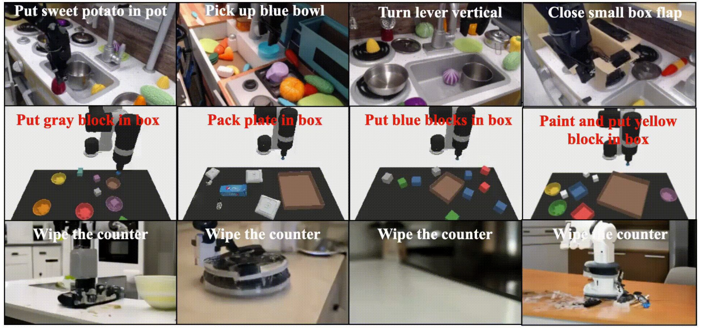
AlphaGo



Robotics

TIBHAR

Google



Put sweet potato in pot
Pick up blue bowl
Turn lever vertical
Close small box flap
Put gray block in box
Pack plate in box
Put blue blocks in box
Paint and put yellow block in box
Wipe the counter
Wipe the counter
Wipe the counter

Syllabus

Cover a number of most commonly used machine learning algorithms in sufficient amount of details on their mechanisms.

Organization in a unified probabilistic framework

- *Background knowledge*
- *Supervised learning*
    - Learning with labels, focusing on predictive performance
- *Unsupervised learning*
    - Learning without labels or without optimizing for predictive task
- *Advanced Topics*
    - Foundation Models Training

Textbooks

- Bishop. Pattern Recognition and Machine Learning. Springer. 2006
- Goodfellow, Bengio, and Courville. Deep Learning. MIT Press. 2016

The material of the class may go beyond these books

Basic / Prerequisites

- Probability
  - Distributions, densities, marginalization, conditioning
- Statistics
  - Mean, variance, maximum likelihood estimation
- Linear Algebra and Optimization
  - Vector, matrix, multiplication, inversion, eigen-value decomposition
- Coding Skills
  - Pytorch and/or JAX

Machine Learning for Apartment Hunting

- Suppose you are to move to Atlanta
- And you want to find the most reasonably priced apartment satisfying your needs:

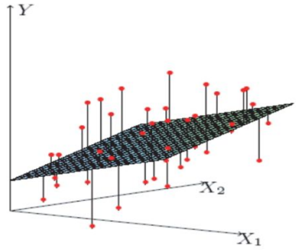
Living area (ft^2)	# bedroom	Monthly rent ($)
230	1	900
506	2	1800
433	2	1500
190	1	800
…		
150	1	?
270	1.5	?

Probabilistic Interpretation of Least Mean Square

- Assume y is a linear in x plus noise ϵ

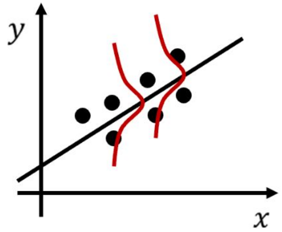
$$y = \theta^\top x + \epsilon$$

Linear algebra



- Assume ϵ follows a Gaussian $N(0, \sigma)$

$$p(y^i | x^i; \theta) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\left(y^i - \theta^\top x^i\right)^2}{2\sigma^2}\right)$$



- By independence assumption, likelihood is

$$L(\theta) = \prod_i^m p(y^i | x^i; \theta) = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^m \exp\left(-\frac{\sum_i^m \left(y^i - \theta^\top x^i\right)^2}{2\sigma^2}\right)$$

Probability

Probabilistic Interpretation of LMS, cont.

- Hence the log-likelihood is:

$$\log L(\theta) = m \log \frac{1}{\sqrt{2\pi}\sigma} - \frac{1}{2\sigma^2} \sum_{i}^{m} \left(y^i - \theta^\top x^i\right)^2$$

Statistics

- LMS is equivalent to MLE of θ !

$$LMS: \quad \frac{1}{m} \sum_{i}^{m} \left(y^i - \theta^\top x^i\right)^2$$

- How to make it work in real data?

Algorithms
Programming

Matrix version of the gradient

- Define $X = (x^1, x^2, \dots x^m)$, $y = (y^1, y^2, \dots, y^m)^\top$, gradient becomes

Linear algebra

$$\frac{\partial L(\theta)}{\partial \theta} = -\frac{2}{m} Xy + \frac{2}{m} XX^\top \theta$$

Linear algebra

$$\Rightarrow \hat{\theta} = (XX^\top)^{-1} Xy$$

Algorithms
Programming

- Matrix inversion in $\hat{\theta} = (XX^\top)^{-1} Xy$ expensive to compute
  - Gradient descent

$$\hat{\theta}^{t+1} \leftarrow \hat{\theta}^t + \frac{\alpha}{m} \sum_i^m \left(y^i - {\hat{\theta}^t}^\top x^i \right) x^i$$

Optimization

Basic / Prerequisites

- Probability
  - Distributions, densities, marginalization, conditioning
- Statistics
  - Mean, variance, maximum likelihood estimation
- Linear Algebra and Optimization
  - Vector, matrix, multiplication, inversion, eigen-value decomposition
- Coding Skills
  - Pytorch and/or JAX

Grading

- Homework (30%)
- Project (40%)
- Exam (30%)
- Bonus: Participation (5%) + Others (x%)

Homework

- There will be three assignments, each account for 10% towards your final score.

- Late policy: Assignments are due at 11:59PM of the due date. You will be allowed 2 total late days (48 hours) without penalty for the entire semester (for homework only, not applicable to exams or projects). Once those days are used, you will be penalized according to the following policy:
  - Homework is worth full credit before the due time.
  - It is worth 75% credit for the next 24 hours.
  - It is worth 50% credit for the second next 24 hours.
  - It is worth zero credit after that.

Homework

You are required to use Latex (OverLeaf Latex Example in the Video), or a word processing software to generate your solutions to the written questions. Handwritten solutions WILL NOT BE ACCEPTED.

You can easily export your Jupyter Notebook to a Python file and import that to your desired python IDE to debug your code for assignments.

Project

Team Size

Each project must be completed in a team of 3-5. Once you have formed your group, please send one email per team to the class instructor list with the names of all team members. If you have trouble forming a group, please send us an email and we will help you find project partners.

The team formation email will be due at 11:59 PM on Feb 16th.

Project

Project Topics:

- Reproduce classic papers, include but not limited to:
  - Deep Residual Learning for Image Recognition
  - Auto-Encoding Variational Bayes
  - A Simple Framework for Contrastive Learning of Visual Representations.
  - Sequence to Sequence Learning with Neural Networks
  - Efficient Estimation of Word Representations in Vector Space
  - etc
- You may also refer to the https://cs231n.stanford.edu/project.html.

Project

Deliverables:

- Presentation (15%)
- Final Report (25%): *All write-ups should use the NeurIPS style.*

  Your final report is expected to be up to 6 pages **excluding references.** *It should have roughly the following format:*

  - *Introduction: problem definition and motivation*
  - *Background & Related Work: background info and literature survey*
  - *Methods – Overview of your proposed method – Intuition on why should it be better than the state of the art – Details of models and algorithms that you developed*
  - *Experiments – Description of your testbed and a list of questions your experiments are designed to answer – Details of the experiments and results*
  - *Conclusion: discussion and future work*

The project final report will be due at 11:59 PM on May 4th

Project

Criteria:

- 30% for proposed method (soundness and originality)
- 30% for correctness, completeness, and difficulty of experiments and figures
- 20% for empirical and theoretical analysis of results and methods
- 20% for quality of writing (clarity, organization, flow, etc.)

Exam

One exam will be held on March 18th in lieu of the regular class:

- It will be a closed-book exam, so no notes or communication with peers is allowed.
- There will be no make-up exams, so be sure to attend on the scheduled date. Missing the exam will result in zero credit.

Participation Bonus

We appreciate student participation in the class! We will be awarding, on a case-by-case basis, up to 5% in extra credit to the top Ed contributors based on the number of (meaningful) instructor-endorsed answers or other significant contributions that assist the teaching staff or other students in the course. The most helpful contributor will receive the greatest amount of extra credit, and other students with significant contributions will receive a percentage of that.

Tentative Schedule

https://bo-dai.github.io/CX4240-spring2026/lectures/

Background Test

- Complete by yourself
- For my review session preparation, not effect to your grade.

Q&A