CSC 580: Principles of Machine Learning

Chicheng Zhang

Department of Computer Science

THE UNIVERSITY OF ARIZONA®

*slides credit: largely based upon CSC 580 Fall 2021 lecture slides by Kwang-Sung Jun

What is machine learning?

What is machine learning (ML)?

- **Tom Mitchell** established Machine Learning Department at CMU (2006).

Machine Learning, Tom Mitchell, McGraw Hill, 1997.

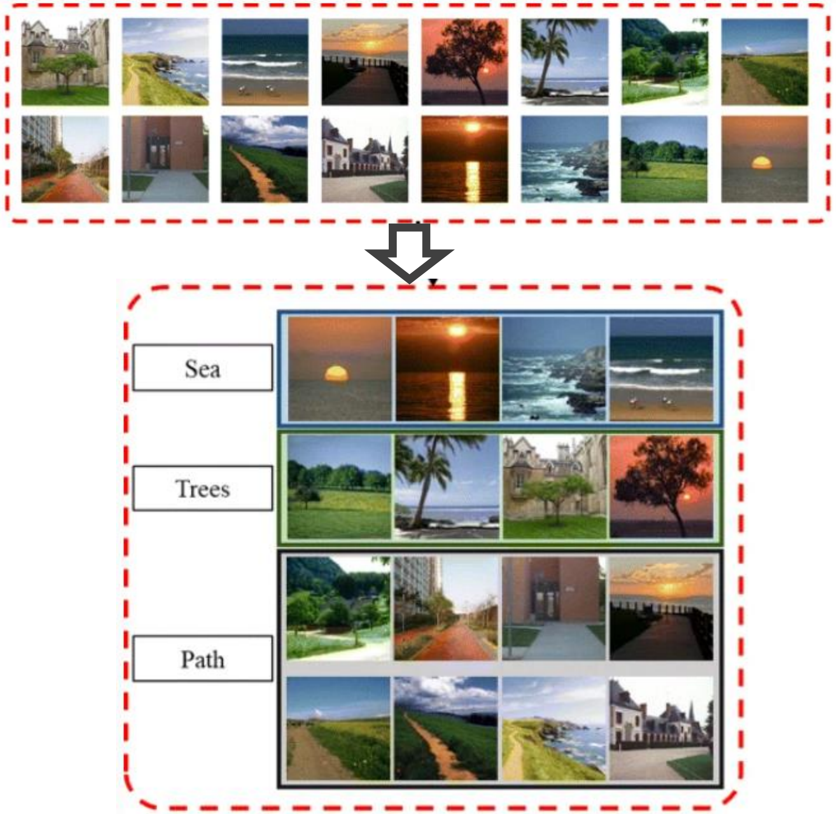
Machine Learning is the study of computer algorithms that improve automatically through experience. Applications range from datamining programs that discover general rules in large data sets, to information filtering systems that automatically learn users' interests.

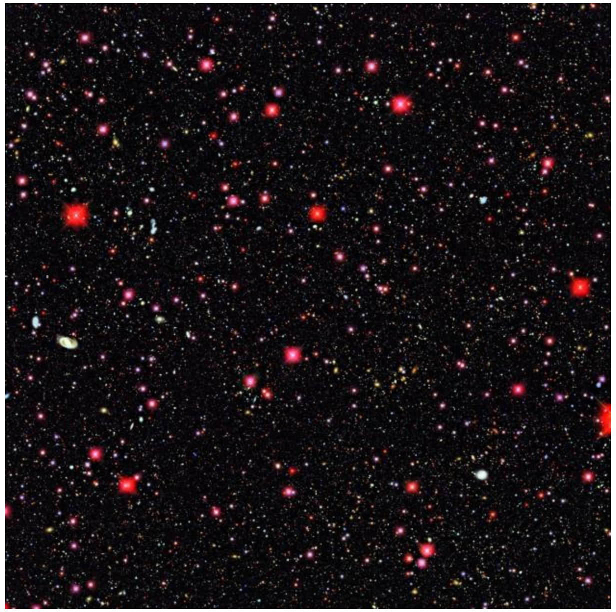
This book provides a single source introduction to the field. It is written for advanced undergraduate and graduate students, and for developers and researchers in the field. No prior background in artificial intelligence or statistics is assumed.

- In short: algorithms adapt to data
- A subfield of **Artificial Intelligence (AI)** – computers perform "intelligent" tasks.
- Classical AI vs ML: rule-driven approaches vs. data-driven approaches

AI Task 1: Image classification

- Predefined categories: C = {sea, trees, path, …}
- Given an image, classify it as one of the set C with the highest accuracy as possible.
- Use: sorting/searching images by category.
- Also: categorize types of stars/events in the Universe (images taken from large surveying telescopes)



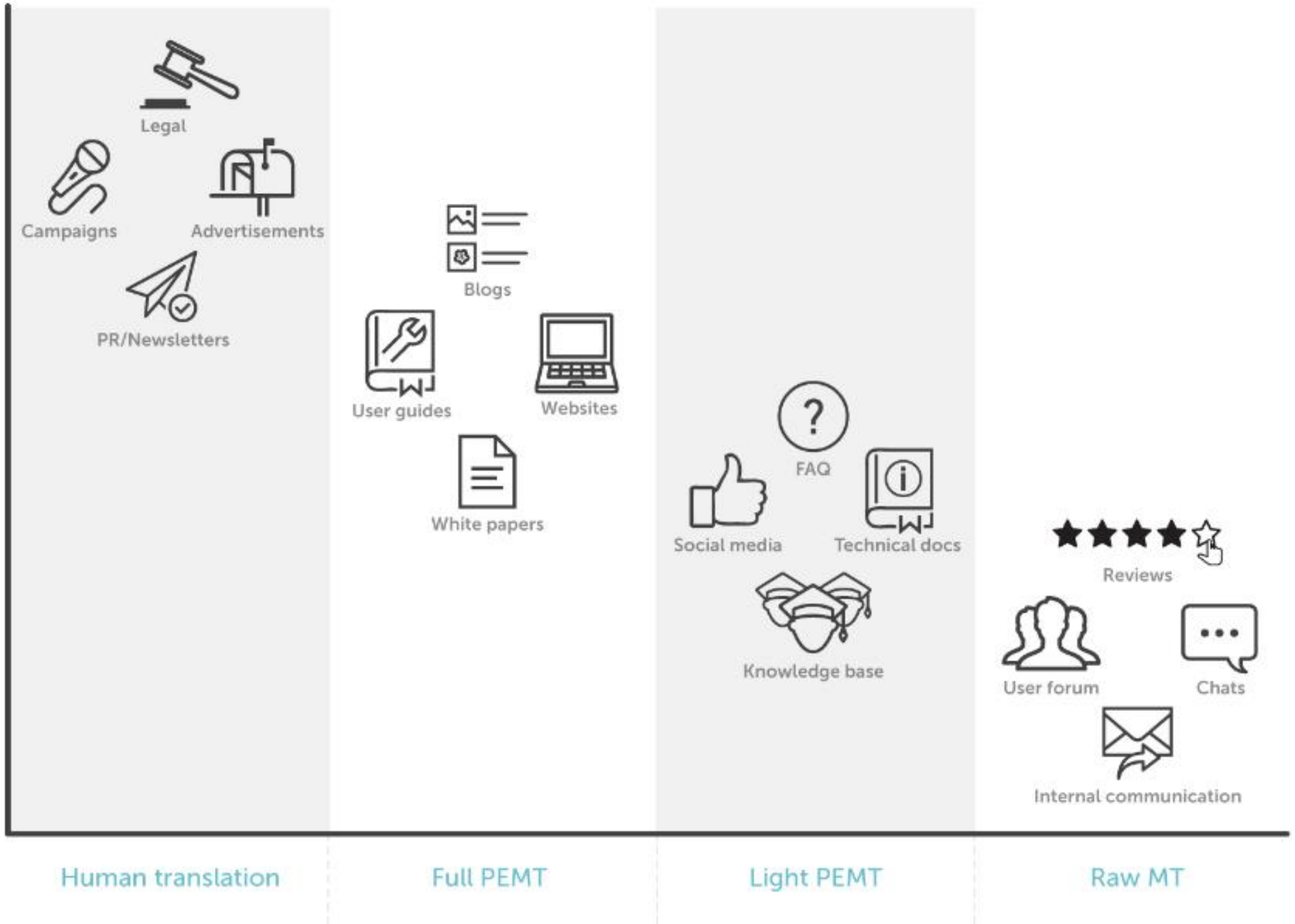


AI Task 2: Recommender systems

- Predict how user would rate a movie
- **Use**: For each user, pick an unwatched movie with the high predicted ratings.
- **Possible approach**: compute user-user similarity or movie-movie similarity, then compute a weighted average.

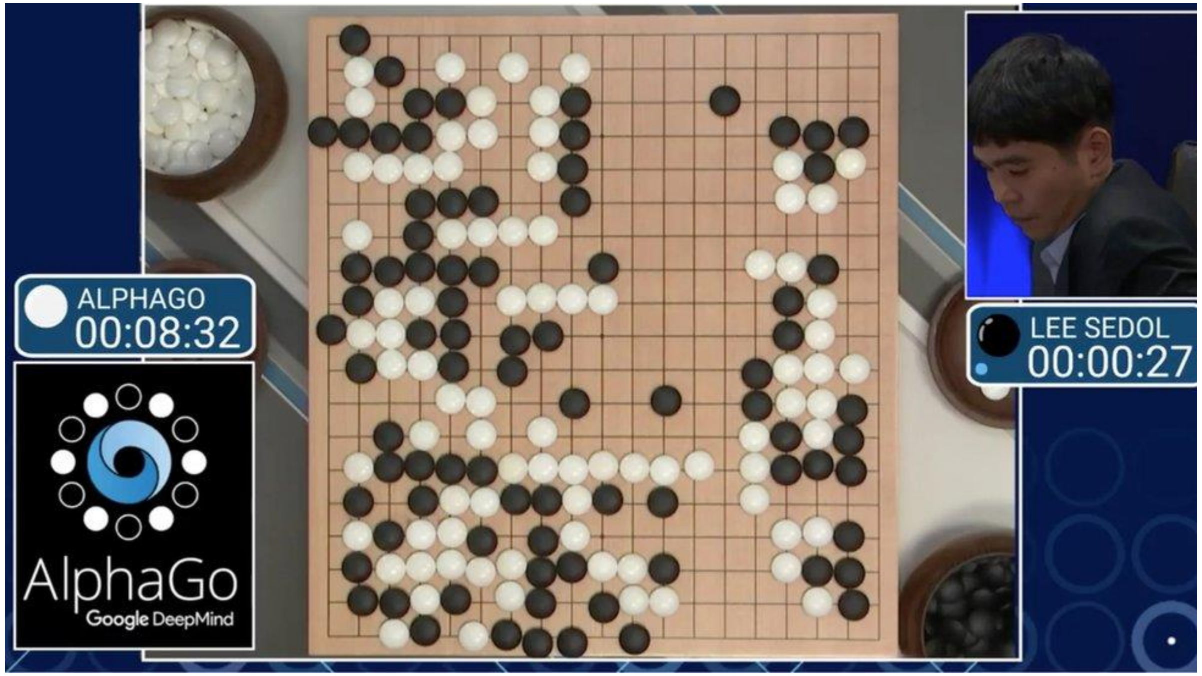
	User 1	User 2	User 3
Movie 1	1	2	1
Movie 2	?	3	1
Movie 3	2	5	2
Movie 4	4	?	5
Movie 5	?	4	2

AI Task 3: Machine translation



AI Task 4: Board games

- Predict win probability of a move in a given game state (e.g., AlphaGo)
- Traditionally considered as a "very smart" task to perform.
- **<u>Use</u>**: Professional go playing, leisure



Hajin Lee
Aug 22, 2020 · 5 min read · Listen

Impact of Go AI on the professional Go world

Traditional AI vs Machine Learning (ML)

- **Traditional AI**: you encode the knowledge (e.g., logical rules), and the machine makes ‘**inference**’, e.g. given ``a -> b and b-> c’’, deduce ``a-> c’’.
  - Example rule: has-feather-texture(object) and has-beak(object) -> is-bird(object).
  - Deductive reasoning

- **ML**: given a number of input and output observations (e.g., animal picture + label), output a **function (can be a set of logical statements or a neural network)** that maps the input to the output accurately.
  - “Big data” setting => better to learn from data than to encode domain knowledge manually.
  - “statistical” / data-driven approach – inductive reasoning

- **Note**: Traditional AI and ML can work synergistically

Interpretability Rules: Jointly Bootstrapping a Neural Relation Extractor with an Explanation Decoder

Zheng Tang, Mihai Surdeanu
Department of Computer Science
University of Arizona, Tucson, Arizona, USA
{zhengtang, msurdeanu}@email.arizona.edu

Work in ML

- The usual CS background is often not sufficient – especially mathematical side, beyond discrete math.

- Applied data scientist
  - Collect/prepare data, build/train models, analyze errors
- ML engineer
  - Implement/fine-tune ML algorithms and infrastructure
- ML researcher
  - Design/analyze models and algorithms
  - Theory: Provide mathematical guarantees. E.g., If I were to achieve 90% accuracy, how many data points do we need? => <u>sample complexity</u>(see e.g. CSC 588)

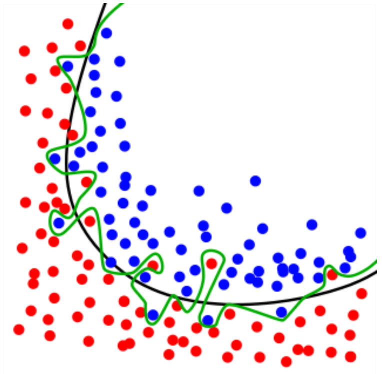
Prerequisites

- Math
  - linear algebra, probability & statistics, multivariate calculus, reading and writing proofs.
  - Q: how many of you are familiar with eigen-decomposition?

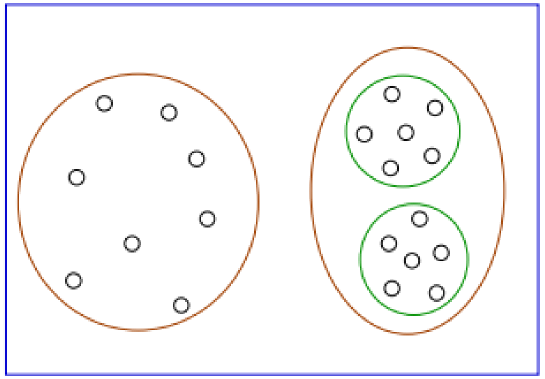
- Software/programming
  - You need be familiar with at least one programming language
  - You need to be fluent at writing functions and using them efficiently.
  - Much ML work is implemented in python with libraries such as numpy and pytorch.

Overview of ML problems

Supervised learning



Unsupervised learning

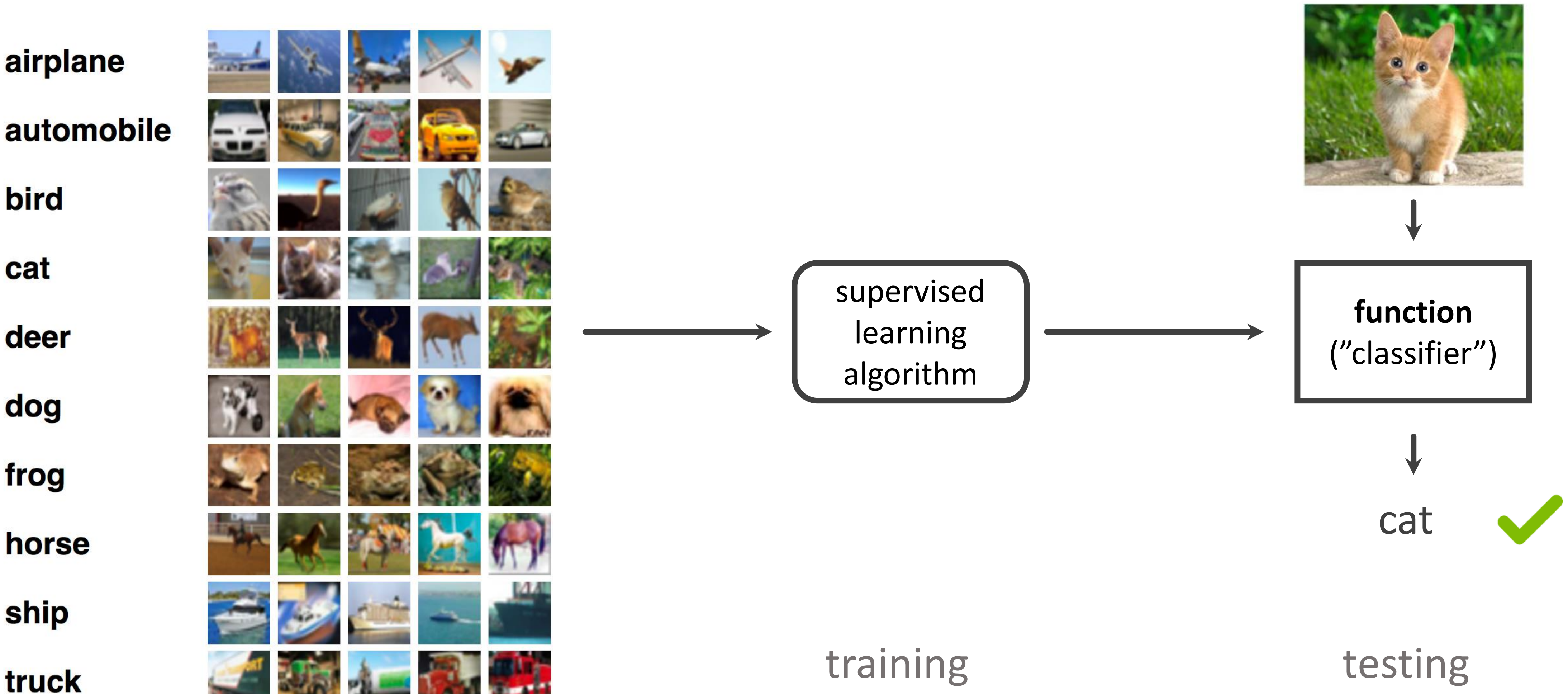


Interactive learning

Supervised Learning

Basic setting: Supervised learning

- Training data: dataset comprised of *labeled examples*: pairs of (feature, label)



Example classifier 1: Decision tree

- Task: predict the rating of a **movie** by a **user**
- If age >= 40 then
  - if genre = western then
    - return 4.3
  - else if release date > 1998 then
    - return 2.5
  - else ..

    …

    end if
- else if age < 40 then

  …
- end if

Example classifier 2: Linear

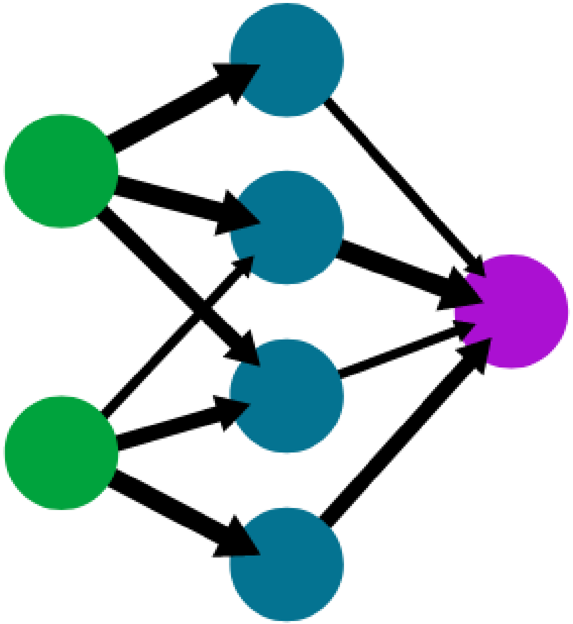
- E.g., Image classification
- Let x be a set of pixel values of a picture (30x30 resolution) => 900 dimensional vector x.
- If $\boxed{0.124 \cdot x_1 - 2.5 \cdot x_2 + \cdots + 2.31 \cdot x_{900}} > 2.12$ then
  - return cat
- else
  - return dog
- end

"linear combination"

- Coefficients: signed "importance weights"

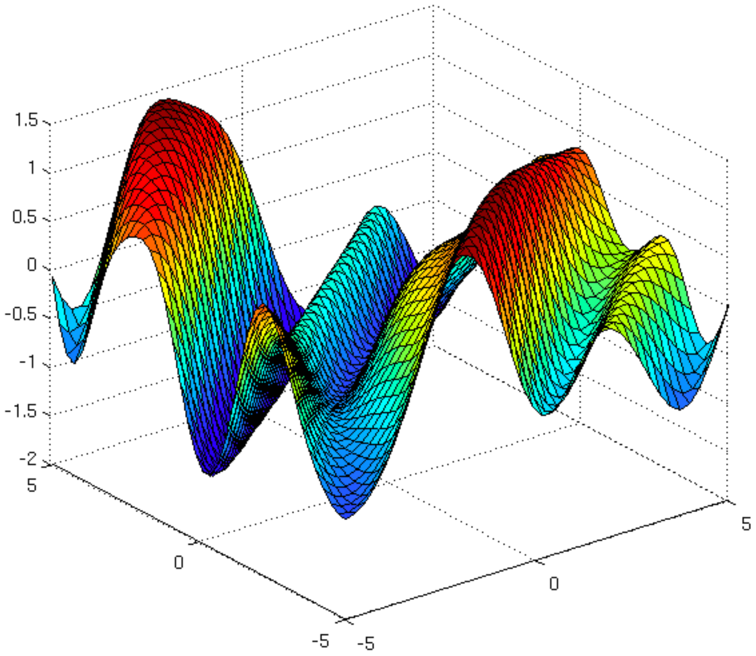
Example function 3: Nonlinear

Neural network



(stacked **linear** models with nonlinear **activation functions**)

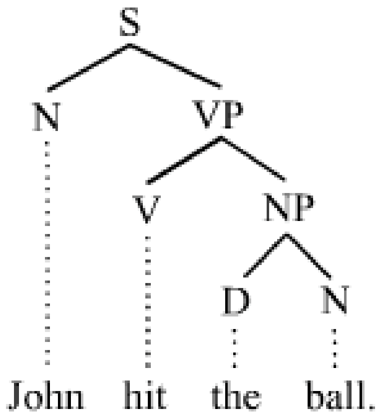
Kernel classifiers



(**linear** in the induced feature space)

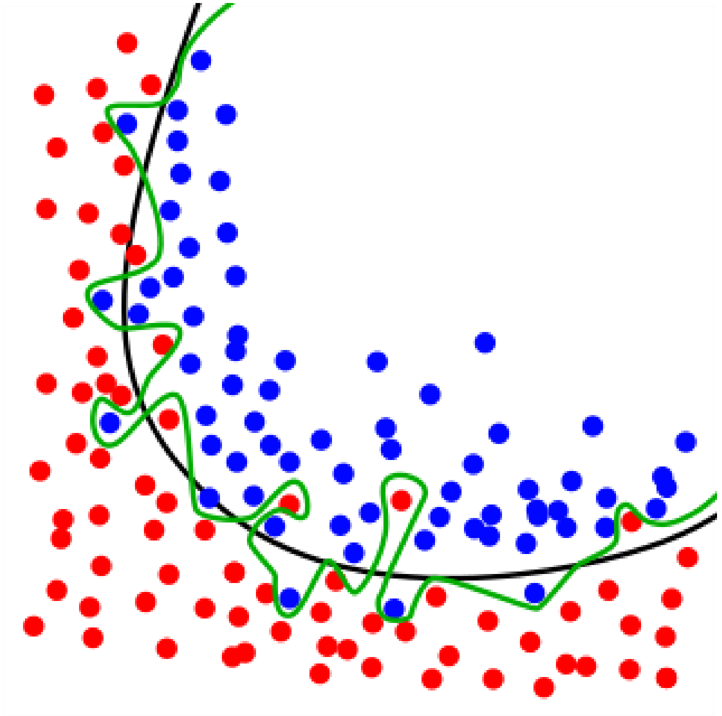
Supervised learning: Types of prediction problems

- Binary classification
  - Given an email, is it spam or not?
- Multi-class classification
  - Image classification with 1000 categories.
- Regression: the label is real-valued (e.g., price)
  - Say I am going to visit Italy next month. Given the air ticket price trends in the past, what would be the price given (the # of days before the departure, day of week)?
- Structured output prediction: more than just a number
  - Given a sentence, what is its grammatical parse tree?



The challenge: How to learn a function

- Why not learn the most complex function that can work flawlessly for the training data and be done with it? (i.e., classifies every data point correctly)
- Extreme: memorization.
  - For a test example, if it exactly matches some training example, output the corresponding label
  - Otherwise, output some default label, say blue
- It does not work.
- Need to learn from training dataset, but don't "over-do" it.
- This is called "generalization" – an important notion.

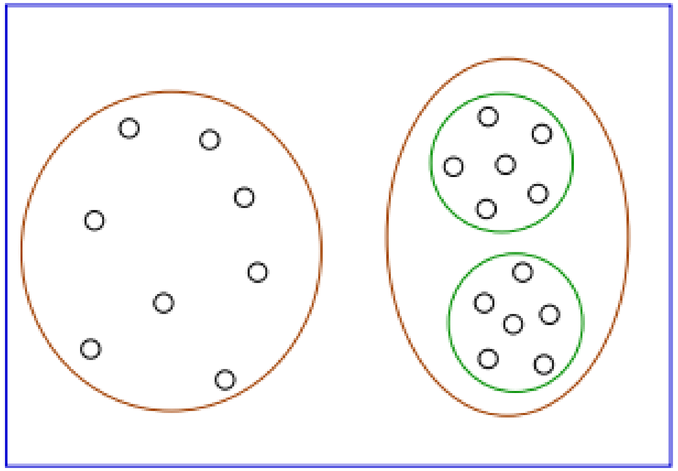


green: memorization
black: optimal decision boundary

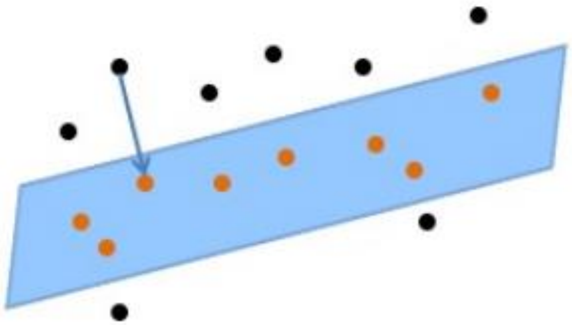
Unsupervised learning

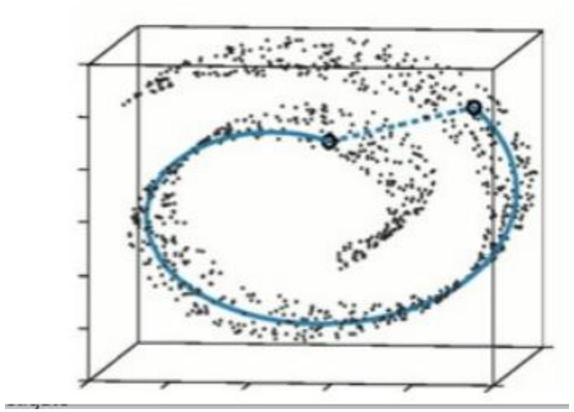
- Finding structures in data, e.g.

- Clustering



- Dimensionality reduction



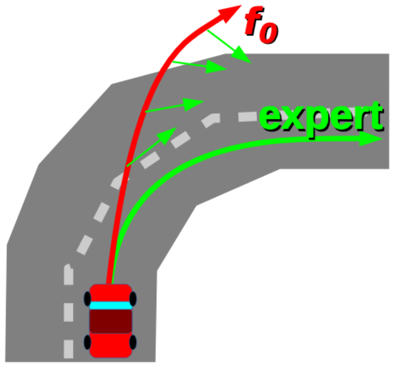


Interactive learning

- Algorithms collect and analyze data to make decisions

- E.g.:
- Reinforcement learning

- Imitation learning



What to expect in the class

- ~~How to use sklearn, pytorch, tensorflow, fine-tuning deep net algorithms.~~
    - You are encouraged to learn these on your own

- Algorithmic and statistical <u>principles</u>
    - Well-studied models and methods.
    - Those that give you some "understanding".
    - These are and will be referred/extended/revisited in the future.

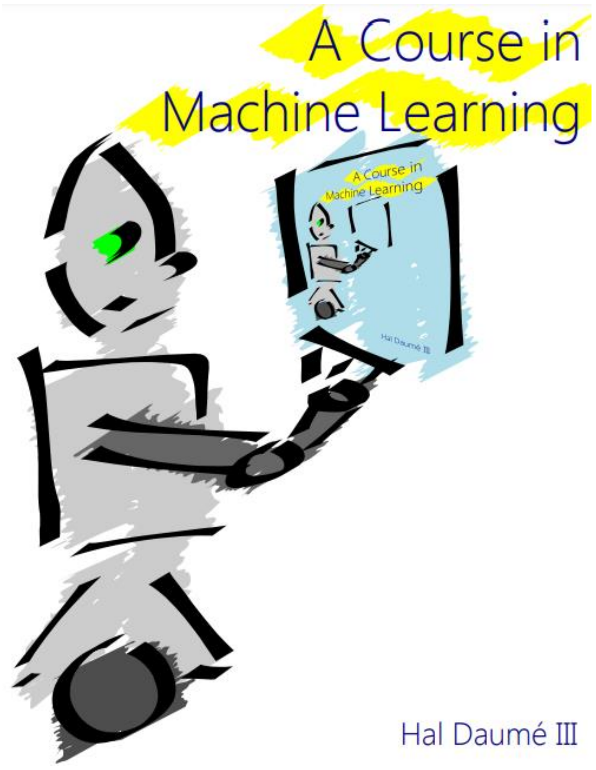
- Programming and proofs
    - No need to be a guru.
    - But you must be familiar enough to (1) follow popular codes and proofs and (2) be able to adapt yourself to new programming tools and proofs in the future.

Logistics

Course structure

- D2L: for important email communications
- Course website: https://zcc1307.github.io/courses/csc580fa22
- Piazza: mainly for Q&A/discussion.
- Gradescope: submitting homeworks

- Book: "A Course in Machine Learning" (CIML) by Hal Daume III
- http://ciml.info/



Syllabus summary

- Warm up
  - Basic supervised learning: decision tree, k-NN, perceptron
  - Practical issues in supervised learning: evaluation, feature selection, etc.
  - Bias-variance decomposition
- Learning methods
  - linear models, kernels
  - naïve Bayes, graphical models (cf. CSC 696H by Prof. Jason Pacheco, Math 577-001 by Prof. Misha Chertkov)
  - neural networks (cf. ISTA 457/INFO 557 by Prof. Steve Bethard)
- Other training methods: ensemble, stochastic gradient descent
- Other paradigms: unsupervised learning, reinforcement learning
- Learning theory
- Also complementary to Math 574M (by Prof. Helen Zhang)

Syllabus summary

- 08/22: HW0 (calibration) assigned
- 08/31: HW1 assigned
- 09/19: HW2 assigned
- 10/12: Midterm exam (at the class meeting time)
- 10/19: Project proposal due
- 11/02: HW3 assigned
- 11/16: HW4 assigned
- 12/13: Final exam at 3:30pm - 5:30pm
- 12/15: Final project report due
- **Due**: HW0 is due in 7 days. HWs 1-4 are due in 10 days.
- NO LATE DAYS

- The instructing staff will assign grades on a scale from 0 to 100, with the following weights:
  - Homework assignments: 40%
  - Project: 20%
  - Midterm exam: 15%
  - Final exam: 15%
  - Participation: 10%

- Project:
  - pick a paper in recent ML venue and implement it
  - pioneering new applications of ML (e.g., connect to your research)
  - talk to me for other ideas.

Office hours

- Tuesdays 4-5pm, GS 720

- Or by appointment

Participation

- You are expected to do **assigned readings**, and participate in in-class discussions

- Stop me at any point to ask questions.

- Any ideas to encourage participation?

- I will also find ways to encourage off-class discussion in Piazza.

Academic integrity

- Case study 1
  - Two students with the same department with the same cultural background.
  - 5-6 lines out of 100 lines exactly the same (python code).

- Case study 2
  - Final project was planned to be something about reinforcement learning
  - However, the submitted final project was something completely different.
  - It says ‘in this thesis, …’ in the abstract. => turns out, the student copy-pasted his/her master’s thesis.

- So, what happened to them?
- **No tolerance**. You will get an F.

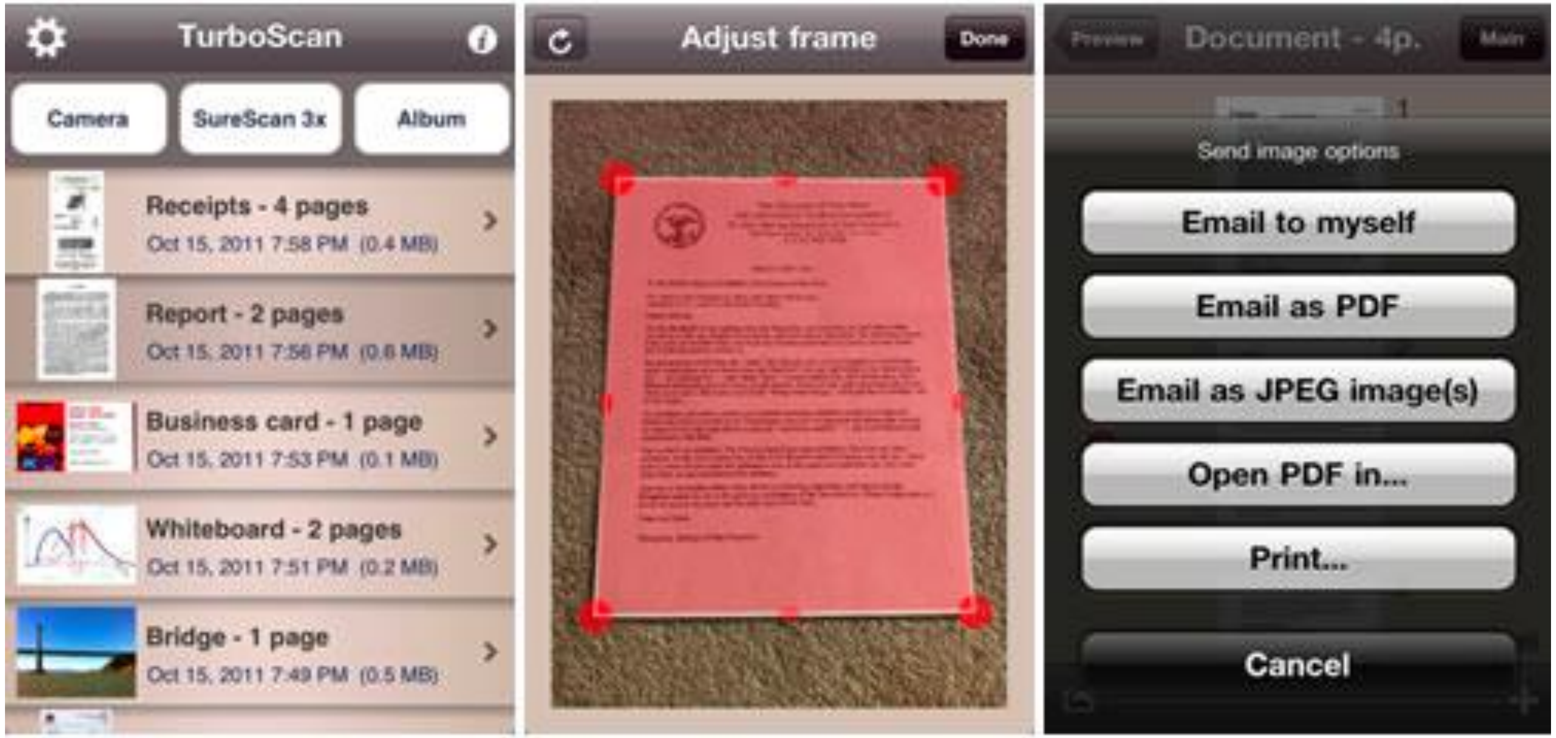
HW0

- Calibration purpose; due on **8/29 5pm**. NO LATE DAYS. Will not accept late submissions.
- Will not be part of the homework score.
- I require that you spend some time to figure out an answer to the homework.
- If you failed to figure out, please explain **what you have done to find an answer** and **where you get stuck**.
  - DON'T: ”I googled it and nothing came up”
  - DO: “I read material A, and there is this statement B that seems to help, but when I tried to apply, C became an issue due to independence. ..."

- The participation score will be deducted (-2 out of 10pts) if …
  - Empty answers
  - No nontrivial efforts to solve it.

HW0 Submission: Gradescope

- Watch the video and follow the instruction: https://youtu.be/KMPoby5g_nE
- Please upload one PDF file.
- If you do it handwritten, then make sure you picture it well. I recommend using TurboScan (smartphone app) or similar ones to avoid looking like slanted or showing the background.



Background refreshers

Probability

- http://cs229.stanford.edu/section/cs229-prob.pdf
- Lecture notes: http://www.cs.cmu.edu/~aarti/Class/10701/recitation/prob_review.pdf

Linear Algebra:

- http://cs229.stanford.edu/section/cs229-linalg.pdf
- Short video lectures by Prof. Zico Kolter: http://www.cs.cmu.edu/~zkolter/course/linalg/outline.html
- Handout associated with above video: http://www.cs.cmu.edu/~zkolter/course/linalg/linalg_notes.pdf

Big-O notation:

- http://www.stat.cmu.edu/~cshalizi/uADA/13/lectures/app-b.pdf
- http://www.cs.cmu.edu/~avrim/451f13/recitation/rec0828.pdf

Other resources:

- The matrix cookbook: https://www.math.uwaterloo.ca/~hwolkowi/matrixcookbook.pdf
- The probability and statistics cookbook: http://statistics.zone/
- Calculus cheatsheet: https://tutorial.math.lamar.edu/pdf/calculus_cheat_sheet_all.pdf

Next lecture (8/24)

- The supervised learning paradigm

- Decision-tree learning

- Assigned reading: CIML Chap. 1 (Decision Trees)

Thank you!

Questions?