Overview

COMS 4721 Spring 2018
What is machine learning?

- **Machine learning**: study of computational mechanisms that “learn” from data to make predictions and decisions.
Application 1: image classification

- Birdwatcher takes photos of birds, organizes by species.

![Indigo bunting](image)

Indigo bunting
Application 1: image classification

- Birdwatcher takes photos of birds, organizes by species.
- **Goal:** automatically recognize bird species in new photos.

![Indigo bunting](image)

Indigo bunting
Application 2: recommender system

- Netflix users watch movies and provide ratings.

(Image credit: Koren, Bell, and Volinsky, 2009.)
Application 2: recommender system

- Netflix users watch movies and provide ratings.
- **Goal**: predict user’s rating of unwatched movie.

(Image credit: Koren, Bell, and Volinsky, 2009.)
Application 2: recommender system

- Netflix users watch movies and provide ratings.
- **Goal**: predict user’s rating of unwatched movie.
- *(Real goal: keep users paying customers.)*

(Image credit: Koren, Bell, and Volinsky, 2009.)
Application 2: recommender system

- Netflix users watch movies and provide ratings.
- **Goal**: predict user’s rating of unwatched movie.
- **(Real goal)**: keep users paying customers.)
- **(Real effect)**: reinforce stereotypes found in the data?)

(Image credit: Koren, Bell, and Volinsky, 2009.)
Linguists provide translations of all English language books into French, sentence-by-sentence.
Application 3: machine translation

- Linguists provide translations of all English language books into French, sentence-by-sentence.
- **Goal**: translate any English sentence into French.
Basic setting: supervised learning

**Training data:** labeled examples

\[ (x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n) \]
Basic setting: supervised learning

**Training data:** labeled examples

$$(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$$

where

- each input $x_i$ is a machine-readable description of an instance (e.g., image, sentence), and
Training data: labeled examples

\[(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\]

where

- each input \(x_i\) is a machine-readable description of an instance (e.g., image, sentence), and

- each corresponding label \(y_i\) is an annotation relevant to the task—typically not easy to automatically obtain.
Basic setting: supervised learning

**Training data:** labeled examples

\[(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)\]

where

- each input \(x_i\) is a machine-readable description of an instance (e.g., image, sentence), and
- each corresponding label \(y_i\) is an annotation relevant to the task—typically not easy to automatically obtain.

**Goal:** learn a function \(\hat{f}\) from labeled examples, that accurately “predicts” the labels of new (previously unseen) inputs.
Learn a function?

The learned function \( \hat{f} \) might look like the following:

1: if age \( \geq 40 \) then
2: if genre = western then
3: return 4.3
4: else if release date > 1998 then
5: return 2.5
6: else
7: 
8: end if
9: else if \cdots then
10: 
11: end if
Learn a function?

The learned function $\hat{f}$ might look like the following:

```
1:  if age $\geq$ 40 then
2:    if genre = western then
3:      return 4.3
4:  else if release date $>$ 1998 then
5:    return 2.5
6:  else
7:    ...
8:  end if
9: else if ... then
10: ...
11: end if
```

Want machine to figure out these if-else clauses from the data, rather than hand-code them ourselves.
Other kinds of functions

"Perceptron" (linear classifier)

\[
\text{if } 0.335 \cdot x_1 + 2.5 \cdot x_2 + \cdots + 6.35 \cdot x_{10} > 4.3 \text{ then return spam}
\]

\[
\text{else return not spam}
\]

"Neural network" (non-linear classifier) . . . and many others!
"Perceptron" (linear classifier)

1: if $0.335 \cdot x_1 + 2.5 \cdot x_2 + \cdots + 6.35 \cdot x_{10^6} > 4.3$ then
2:   return spam
3: else
4:   return not spam
5: end if
Other kinds of functions

“Perceptron” (linear classifier)

1: if $0.335 \cdot x_1 + 2.5 \cdot x_2 + \cdots + 6.35 \cdot x_{10^6} > 4.3$ then
2: return spam
3: else
4: return not spam
5: end if

“Neural network” (non-linear classifier)
Other kinds of functions

“Perceptron” (linear classifier)

1: if $0.335 \cdot x_1 + 2.5 \cdot x_2 + \cdots + 6.35 \cdot x_{10^6} > 4.3$ then
2: return spam
3: else
4: return not spam
5: end if

“Neural network” (non-linear classifier)

... and many others!
In 2016-2017, AlphaGo beats human world champions in Go.
Recent stories: AlphaGo


1. Learn to predict “value” of a Go board.
2. Learn to predict “best action” given current state.
3. Monte Carlo Tree Search.
Recent stories: behavior classification

JAABA: interactive machine learning for automatic annotation of animal behavior

Mayank Kabra, Alice A Robie, Marta Rivera-Alba, Steven Branson & Kristin Branson

Received 16 June 2012 Accepted 06 November 2012 Published online 02 December 2012

High-throughput behavior classification from video via machine learning.
Recent stories: behavior classification

JAABA: interactive machine learning for automatic annotation of animal behavior

Mayank Kabra, Alice A Robie, Marta Rivera-Alba, Steven Branson & Kristin Branson

Received 16 June 2012 | Accepted 06 November 2012 | Published online 02 December 2012

• Adaptively chooses frames to be annotated by user.

• Minimizes human effort while maximizes classifier accuracy.

High-throughput behavior classification from video via machine learning.

(Image credits: Nature Methods)
Recent stories: COMPAS

Software from Northpointe Inc. predicts “risk of recidivism”.

Used in criminal justice system, e.g., in parole hearings.

(Image credit: ProPublica, 2016)
Software from Northpointe Inc. predicts “risk of recidivism”.

Used in criminal justice system, e.g., in parole hearings.

(Image credit: ProPublica, 2016)

ProPublica investigation in 2016 found that

[COMPAS] was particularly likely to falsely flag black defendants as future criminals, wrongly labeling them this way at almost twice the rate as white defendants.
Why is machine learning challenging?

Task: given image, predict if it depicts indigo bunting or not.
Why is machine learning challenging?

**Task**: given image, predict if it depicts indigo bunting or not.

Why is this hard?
Why is machine learning challenging?

**Task**: given image, predict if it depicts indigo bunting or not.

**Why is this hard?**

1. Data sparsity.
Why is machine learning challenging?

Task: given image, predict if it depicts indigo bunting or not.

Why is this hard?

1. Data sparsity.
2. Complex relationship between input and label.
Why is machine learning challenging?

**Task**: given image, predict if it depicts indigo bunting or not.

**Why is this hard?**

1. Data sparsity.
2. Complex relationship between input and label.
3. Too many possible functions!
This course (COMS 4721)

Topics

- Principles of *supervised machine learning*
  - Algorithmic and statistical techniques
  - Some well-weathered algorithms and models
- If time permits:
  - Probabilistic modeling
  - Online decision-making

Course requirements

1. Complete assigned reading before each lecture.
2. Attend lecture.
3. Complete homework assignments (theory & programming).
4. Complete two in-class exams (March 7, April 30).
Prerequisites

Mathematical prerequisites

▶ Linear algebra (e.g., vector spaces, orthogonality, spectral decomposition)
▶ Probability (e.g., conditional probability, independence, random variables)
▶ Multivariate calculus (e.g., limits, Taylor expansion, gradients)
▶ Basic algorithms & data structures (e.g., correctness & efficiency analysis)

Computational prerequisites

▶ Regular access to and ability to program in Python or MATLAB.

MATLAB is available for download for SEAS students:
http://portal.seas.columbia.edu/matlab/

Python (and packages like SciPy & NumPy) are available online for free.
Resources

Course website

http://www.cs.columbia.edu/~djhsu/coms4721-s18/

Course staff

- **Instructor**: Prof. Daniel Hsu
- **Teaching assistants**: Ben Lai, Che Shen, Connor Hargus, John Hyun
  Dong Lee
  (see course website)
- **Office hours, online forum (Piazza)**: see course website
You must read, understand, and abide by the course policies.
http://www.cs.columbia.edu/~djhsu/coms4721-s18/syllabus.html

Some highlights:

- **No late assignments, no make-up exams.**
  Lowest homework score (besides that of “HW0”) will be dropped.

- **Disability services:** Make arrangements for accommodations (e.g., for in-class exams) and other services within first two weeks of class.
  https://health.columbia.edu/disability-services

- **Academic rules of conduct:** Any violation will result in a penalty to be assessed at the instructor’s discretion (e.g., failing grade in the class), even for the first infraction.
First homework assignment ("Homework 0") due Monday.

- **Required**: submit on Gradescope.
- Partly intended to help you "page-in" prerequisites; not supposed to be very time-consuming.
- **If you have difficulty completing the assignment, please re-consider taking this course at this time.**
- (We will also use HW0 score to prioritize waiting list.)
Key takeaways

1. Examples of machine learning problems and why they are challenging.
2. Course information.