Welcome to DATA1030: Hands-on Data Science!

Instructor: Andras Zsom

HTA: Isaac Wecht

TAs: Kameel Dossal, YouJung Koo, Yang Zheng,

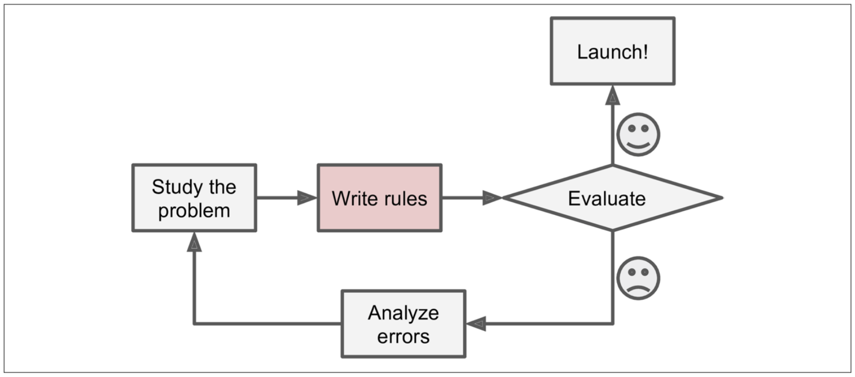
Tianqi Cheng, Ella Liang, Ji Zhang, Chen Wei

The goal of this course: supervised Machine Learning (ML)

- supervised ML is probably the most successful area in ML (based on value created)
  - **online advertising**: given an ad and user info, will the user click on the ad?
  - **real estate**: given home features, can we predict the house price?
  - **finance**: given an applicant and a finalcial product (e.g., a loan), will this applicant be able to successfully pay back the loan?
  - **health care**: given a patient, symptoms, and maybe test results, can we predict the illness?
  - ...
- supervised ML pros:
  - **automation**: computers perform calculations faster than humans (and computers are cheaper)
  - **learn from examples**: no need to explicitly tell the computer what to do. the computer figures out what to do based on examples (data)
- supervised ML con:
  - it can be difficult or labor-intensive to collect training data
  - there is no guarantee that you will be able to develop an accurate model based on the data you have

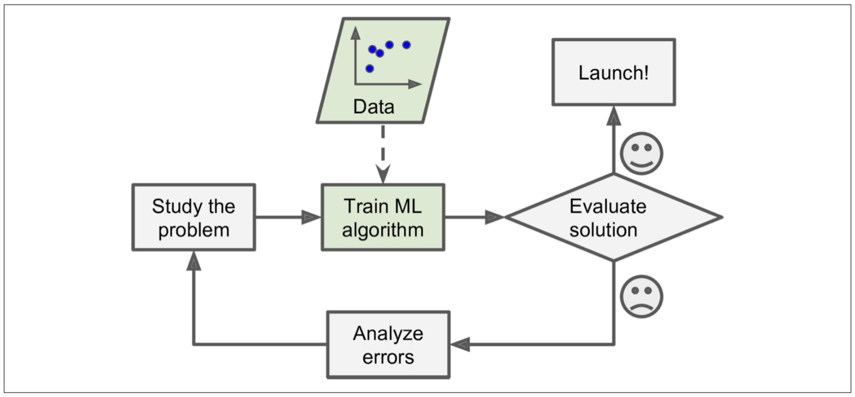
Example: spam filters

- Traditional coding pipeline with explicit instructions



Example: spam filters

- ML pipeline



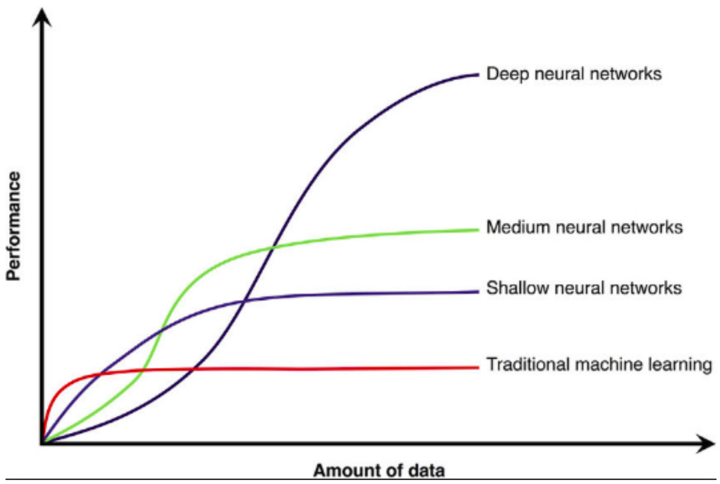
- the data: feature matrix (X) and target variable (Y)
  - X can be structured (tabular data most commonly stored in excel and csv files or SQL databases)
  - X can be unstructured (e.g., images, text, voice recording, video)
  - Y can be categorical, the problem is **classification** (e.g., click or not click on an ad, sick or not sick)
  - Y can be continuous, the problem is **regression** (e.g., predict house price, stock price, age)
- **we focus on structured data during this class!**

Structured data

X	feature_1	feature_2	...	feature_j	...	feature_m	Y
data_point_1	x_11	x_12	...	x_1j	...	x_1m	**y_1**
data_point_2	x_21	x_22	...	x_2j	...	x_2m	**y_2**
...	...	...	...	...	...	...	**...**
data_point_i	x_i1	x_i2	...	x_ij	...	x_im	**y_i**
...	...	...	...	...	...	...	**...**
data_point_n	x_n1	x_n2	...	x_nj	...	x_nm	**y_n**

Other areas of ML

- unsupervised ML
  - only the feature matrix X is available, there is no target variable
  - the goal is to find structure (clusters) in the data
  - often used in customer segmentation
- recommender systems
  - recommend products to a customer based on what products similar customers enjoyed
- reinforcement learning
  - the learning system, called an agent, can observe the environment, select and perform actions, and get rewards and penalties in return. Goal: come up with strategy to maximize rewards
  - often used when virtual environment is available (e.g., games like go or warcraft)
  - sounds appealing to use in real environments (like self-driving cars) but agents learn slow, lots of cars would need to be broken to teach an agent to drive this way
- deep learning
  - uses neural networks and often works with unstructured data
  - technically deep learning is supervised or unsupervised
  - extremely successful on large datasets



Quiz

Learning objectives

By the end of the semester, you will be able to

- explore and visualize the dataset,
- develop a ML pipeline from scratch to deployment,
- make data-driven decisions during the pipeline development,
- handle non-standard ML problems like missing data, non-iid data,
- provide explanations with your model,
- explain your findings to technical and non-technical audiences.

A few words about python

- widely used in data science because of sklearn, pandas, deep learning packages
  - packages are easy to (mis)use
- relatively easy to write code but difficult to write computationally efficient code
  - the divide between package developers and users is huge!
  - you will need to spend a lot of time reading the manuals and verifying results
- the lecture notes contain code that has been tested
  - this is misleading!
  - I spent a lot of time testing the code but I deleted those lines to keep the final code clean
  - but when you write code, you should absolutely PRINT ALL VARIABLES and TEST EVERY SINGLE LINE!
  - you will learn how to interpret error messages and how to debug your code
- test-driven code development is encouraged
  - first come up with a test
    - create a couple of test cases with known results
    - i.e., if my code does what I think it should, I'll get a certain output given certain input
  - then write the code

Course structure

Canvas: https://canvas.brown.edu/courses/1092452

Course components:

- lectures
  - in person but recordings will be posted on canvas
- weekly problem sets, submit them on Gradescope
  - coding problems and questions with 1-2 paragraph answers

    - the questions prepare you for your job interviews
- one semester-long project
    - find a dataset and come up with your own machine learning question
    - develop code individually, but feel free to discuss with others
    - assigned TA mentor with regular dedicated meetings

Grading

- weekly problem sets: **50%** weight
- project: **50%** weight
    - make sure to spend sufficient time on this each week!
    - the semester will go by very quickly...
- **90% minimum is necessary to get an A** but I reserve the right to lower the threshold
- my experience is that Bs are rare, C is given under exceptional circumstances

Project

- look for datasets on the UCI Machine Learning Repository, on Kaggle, or google's dataset search engine.
- Bring your own dataset!
    - if you have your own dataset you'd like to work with, this is the perfect opportunity!
- Avoid the most popular datasets!
    - no Titanic, no iris for example
- avoid these four datasets because we will use them in class and you'll work with them in the problem sets
    - adult dataset
    - kaggle house price dataset
    - hand postures dataset
    - diabetes dataset
- work on a classification or regression problem!
- start looking for datasets now and talk to the TAs or come to my office hours if you have questions!
- there are three main reasons why a ML problem is difficult:
    - missing data
    - dataset is not IID (e.g., time series data, or one object is described by multiple data points)
    - dataset is large (more than 100k points) so it is difficult to manage it on your laptop
- choose datasets with at least one difficulty!

Override codes

- DSI master's students get priority because the course is mandatory for them
- everyone else who fulfilled the prereqs will get overrides on a first come first serve basis
- submit override requests on cab.brown.edu
- Please DO NOT send me emails about overrides! I'm drowning in email, I simply can't answer them. :(

Generative AI

- while I can't ban generative AI tools (like ChatGPT, Bard, github's copilot), I also don't recommend relying on them too much
- I tried to solve some problem sets with ChatGPT's Code Interpreter (see here)
- It is not very good with complex or ill-defined tasks
- It is not reproducable
- You might not be able to tell when it gives wrong answers while you are still learning
- Most companies still do live coding interviews and technical interviews with no tools allowed
    - you will likely not succeed on those if you rely too much on generative AI and code completion tools
- Use generative AI if
    - you need some help to debug your code
    - you want to fix the grammar of some text you wrote
    - you need some data science concept clarified
- If you use generative AI
    - cite the tool used
    - describe how you used the tool (i.e., what was your prompt)

  - disclose what was your contribution vs the tool's contribution
- It is cheating to use generative AI to
  - solve the coding exercises for you
  - answer the essay questions for you
- If you have any questions about academic integrity, plagiarism, what's considered cheating and what's not, don't hesitate to ask!

Rough deadlines

- **1st project presentation**: early/mid October (multiple dates)
  - short presentation on dataset, EDA, and ML question (6 min + 3 min questions per student)
  - rubric will be available two weeks in advance
- **1st project progress report**: early/mid October
  - dataset selection, EDA, and formulate your ML question
  - rubric will be available two weeks in advance
- **final presentations**: early December (probably the week of December 4)
  - another short presentation on ML pipeline and results
  - rubric will be available two weeks in advance
- **final project report**: early December (probably the week of December 4)
  - the complete ML pipeline and results
  - rubric will be available two weeks in advance
- **final exam**: December 13th, 2pm
- grades finalized and submitted by December 15th
- **Feel free to fly out on or after December 15th for the holidays as far as this course is concerned!**

Other course resources

- Ed discussion: course forum
  - feel free to discuss any questions or concerns regarding the material
  - please post publicly whenever possible (but you can still post anonymously)
    - if you have a question, it is likely that multiple students have the same question
  - the TAs and I will keep an eye on it and answer questions in a timely manner
  - disclaimer: I turn off my laptop after 5pm and during the weekends
- office hours (TAs and mine)
  - I'll post dates and locations on the course forum
- An Introduction to Statistical Learning (book)
- Introduction to Machine Learning with Python (book)
- Harry Potter and the Methods of Rationality (fan fiction by Eliezer Yudkowski)
  - half joking, half serious about this one :)

Mud card

In [ ]:

```
```