Lecture 1: What is Machine Learning?

ECE 2410 – Introduction to Machine Learning

Farzad Farnoud

University of Virginia

Spring 2026

Today's Agenda

The Goal: Function Approximation

Goal

We want to build a **function** f with a desirable input-output relationship.

Input $x \longrightarrow \boxed{f} \longrightarrow$ **Output** $y = f(x)$

Examples:

- $x =$ email text $\rightarrow f(x) =$ spam or not spam
- $x =$ image pixels $\rightarrow f(x) =$ cat, dog, or bird
- $x =$ house features $\rightarrow f(x) =$ price
- $x =$ patient data $\rightarrow f(x) =$ diagnosis

Two Approaches to Building f

Rule-based algorithms

Human writes the rules

Input x

↓

f written by human based on rules

↓

Output $f(x)$

"If email contains 'free money', mark as spam"

Machine Learning

Learn from solved examples

Examples (x_i, y_i)

↓

Learning

↓

Function $\hat{f}$

Input x

↓

$\hat{f}$

↓

Output $\hat{f}(x)$

"Here are 10,000 emails labeled spam/not spam. Learn the pattern."

ML as Function Approximation

Key Insight

Machine learning is **function approximation** from example input-output pairs.

Given:

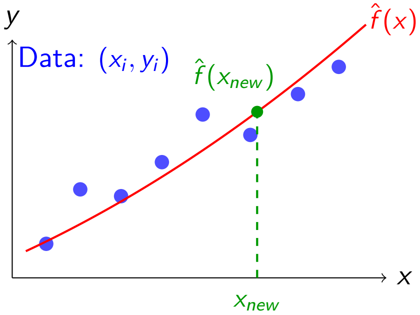
Solved examples (training data)
$\{(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\}$

Goal:

Find a function $\hat{f}$ such that
$\hat{f}(x_i) \approx y_i$ for all examples

Hope:

$\hat{f}$ works well on *new, unseen* inputs too!



Why Machine Learning?

When to use ML instead of writing rules:

- Rules are **too complex** to program manually
- The problem **changes over time** (need adaptation)
- Patterns exist but are **hard to articulate**
- Large amounts of **data are available**

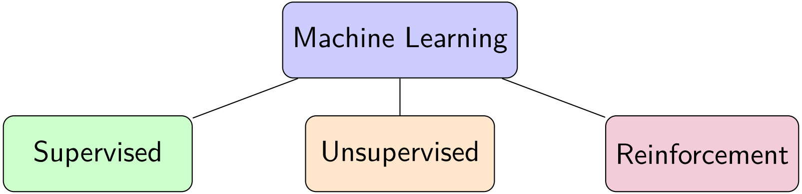
Examples where rules fail:

- Spam filtering — how do you describe *all* spam patterns?
- Image recognition — how do you describe a “cat” in pixels?
- Speech recognition — accents, background noise, vocabulary...
- Medical diagnosis — complex interactions between symptoms

The ML Promise

Give me enough examples, and I’ll figure out the rules myself.

Types of Machine Learning



Supervised
Learn from labeled (solved) examples

Unsupervised
Find patterns in unlabeled data

Reinforcement
Learn by trial and error

Supervised Learning

Key idea: Learn from labeled examples (input-output pairs)

Classification

- Output is a **category**
- Email: spam or not spam?
- Image: cat, dog, or bird?
- Medical: disease or healthy?

Regression

- Output is a **number**
- House price prediction
- Stock price forecasting
- Temperature prediction

This Semester

We'll cover: k-Nearest Neighbors (classification), Linear Regression, Neural Networks

Unsupervised Learning

Key idea: Find hidden patterns in data *without* labels

Clustering

- Group similar items together
- Customer segmentation
- Document organization
- Image compression

Other Examples

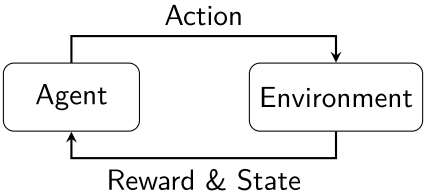
- Dimensionality reduction
- Anomaly detection
- Association rules

This Semester

We'll cover: k-Means Clustering

Reinforcement Learning

Key idea: Learn by interacting with an environment



Examples:

- Game playing (AlphaGo, Chess)
- Robotics (walking, grasping)
- Self-driving cars

Note

We won't cover RL in depth this semester, but it's good to know it exists!

Activity: Predict the Major

The Challenge

Can we predict someone's **major** based on where they sit in the classroom?

Setup:

- **Training set:** Everyone born **on or before May 31**, please stand up (and stay standing), and tell us your major when asked
- **Test set:** Everyone else (June–December birthdays) stays seated

Activity I:

- Pick a seated student (test point)
- Look at the $k = 3$ closest standing students (neighbors)
- Predict: What's the most common major among those neighbors?
- Reveal the true answer — were we right?

Activity II: "Guess" your own major based on where you sit. Did you guess correctly?

ML is Everywhere

Generative AI

- ChatGPT, Claude, Gemini
- Image generation (DALL-E)
- Code assistants (Copilot)

Social Media & Recommendations

- Content feeds (TikTok, Instagram)
- Netflix, Spotify, Amazon
- Ad targeting

Autonomous Systems

- Self-driving cars
- Delivery robots, drones

Vision & Language

- Face recognition
- Medical imaging
- Translation, voice assistants

Research & Development

- Drug discovery
- Protein folding (AlphaFold)
- Climate modeling
- Materials science

What You'll Learn This Semester

Part 1: Classification & Clustering

- k-Nearest Neighbors
- k-Means Clustering
- Evaluation metrics
- Model selection

Part 2: Regression

- Linear Regression
- Least Squares

Part 3: Neural Networks

- Multilayer Perceptrons
- CNNs for images
- Training with gradient descent

Part 4: Generative AI

- RNNs and sequences
- Attention & Transformers
- Large Language Models

Assessment

Homework

- Theoretical (written) and coding problems
- Strengthen and teach new concepts
- **You must be able to explain your solutions**

In-class Activities

- Focus on current lecture material
- Not announced in advance
- May be group or individual

Quizzes

- **When:** Typically on Wednesdays announced in advance
- **Content:** Based on recent HW, lectures
- **Coding in Quizzes:** *Typically*, interpret/debug code, not write from scratch

Exams

- Cumulative, in-class
- 75 minutes each
- Two midterms (no final)

Final Project: Applied ML project with written report (details posted closer to deadline)

Grading

Grading Breakdown

- Homework: 25% (lowest dropped)
- Quizzes/Activities: 20% (lowest dropped)
- Midterms (2): 35%
- Final Project: 20%

Late Work

- Homework: -10% if late (up to 24h)
- No submissions accepted after 24h

Missed Assessments

- Lowest score dropped (covers emergencies)
- Further excuses require justification for **both** the dropped & new absence

Classroom & Communication

Electronic Devices

- **Use for learning:** Notes, coding demos.
- **Avoid distractions:** Please close email, social media, etc.
- *Respect your attention and your neighbors'.*

Collaboration

- Discuss concepts: **Encouraged!**
- Code/Solutions: Must be your own. **Must be able to explain it**

Communication

- **Piazza:** Community Q&A
- **Piazza:** Qs for instruction team (me+TAs)
- **Email:** instructor for private matters. Subject must contain [25F-Intro2ML]

AI Use Policy

Philosophy

AI tools (ChatGPT, Claude, etc.) are powerful assistants. We want you to use them responsibly to learn faster, not to avoid learning.

Allowed

- Explaining concepts
- Debugging syntax errors
- Generating plot code
- "Act as a Teaching Assistant"

Not Allowed

- Generating full solutions
- Copy-pasting without understanding
- Using during exams

See details, including how to initialize an AI assistant, on course website.

Getting Started with HW01

Let's start HW1!

Follow the following steps:

1. **Download:** Go to Canvas and download `HW01_HelloJupyter.ipynb`
2. **Option 1 (Colab):** Upload to Google Drive → Right-click → Open with Google Colab
3. **Option 2 (Local):** Launch Jupyter Lab or Notebook locally → Open file
4. **Initialize AI:** Follow instructions to set up your AI assistant
5. **Start:** Run the tutorial cells and complete the exercises!

Due next Monday before class!

Key Takeaways

1. **Machine Learning** = algorithms that learn from data
2. **Three types:**
   - Supervised (labeled data) → Classification, Regression
   - Unsupervised (no labels) → Clustering
   - Reinforcement (rewards) → Game playing, robotics
3. **This semester:** Classification, Regression, Neural Networks, LLMs

Next Time

Lecture 2: k-Nearest Neighbors (kNN) - our first ML algorithm!

Questions?

See you Wednesday!