A classical problem in algorithm design: **Shortest Path in Graphs**

**Input:** A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

**Output:** A path with minimum cost from source to destination
Three important problem-solving pillars:

- **Data models**, the abstractions used to describe problems (graphs, trees, lists, sequences, algebra, logic, etc)

- **Data structures**, the programming-language constructs used to represent data models (arrays, stacks, queues, AVL trees, etc)

- **Algorithms**, the techniques used to obtain solutions by manipulating data as represented by the abstractions of a data model, by data structures, or by other means (dynamic programming, greedy algorithms, divide & conquer, etc)

Dijkstra's algorithm
Traditional computer science

We are given a **well defined** description of a problem to map a given input to its corresponding output!

Input-output relations are fixed and **well defined**!

E.g., sorting, shortest path, spanning trees, scheduling, etc.

The goal is to write a computer program that efficiently generates an output for a given input.
Algorithmic problem solving

A toy problem in algorithm design: is a given list sorted?

**Input**: a sequence of number

**Output**: YES or NO

3 4 1 2
Algorithmic problem solving

A toy problem in algorithm design: is a given list sorted?

**Input**: a sequence of number

**Output**: YES or NO

Program

YES or NO
Algorithmic problem solving

A toy problem in algorithm design: **is a given list sorted?**

**Input:** a sequence of numbers

**Output:** YES or NO

3 4 1 2

Program

**Well defined** and **known** function!

One-to-one mapping from inputs to outputs!
Let’s play a numbers game

Guess my model/rule/function:

I have a specific function about a set of **TWO numbers**.

You can ask me questions about it only in the following term:

  **Guess two consecutive numbers**

  I will tell you if your guess fits my model (follows the rule)

I can only say YES/NO
Let’s play a numbers game

**Input**: two numbers

**Output**: YES or NO

Depending on the complexity of function, we can make queries, gather samples, and guess the unknown but fixed function!

Well defined BUT unknown function!

One-to-one mapping from inputs to outputs!
Can I eat this mushroom?

- I do not know what type it is?
- I have never seen it before?
- Is it edible or poisonous?

Program

YES or NO
How about now?

Similar to numbers game, can we gather samples from function and guess the function that maps mushrooms to YES (edible) or NO (poisonous)?

- **edible**

![Edible Mushrooms](image1.jpg)

- **poisonous**

![Poisonous Mushrooms](image2.jpg)

Suppose we are given examples of edible and poisonous mushrooms.
Can I eat this mushroom?

Mushroom poisoning epidemiology in the United States

William E. Brandenburg\textsuperscript{a,b} and Karlee J. Ward\textsuperscript{c}

\textsuperscript{a}Family Medicine Residency of Idaho, RTT Caldwell, 777 N. Raymond Street, Boise, Idaho 83704-9251; \textsuperscript{b}West Valley Medical Center, 1717 Arlington Avenue, Caldwell, Idaho 83605; \textsuperscript{c}Pediatric Intensive Care Unit, Saint Luke’s Hospital, 190 E Bannock Street, Boise, Idaho 83712

**ABSTRACT**

Ingestion of wild and potentially toxic mushrooms is common in the United States and many other parts of the world. US poison centers have been logging cases of mushroom exposure in The National Poison Data System (NPDS) annual publications for over 30 years. This study compiles and analyzes US mushroom exposures as reported by the NPDS from 1999 to 2016. Over the last 18 years, 133,700 cases (7428/\textperthousand/year) of mushroom exposure, mostly by ingestion, have been reported. Cases are most frequently unintentional (83\%, $P < 0.001$); cause no or only minor harm (86\%, $P < 0.001$); and in children <6 years old (62\%, $P < 0.001$). Approximately 704 (39/year) exposures have resulted in major harm. Fifty-two (2.9/year) fatalities have been reported, mostly from cyclopeptide (68--89\%)--producing mushrooms ingested by older adults unintentionally. The vast majority of reported ingestions resulted in no or minor harm, although some groups of mushroom toxins or irritants, such as cyclopeptides, ibotenic acid, and monomethylhydrazine, have been deadly. Misidentification of edible mushroom species appears to be the most common cause and may be preventable through education.

**ARTICLE HISTORY**

Received 7 February 2018  
Accepted 18 May 2018

**KEYWORDS**

Cyclopeptide;  
monomethylhydrazine;  
mycotoxin; National Poison Data System; psilocybin

**Ill-defined and unknown function!**

There is no universal rule that maps inputs to outputs. Even worse, the outputs might be noisy and not accurate :-(

Program

YES or NO
What we can do with data: memorization

If you find a mushroom, just search for it in your database.

If you do NOT have it in your database, just say I DO NOT KNOW, sorry!
What we can do with data: **data analytics**

**Data:** What we know about the world

**Data analytics:** the sciences and technology (statistical and computational) for discovering patterns from authorized data/information (often massive, ill-structured) to answer questions or verify a hypothesis!

- Which state reported most death?
- Death among younger is higher!

---

Physical sciences  
Biological sciences  
Social sciences  
Advertisement

---

**THE SCIENTIFIC METHOD**

1. **Question**
2. **Hypothesis**
3. **Experiment**
4. **Result**
5. **Observation**
6. **Conclusion**
What we can do with data: data analytics

Datafication refers to the collective tools, technologies and processes used to transform an organization to a data-driven enterprise using data.

“If you torture the data long enough, it will confess.”

— Ronald H. Coase
What we can do with data: predicting future

edible

poisonous

Instead of searching, can I build a smart algorithm that can PREDICT?

I hate saying sorry :-)

16
Learning

Can we write a program to **PREDICT** other (probably not seen) mushrooms?

- edible

- poisonous

**Input/output relationship is difficult to specify.**

**A pattern exists.** We don’t know it. We have data to **discover** it.
What is “Machine Learning”?

“Learning” (in nature): Using past experience to make future decisions or guide future actions

“Machine Learning” (an engineering paradigm): Using data and examples, instead of expert knowledge, to automatically create systems that perform complex tasks (e.g., make predictions or decisions)
What is “Machine Learning”?

A ML algorithm is a computer program that generates another computer program that captures knowledge in data and can be used for future prediction:

- **Data** (for training)
- **Learning Algorithm** (a computer program)
- **Knowledge/Model** (another computer program)

Input/output pairs

**Training/Learning**
Data driven problem solving

“Machine Learning” is a data-driven problem solving strategy!

We are given a bunch of inputs and corresponding outputs!

Little knowledge on how the mapping works (sometimes even impossible to model the exact mapping)
The ultimate goal of machine learning is **generalization NOT memorization**!

Making future prediction on unseen examples as accurate as possible!
A self-driving car system uses dozens of components that include detection of cars, pedestrians, and other objects.

One way to build a detection system is to write down rules:

```python
# pseudocode example for a rule-based classification system
object = camera.get_object()
if object.has_wheels():  # does the object have wheels?
    if len(object.wheels) == 4: return "Car"  # four wheels => car
    elif len(object.wheels) == 2:
        if object.seen_from_back():
            return "Car"  # viewed from back, car has 2 wheels
        else:
            return "Bicycle"  # normally, 2 wheels => bicycle
    return "Unknown"  # no wheels? we don't know what it is
```

In practice, it's almost impossible for a human to specify all the edge cases.
Self-driving cars: machine learning

The machine learning approach is to teach a computer how to do detection by showing it many examples of different objects.

No manual programming is needed: the computer learns what defines a pedestrian or a car on its own!

Machine learning is a field of study that gives computers the ability to learn without being explicitly programmed. (Arthur Samuel, 1959.)

Data

Learning Algorithm
(a computer program)

Knowledge/Model
(another computer program)

Input/output pairs
(videos of good drivers)

Training

Inference
More data, less expert knowledge

Expert knowledge → More Data → Machine Learning

Use data to fit a specific model

Expert Systems
(no data at all)
(modeling the knowledge of domain experts)

No Free Lunch
The ultimate goal of machine learning is **generalization NOT memorization**!
Making future prediction on unseen examples as accurate as possible!

**Training**

- Slides
- Homeworks
- Examples
- ...

**Inference (Testing)**

- Final Exam
What is NOT machine learning?

- **Memorizing** a database
  
  Dog versus cat example (memorize all input images and do image search!)

- **Sorting** a database

- Doing your taxes

- Playing tic-tac-tao

- No “learning” involved
  
  - We know how to do
  
  - We can program a computer to do it
"Machine Learning" is a data-driven problem solving strategy!

We’re interested in the **knowledge/information** rather than the data!

We are drowning in data and starving for information. 

*Rutherford Roger*
Modeling knowledge

A simple model you learned in STAT courses!

Data (history)

1 0 1 1 0 1 1 0 1 1

Inference (future)

Can you make a guess via memorization or data analytics?
Modeling knowledge

A simple model you learned in STAT courses!

Use maximum likelihood estimation (MLE) to estimate the probability of head $p = 0.7$

Data (history)

Inference (future)

We can distill a large number of 0 and 1s into a single number (knowledge) and make prediction! We model knowledge as a probability distribution.
Knowledge in brain

Neuroscientists have long believed that learning and memory formation are made by the strengthening and weakening of connections among brain cells.

Approximately 86 billion neurons in the human brain.

- Fruit fly: 100 thousand neurons
- Mouse: 75 million neurons
- Cat: 250 million neurons
- Chimpanzee: 7 billion neurons
- Elephant: 257 billion neurons
The neurons are connected to one another with the use of **axons** and **dendrites**, and the connecting regions between axons and dendrites are referred to as **synapses**.

Each input to a neuron is **scaled with a weight**, which affects the function computed at that unit. The neuron is **fired (activated)** when the input is beyond a threshold.

An artificial neural network computes a function of the inputs by **propagating the computed values from the input neurons to the output neuron(s)** and using the **weights as intermediate parameters**.

The **training data** provides feedback to the **correctness of the weights** in the neural network depending on how well the predicted output (e.g., probability of credit approval) for a particular input matches the annotated output label in the training data.

Learning occurs by changing the weights connecting the neurons.
A single neuron can be modeled as a function (computation unit) with inputs and outputs!

\[
\sum_{i=1}^{d} w_i x_i + b
\]
Mathematical model of brain

Input

Knowledge Model

Brain

Output

edible or poisonous

Lifelong learning of weights of neurons with the help of sleeping

Artificial Neural Network

edible or poisonous

Machine learning algorithms distill knowledge into weights of neurons automatically!
Why it could be hard?

Please stay tuned! Will elaborate more on this.
“Machine Learning” is a data-driven problem solving strategy!

- Using salary, debt, years in residence, etc., approve for credit or not!

- No magic credit approval formula.

- Banks have lots of data, whether or not they defaulted on their credit, and customer information such as salary, debt, etc.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>32</td>
</tr>
<tr>
<td>Gender</td>
<td>Male</td>
</tr>
<tr>
<td>Salary</td>
<td>40,000</td>
</tr>
<tr>
<td>Debt</td>
<td>26,000</td>
</tr>
<tr>
<td>Years in job</td>
<td>2</td>
</tr>
<tr>
<td>Years at home</td>
<td>3</td>
</tr>
</tbody>
</table>

A pattern exists. We don’t know it. We have data to learn it.
### Netflix problem

The Netflix challenge: improve our accuracy by 10% and win $1M

<p>| | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="Image" /></td>
<td><img src="image2.png" alt="Image" /></td>
<td><img src="image3.png" alt="Image" /></td>
<td><img src="image4.png" alt="Image" /></td>
<td><img src="image5.png" alt="Image" /></td>
<td><img src="image6.png" alt="Image" /></td>
<td><img src="image7.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image8.png" alt="Image" /></td>
<td><img src="image9.png" alt="Image" /></td>
<td><img src="image10.png" alt="Image" /></td>
<td><img src="image11.png" alt="Image" /></td>
<td><img src="image12.png" alt="Image" /></td>
<td><img src="image13.png" alt="Image" /></td>
<td><img src="image14.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image15.png" alt="Image" /></td>
<td><img src="image16.png" alt="Image" /></td>
<td><img src="image17.png" alt="Image" /></td>
<td><img src="image18.png" alt="Image" /></td>
<td><img src="image19.png" alt="Image" /></td>
<td><img src="image20.png" alt="Image" /></td>
<td><img src="image21.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image22.png" alt="Image" /></td>
<td><img src="image23.png" alt="Image" /></td>
<td><img src="image24.png" alt="Image" /></td>
<td><img src="image25.png" alt="Image" /></td>
<td><img src="image26.png" alt="Image" /></td>
<td><img src="image27.png" alt="Image" /></td>
<td><img src="image28.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image29.png" alt="Image" /></td>
<td><img src="image30.png" alt="Image" /></td>
<td><img src="image31.png" alt="Image" /></td>
<td><img src="image32.png" alt="Image" /></td>
<td><img src="image33.png" alt="Image" /></td>
<td><img src="image34.png" alt="Image" /></td>
<td><img src="image35.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image36.png" alt="Image" /></td>
<td><img src="image37.png" alt="Image" /></td>
<td><img src="image38.png" alt="Image" /></td>
<td><img src="image39.png" alt="Image" /></td>
<td><img src="image40.png" alt="Image" /></td>
<td><img src="image41.png" alt="Image" /></td>
<td><img src="image42.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image43.png" alt="Image" /></td>
<td><img src="image44.png" alt="Image" /></td>
<td><img src="image45.png" alt="Image" /></td>
<td><img src="image46.png" alt="Image" /></td>
<td><img src="image47.png" alt="Image" /></td>
<td><img src="image48.png" alt="Image" /></td>
<td><img src="image49.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image50.png" alt="Image" /></td>
<td><img src="image51.png" alt="Image" /></td>
<td><img src="image52.png" alt="Image" /></td>
<td><img src="image53.png" alt="Image" /></td>
<td><img src="image54.png" alt="Image" /></td>
<td><img src="image55.png" alt="Image" /></td>
<td><img src="image56.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image57.png" alt="Image" /></td>
<td><img src="image58.png" alt="Image" /></td>
<td><img src="image59.png" alt="Image" /></td>
<td><img src="image60.png" alt="Image" /></td>
<td><img src="image61.png" alt="Image" /></td>
<td><img src="image62.png" alt="Image" /></td>
<td><img src="image63.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image64.png" alt="Image" /></td>
<td><img src="image65.png" alt="Image" /></td>
<td><img src="image66.png" alt="Image" /></td>
<td><img src="image67.png" alt="Image" /></td>
<td><img src="image68.png" alt="Image" /></td>
<td><img src="image69.png" alt="Image" /></td>
<td><img src="image70.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image71.png" alt="Image" /></td>
<td><img src="image72.png" alt="Image" /></td>
<td><img src="image73.png" alt="Image" /></td>
<td><img src="image74.png" alt="Image" /></td>
<td><img src="image75.png" alt="Image" /></td>
<td><img src="image76.png" alt="Image" /></td>
<td><img src="image77.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image78.png" alt="Image" /></td>
<td><img src="image79.png" alt="Image" /></td>
<td><img src="image80.png" alt="Image" /></td>
<td><img src="image81.png" alt="Image" /></td>
<td><img src="image82.png" alt="Image" /></td>
<td><img src="image83.png" alt="Image" /></td>
<td><img src="image84.png" alt="Image" /></td>
</tr>
<tr>
<td><img src="image85.png" alt="Image" /></td>
<td><img src="image86.png" alt="Image" /></td>
<td><img src="image87.png" alt="Image" /></td>
<td><img src="image88.png" alt="Image" /></td>
<td><img src="image89.png" alt="Image" /></td>
<td><img src="image90.png" alt="Image" /></td>
<td><img src="image91.png" alt="Image" /></td>
</tr>
</tbody>
</table>

- No magic rating formula
- But, we have a lot of ratings and reviews issued by users!
- Can we learn from available data to learn the taste of future users and unseen movies for existing users?
Artwork personalization
A combination of deep learning, natural language processing, and computer vision enables Amazon to hone in on the right amount of packaging for each product.
Can we learn from the history of home prices to zestimate the future prices?

- No formula!

- But history of prices plus characteristics of properties over the years!
- How can you rank millions of documents to show top 10 to a user?

- We have the minions of feedback from users who issued the same or similar query (search engine logs)
- Cancer detection
- Drug discovery
- Protein Folding

Proteins are essential to life, supporting practically all its functions. They are large complex molecules, made up of chains of amino acids, and what a protein does largely depends on its unique 3D structure. Figuring out what shapes proteins fold into is known as the “protein folding problem”, and has stood as a grand challenge in biology for the past 50 years. In a major scientific advance, the latest version of our AI system AlphaFold has been recognised as a solution to this grand challenge by the organisers of the biennial Critical Assessment of protein Structure Prediction (CASP). This breakthrough demonstrates the impact AI can have on scientific discovery and its potential to dramatically accelerate progress in some of the most fundamental fields that explain and shape our world.

A protein’s shape is closely linked with its function, and the ability to predict this structure unlocks a greater understanding of what it does and how it works. Many of the world’s greatest challenges, like developing treatments for diseases or finding enzymes that break down industrial waste, are fundamentally tied to proteins and the role they play.

Drug discovery
Autonomous driving
Metaverse and retail
Machine learning is everywhere?

Machine learning is starting to take over decision-making in many aspects of our life, including:

(a) Keeping us safe on our daily commute in self-driving cars,
(b) Making accurate diagnoses based on our symptoms and medical history,
(c) Pricing and trading complex securities, and
(d) Discovering new science, such as the genetic basis for various diseases.
Ingredients

Smart algorithms

Data

High-performance computing
Supervised learning is task of learning a model (function) that maps an input to an output based on example input-output pairs (labeled training data).

We need labeled data (sometimes easy to gather, sometimes we have to pay)
We are **collecting and storing** data at an unprecedented rate!

Where does data come from:

- YouTube, Facebook, MOOCs, news sites.
- Credit cards transactions and Amazon purchases.
- Transportation data (Google Maps, Waze, Uber)
- Gene expression data and protein interaction assays.
- Maps and satellite data.
- Large hadron collider and surveying the sky.
- Phone call records and speech recognition results.
- Video game worlds and user actions

What do you do with all this data?

- **Too much data** to search through it manually.

But there is valuable information in the data.

- How can we use it for fun, profit, health, and/or the greater good?

Data-driven problem solving (e.g., data mining and machine learning) are key tools we use to make sense of large datasets.
Data centers
There are **questions** that **data** can help us answer—questions like

- “Which stocks I should invest in?”
- “How can I live a healthier lifestyle?”
- “How can I understand my customers’ changing tastes, so that my business can serve them better?”

For instance, stock prices are observed at the exchange, aggregated by an intermediary like Thomson Reuters, stored in a database, bought by a company, converted into a Hive store on a Hadoop cluster, pulled out of the store by a script, subsampled, massaged, and cleaned by another script, dumped to a file, and converted to a format that you can try out in your favorite modeling.

The garden of bifurcating paths between data and answers
What types of ML are there?

- The nature of data (statistical, adversarial, benign, etc)
- The availability of outputs
- The availability of data (streaming, batch, etc)
- Interaction with environment
What types of ML are there?
The landscape of algorithms

**Supervised learning**
(Regression, Perceptron, Logistic Regression, Nearest Neighbor, Decision Trees, Support Vector Machines, Ensemble Methods (Bagging + Boosting), Deep Learning)

**Unsupervised learning**
(Clustering, Principle Component Analysis (PCA), Matrix factorization)

**Online/Reinforcement learning**
(Bandits, Markov decision process, Sarsa and Q-learning)

Labeled training data is given initially
Interactive (feedback after decisions)
No labeled training

Predict unseen data
Maximize rewards
Extract hidden patterns
Supervised learning

Given labeled examples, find the right prediction of an unlabeled example. (e.g. given annotated images learn to detect faces)

- Regression
- Binary classification (the label set is binary, e.g., spam detection)
- Multiclass classification (the label set is larger than two, digit recognition)
- Online (adversarial learning) (data are adversarial and come in a stream, e.g., stock prediction, weather prediction)
- Ranking (ranking instances, e.g., Search Engines)
Regression vs Classification

**Regression**

- Example: Tracking a mortgage
- Houses data

- Banks: Estimate the loan based on location, orientation, view, etc

- Output values: continuous

**Classification**

- Example: Spam classification
- Incoming emails

- How to group email in categories

- Output values: discrete, categorical
Abstract binary classification

In binary classification the label set is binary, \( \mathcal{Y} = \{-1, +1\} \)

Examples:
- Cancer diagnostics
- Credit approval or denial
- ….
The classification problem

- Example: Spam classification

- Input: incoming emails
- Output: the category of email

- Goal: group emails in categories

- Unlike regression, the labels set is discrete!

- Labels: discrete, categorical
Regression

- The goal is to make **quantitative** (real valued) predictions on the basis of a (vector of) **features** or **attributes**

  - Does number of **lung cancer deaths** change with **number of cigarettes**?
  - Does number of **skin cancer deaths** change with **latitude**?
  - Does number of **gun deaths** change with **gun ownership**?
  - Can we predict the **price of a stock**?

- What can we assume about the problem (recall No Free Lunch theorem)?
- How do we formalize the regression problem?
- How do we evaluate predictions?
- How to efficiently learn a model?
Unsupervised learning

Given data try to discover similar patterns, structure, sub-spaces (e.g. automatically cluster news articles by topic)

- No labeled are available
- But there is a pattern in data

Labeled data (colors)  Supervised

Un-labeled data (single color)  Unsupervised
Reinforcement learning

- Try to learn from delayed feedback (e.g. robot learns to walk, fly, play chess)
  - No training data at the beginning
  - We have to interact with an unknown environment to learn
And ...

- Active learning (we can request the label of an example)
- Semisupervised learning (we have both labeled and unlabeled examples)
- Structured prediction
- Graphical models
- Expectation maximization
- Approximate inference
- Density estimation
Instructor:

Mehrdad Mahdavi  mzm616@psu.edu
Office hours: Tu/Th 12:00 PM - 1:00 PM
Office: Westgate W365

Course meeting time:

Tu/Th 13:35-14:50 in Health and Hum Dev 254.

Teaching assistances (please check Canvas for OHs):

Pouria
Guney
# The calendar

<table>
<thead>
<tr>
<th>Time</th>
<th>Monday</th>
<th>Tuesday</th>
<th>Wednesday</th>
<th>Thursday</th>
<th>Friday</th>
</tr>
</thead>
<tbody>
<tr>
<td>9 AM-10 AM</td>
<td></td>
<td>Mehrdad's OH W365</td>
<td>Pouria's OH</td>
<td>Mehrdad's OH W365</td>
<td>Guney's OH</td>
</tr>
<tr>
<td>10 AM-11 AM</td>
<td></td>
<td>Lecture Health and Hum Dev 254</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11 AM-12 AM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12 AM-1 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1 PM-2 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2 PM-3 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3 PM-4 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4 PM-5 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5 PM-6 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
NO required textbook. All the materials can be found online and will be pointed to in Canvas!

- **[CML]** A Course in Machine Learning  
  by Hal Daumé III  
  (available online)

- **[PRML]** Pattern Recognition and Machine Learning  
  by Christopher Bishop  
  (available online)

- **[MLPP]** Machine Learning: A Probabilistic Perspective  
  by Kevin R Murphy

- **[ESL]** The Elements of Statistical Learning:  
  Data Mining, Inference, and Prediction,  
  by Hastie, Trevor, Tibshirani, Robert, Friedman, Jerome  
  (available online)

- **[RL]** Reinforcement Learning: An Introduction  
  by R Sutton and A Barto, MIT Press  
  (available online)

You are strongly recommended to not rely just on the slides and read the provided material!
Prerequisites

Three pillars of ML:

- Statistics / Probability
- Linear Algebra
- Multivariate Calculus

Should be confident in at least 1/3, ideally 2/3.
Math Resources

Mathematics of Machine Learning

- Linear Algebra
- Analytic Geometry
- Matrix Decompositions
- Vector Calculus
- Probability and Distribution
- Continuous Optimization

https://mml-book.github.io/
Grading Policy

- **5% (or more):** Attendances, quizzes, class participation, things that show initiative
- **40%:** Homework
- **25%:** Midterm
- **30%:** Final

There will be up to 10 random quizzes during the semester that will mostly contribute to 5% attendance.

The attendances will be based on weighted random sampling (random quizzes).

- Weight will be your score in the quiz.
Homeworks & exams

- There will be 5 to 6 assignments
- Assignments are both theory and programming
- HWs are challenging, start early!
- Play smart: take advantage of OHs held by TAs and myself to learn as much as you can (free resources)
- Penalty for late submissions, 25% of score per day

- There will be **two exams** in this course: a midterm and the final

- No make-up exam.
- The final is cumulative with more weight on after midterm material
## Course topics

### Background

- Calculus
- Linear algebra
- Probability, statistics, information theory
- Basic convex analysis and optimization  
  (Gradient Descent + Stochastic GD)

### Basics of Machine Learning

- The process of learning
- Training, evaluation, & generalization
- Model selection
- Regularization
- Bias-variance decomposition

### Supervised learning

- Regression (OLS + Ridge + Lasso)
- Perceptron
- Deep Artificial Neural Networks
- Logistic Regression
- Nearest Neighbor
- Decision Tress
- Support Vector Machines
- Ensemble Methods  
  (Bagging + Boosting)

### Unsupervised learning

- Clustering (k-means & k-means++)
- Mixture of Gaussians
- Principle Component Analysis (PCA)
- Matrix factorization

### Reinforcement learning

- Bandits and Multiplicativce Update
- Markov decision process
- Dynamic programming
- Sarsa and Q-learning algorithms
There are few open-source library/tools that you will use some of them in doing homework:

- Python (numpy, scipy, matplotlib, etc)
- SciKit-Learn
- TensorFlow
- PyTorch
- Pandas
- R
Sanity check

- You are comfortable with doing math and statistics (calculus, linear algebra, optimization, etc)!
- Willing to write Python code.
- Not afraid of dealing with data.
All the assignments will be graded in Gradescope.

Please claim the course with entry Code: DJXX68)

- All the implementation/codes needs to be submitted to Canvas
- All the PDFs needs to be submitted to Gradescope

* Please claim the course in Gradescope
So in this course ...

- You will be trained as **machine learning** scientists rather **data scientists**!
- Data science skills are very important as well, but not the focus of this course.
- So, do not expect to learn about Hadoop or Spark in this course!
- Do not wait for a lecture on cloud computing!
- The approach is this course will be **constructive**, how to design machine learning algorithms from intuition, formulate them as a mathematical problem, and finally solve them!
- Be ready to do some math (calculus, linear algebra, probability, optimization)
- Deep learning is just one kind of machine learning that is popular know!
- We will use Python and other open source packages (numpy, scikit-learn, pandas) to get our hands dirty with data!
Todos

* Refresh your mind on linear algebra, probability, statistics, calculus, etc. Please check Canvas for some good resources but you can start from these books:

  - Mathematics for Machine Learning by Marc Peter Deisenroth, A. Aldo Faisal, Cheng Soon Ong
  - Mathematical Foundations for Data Analysis by Jeff M. Phillips

* Claim the course in Gradescope
“A breakthrough in machine learning would be worth ten Microsofts” (Bill Gates, Chairman, Microsoft)

“Machine learning is the next Internet” (Tony Tether, form. Director, DARPA)

“Machine learning is the hot new thing” (John Hennessy, President, Stanford)

“Web rankings today are mostly a matter of machine learning” (Prabhakar Raghavan, Director Research, Yahoo)

“Machine learning is going to result in a real revolution” (Greg Papadopoulos, CTO, Sun)

“Machine learning is today’s discontinuity” (Jerry Yang, CEO, Yahoo)

“AI vastly more risky than North Korea” (Elon Musk, CEO Tesla, Spaxe X, …)
ML is becoming ubiquitous in science, engineering and beyond

This class should give you the basic foundation for applying ML and developing new methods

The fun begins…