

# CS4641 Spring 2025

# Machine Learning

Bo Dai  
School of CSE, Georgia Tech  
[bodai@cc.gatech.edu](mailto:bodai@cc.gatech.edu)



# Bo Dai

Assistant Professor  
School of Computational Science and Engineering

Address: Coda E1342A  
Email: [bodai@cc.gatech.edu](mailto:bodai@cc.gatech.edu)  
Homepage: <https://bo-dai.github.io>

Research: Reinforcement Learning, Generative AI  
<https://bo-dai.github.io/CS4641-spring2025/>

# Teaching Assistant

Leading TA



## Haotian Sun

Address: Coda East Wing

Email: [haotian.sun@gatech.edu](mailto:haotian.sun@gatech.edu)



## Tianyi Chen

Address: Coda West Wing

Email: [tchen667@gatech.edu](mailto:tchen667@gatech.edu)



## Zihao Zhao

Address: TBD

Email: [zzhao628@gatech.edu](mailto:zzhao628@gatech.edu)

# Logistics

Time: Monday/Wednesday 5:00-06:15 pm

Location: College of Computing 16

Discussion & HW submission: Ed Discussion & Canvas

Course Website <https://bo-dai.github.io/CS4641-spring2025/>

Office Hour:

- Instructor: TBD
- TA: TBD

# 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, <sup>[1]</sup> 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.

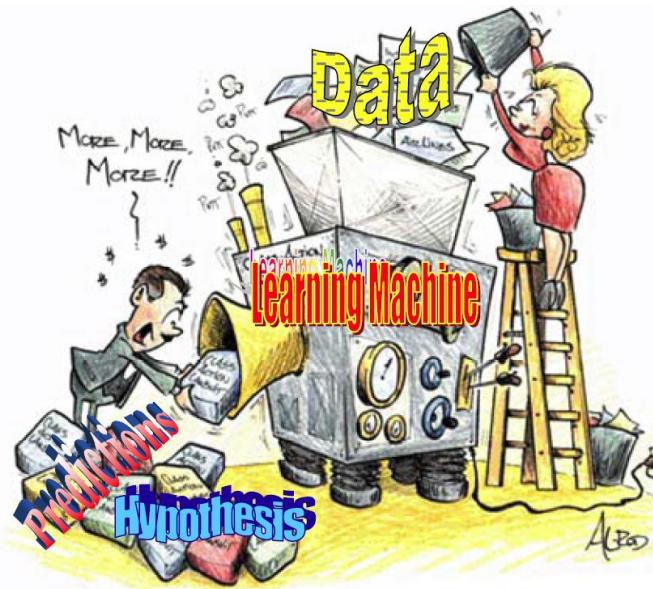
-- Gemini

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

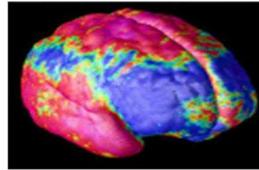
# Personal Opinion

- 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

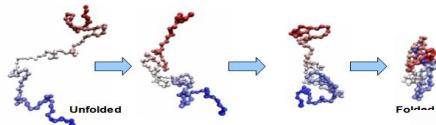


# Machine Learning Applications

Brain

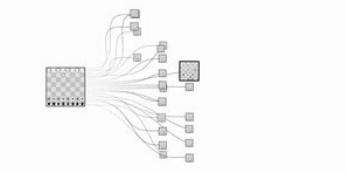


Genom  
e



Protein

Game

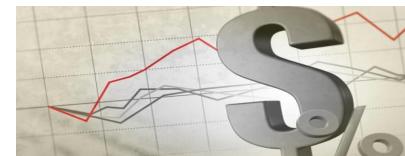


Galaxy

Self-driving car



Robot



Finance

Weather



Music



Sustainability

Chatbot<sub>8</sub>

# 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

# 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

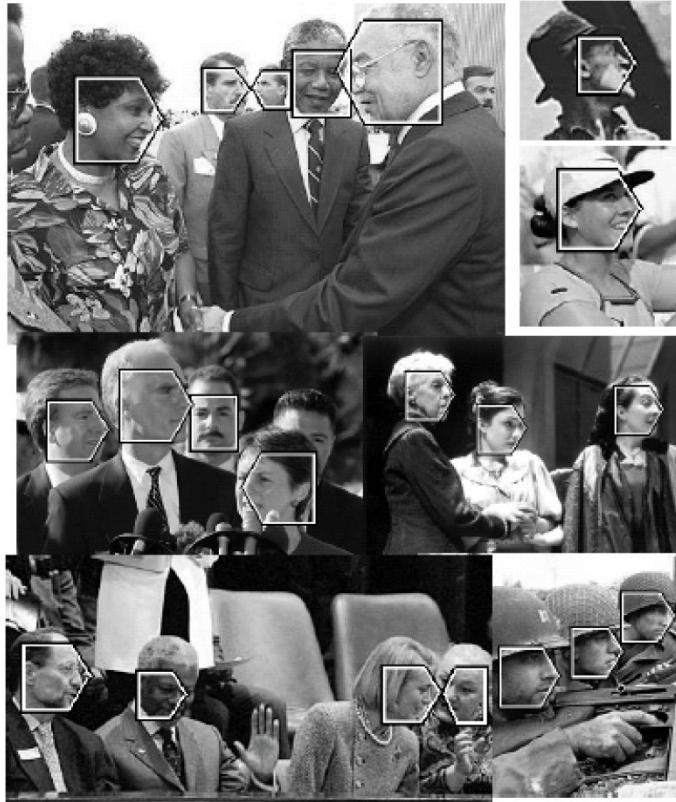
			
mite	container ship	motor scooter	leopard
mite black widow cockroach tick starfish	container ship lifeboat amphibian fireboat drilling platform	motor scooter go-kart moped bumper car golfcart	leopard jaguar cheetah snow leopard Egyptian cat
			
grille	mushroom	cherry	Madagascar cat
convertible grille pickup beach wagon fire engine	agaric mushroom jelly fungus gill fungus dead-man's-fingers	dalmatian grape elderberry ffordshire bullterrier currant	squirrel monkey spider monkey titi indri howler monkey

What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

# Face Detection

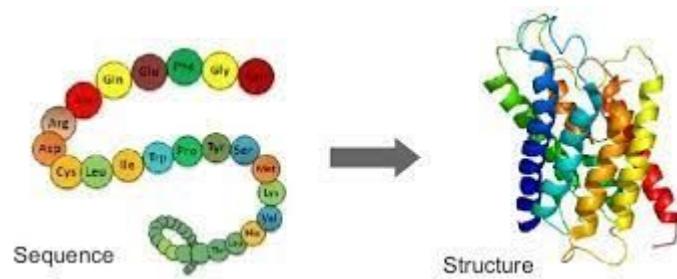


What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

# Protein Prediction



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%

What are the desired outcomes?

What are the inputs (data)?

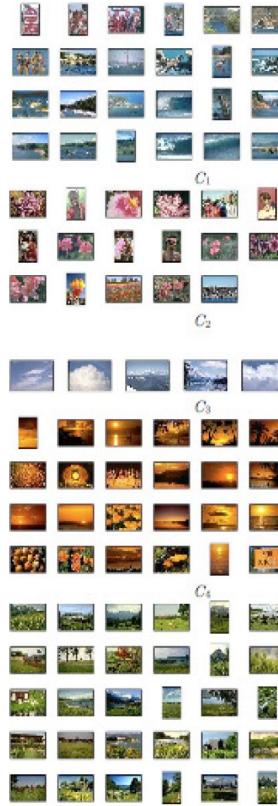
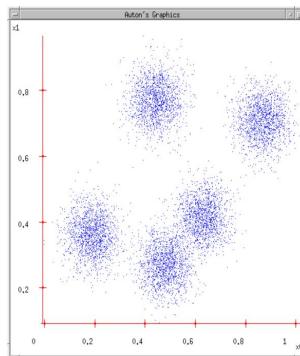
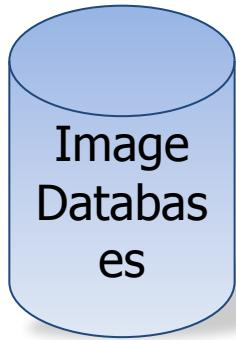
What are the learning paradigms?



# 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

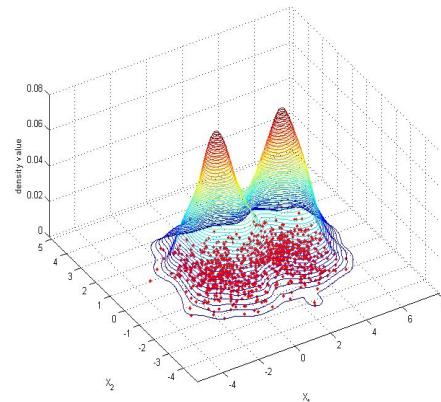
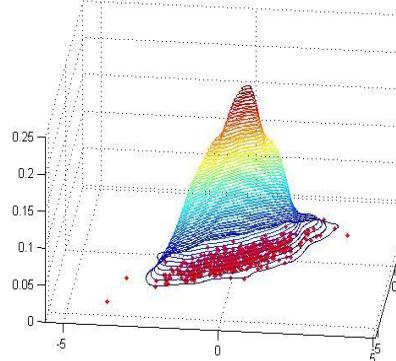


What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

# Generative Models

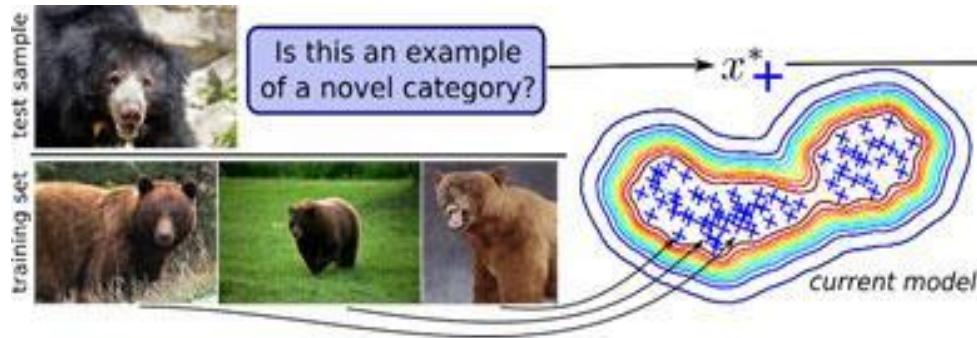


What are the desired outcomes?

What are the inputs (data)?

What are the learning paradigms?

# Novelty/Abnormality Detection



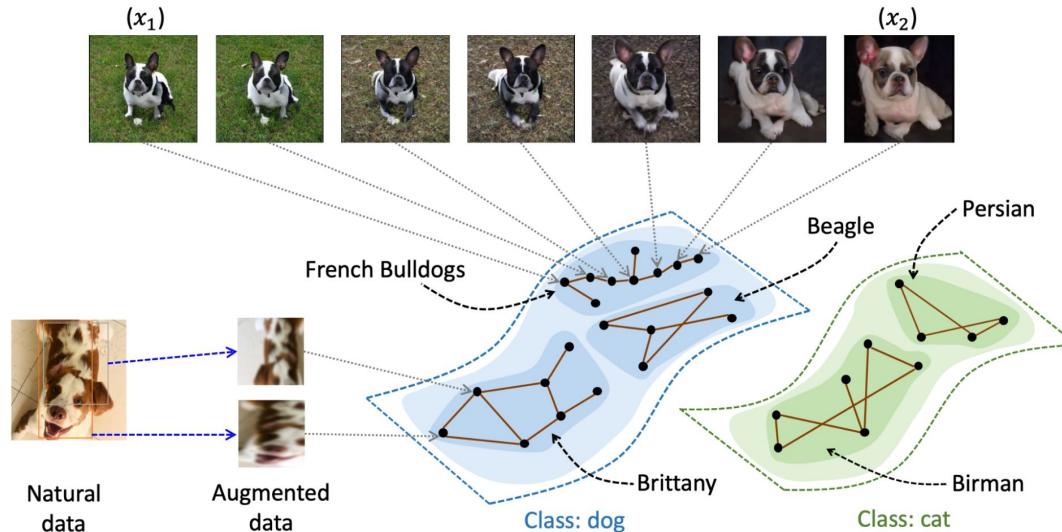
What are the desired outcomes?



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



Machine learning is a branch of artificial intelligence that focuses on the development of algorithms and statistical models that enable computers to perform specific tasks without explicit instructions. Instead of being programmed for each task, these systems learn patterns and behaviors from data.

Key components of machine learning include:

1. **Data:** Large sets of data are used to train models. This data can be structured (like databases) or unstructured (like images or text).
2. **Algorithms:** Techniques that allow the model to find patterns in data. Common algorithms include decision trees, neural networks, and support vector machines.
3. **Training:** The process where a machine learning model learns from the provided data, adjusting itself to minimize errors in predictions or classifications.
4. **Testing:** After training, the model is tested with new, unseen data to evaluate its performance and generalization capabilities.
5. **Applications:** Machine learning is used in many fields, including natural language processing (like chatbots), image recognition, medical diagnosis, and recommendation systems.

Overall, machine learning enables systems to improve their accuracy or performance over time as they are exposed to more data.



2.0 Flash Experimental. Might not work as expected.

Machine learning is a subfield of artificial intelligence (AI) that focuses on enabling computers to learn from data without being explicitly programmed. It involves the use of algorithms that can identify patterns in data, make predictions, and improve their performance over time as they are exposed to more data.

Here's a breakdown of key aspects:

- **Learning from data:** Instead of relying on hard-coded rules, machine learning algorithms learn from data. This data can be in various forms, such as numbers, text, images, or sounds.
- **Identifying patterns:** Machine learning algorithms are designed to find patterns and relationships within data. These patterns can be used to make predictions or decisions.
- **Improving performance:** As machine learning algorithms are exposed to more data, they can refine their understanding of the underlying patterns and improve their accuracy in making predictions or decisions.

Types of Machine Learning:

There are several types of machine learning, including:

- **Supervised learning:** In this type of learning, the algorithm is trained on labeled data, meaning the data is already tagged with the correct answers. The algorithm learns to map inputs to outputs based on this labeled data.
- **Unsupervised learning:** In this type of learning, the algorithm is trained on unlabeled data. The algorithm must find patterns and relationships in the data on its own.
- **Reinforcement learning:** In this type of learning, the algorithm learns through trial and error. It interacts with an environment and receives rewards or penalties based on its actions. Over time, the algorithm learns to take actions that maximize its rewards.

Applications of Machine Learning:

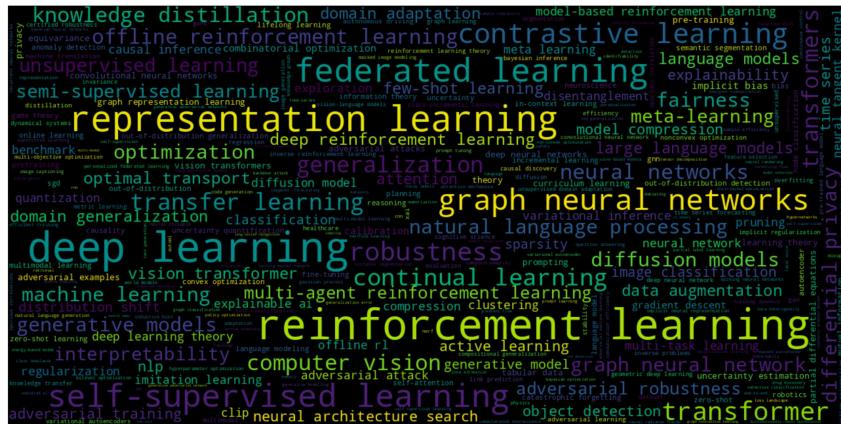
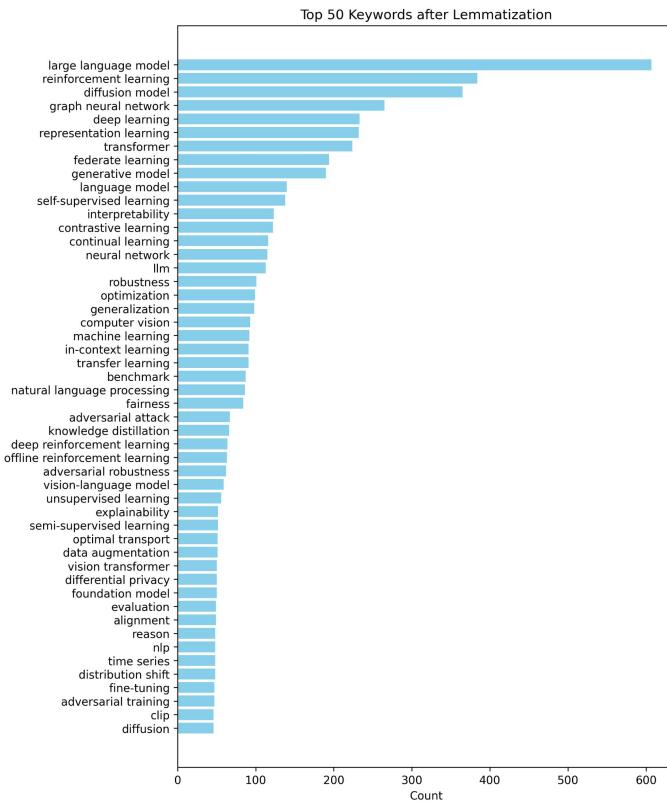
Machine learning is used in a wide range of applications, including:

- **Image recognition:** Identifying objects in images.

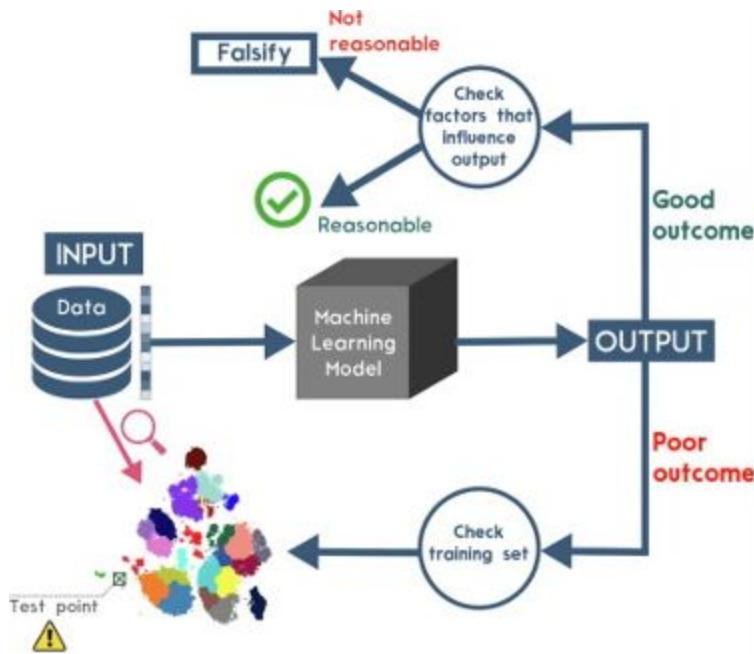
# Syllabus: Large Language Models

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

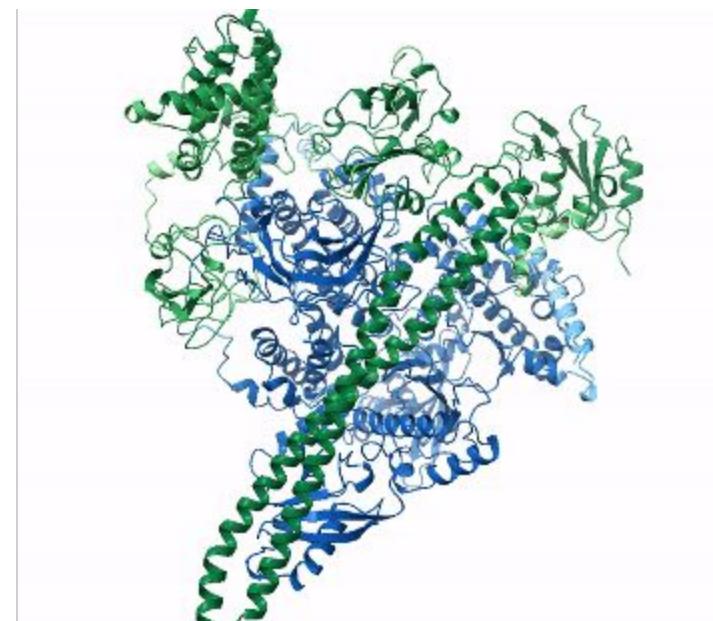
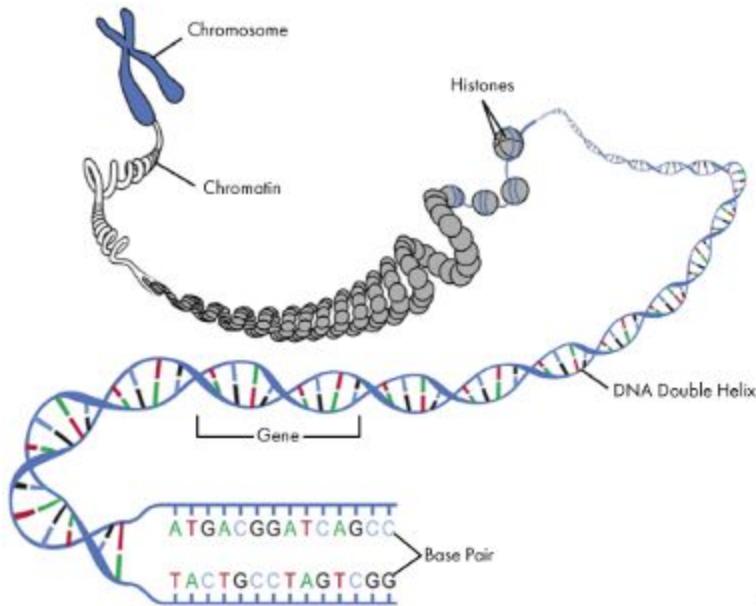
# Modern Topics in Machine Learning



# Industrial Engineering



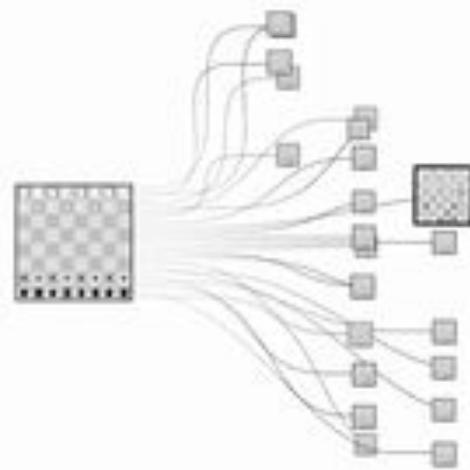
# Bioinformatics



# Robotics



# AlphaGo



# Textbooks

- Bishop. [Pattern Recognition and Machine Learning](#). Springer. 2006
- Goodfellow, Bengio, and Courville. [Deep Learning](#). MIT Press. 2016

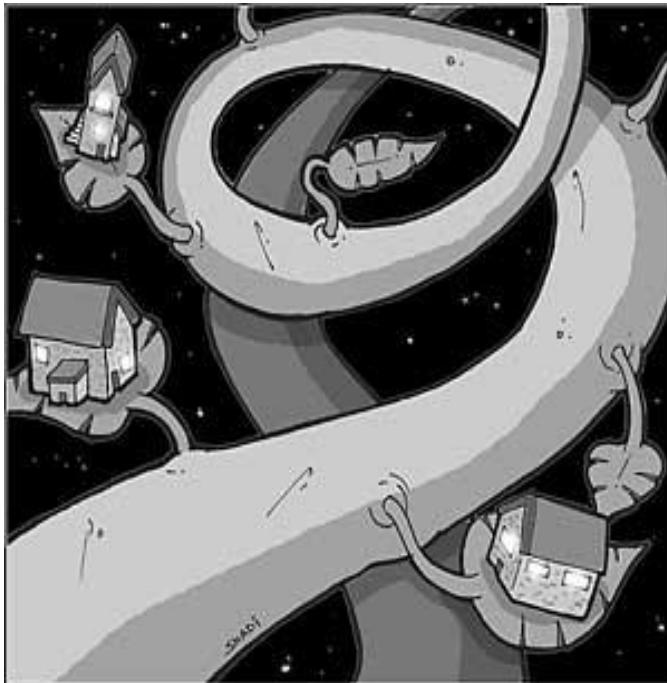
*The material of the class may go beyond these books*

# 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

# 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 <sup>2</sup> )	# bedroom	Monthly rent (\$)
230	1	900
506	2	1800
433	2	1500
190	1	800
...		
150	1	?
270	1.5	?

# Linear Regression Model

- Assume  $y$  is a linear function of  $x$  (features) plus noise  $\epsilon$

$$y = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n + \epsilon$$

where  $\epsilon$  is an error model as Gaussian  $N(0, \sigma^2)$

Probability

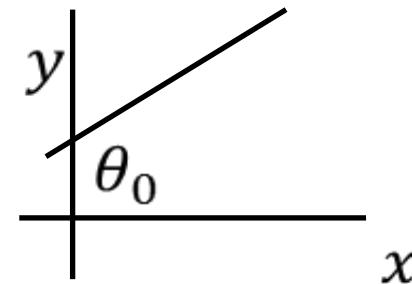
- Let  $\theta = (\theta_0, \theta_1, \dots, \theta_n)^T$ , and augment data by one dimension

Linear algebra

$$x \leftarrow (1, x)^T$$

Then  $y = \theta^T x + \epsilon$

Linear algebra



# Least mean square method

- Given  $m$  data points, find  $\theta$  that minimizes the mean square error

$$\hat{\theta} = \operatorname{argmin}_{\theta} L(\theta) = \frac{1}{m} \sum_{i=1}^m (y^i - \theta^\top x^i)^2$$

Optimization

Statistics

- Set gradient to 0 and find parameter

Optimization

Linear algebra

$$\frac{\partial L(\theta)}{\partial \theta} = -\frac{2}{m} \sum_{i=1}^m (y^i - \theta^\top x^i) x^i = 0$$

$$\Leftrightarrow -\frac{2}{m} \sum_{i=1}^m y^i x^i + \frac{2}{m} \sum_{i=1}^m x^i x^{i\top} \theta = 0$$

Statistics

Statistics

# 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

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

Optimization

# Probabilistic Interpretation of LMS

- Assume  $y$  is a linear in  $x$  plus noise  $\epsilon$

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

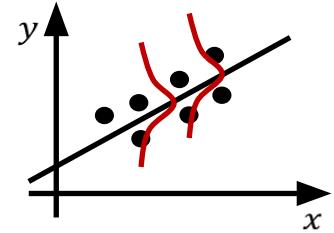
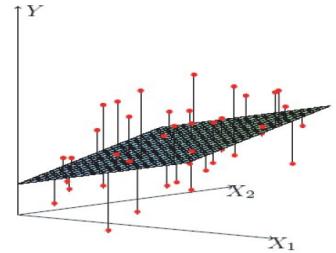
- Assume  $\epsilon$  follows a Gaussian  $N(0, \sigma)$

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

- By independence assumption, likelihood is

$$L(\theta)$$

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



Probability

# Probabilistic Interpretation of LMS, cont.

- Hence the log-likelihood is:

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

- LMS is equivalent to MLE of  $\theta$  !

$$LMS: \frac{1}{m} \sum_i^m (y^i - \theta^\top x^i)^2$$

- How to make it work in real data?

Statistics

Algorithms  
Programming

# Grading

- Homework (30%)
- Project (40%)
- Exam (30%)
- Participation bonus (5%)

# 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 10th**.

# 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](#)
  - 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 5 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 April 28th**

# 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 12 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/CS4641-spring2025/lectures/>

# Q&A