10-301/601: Introduction to Machine Learning Lecture 1 – Problem Formulation & Notation

Henry Chai

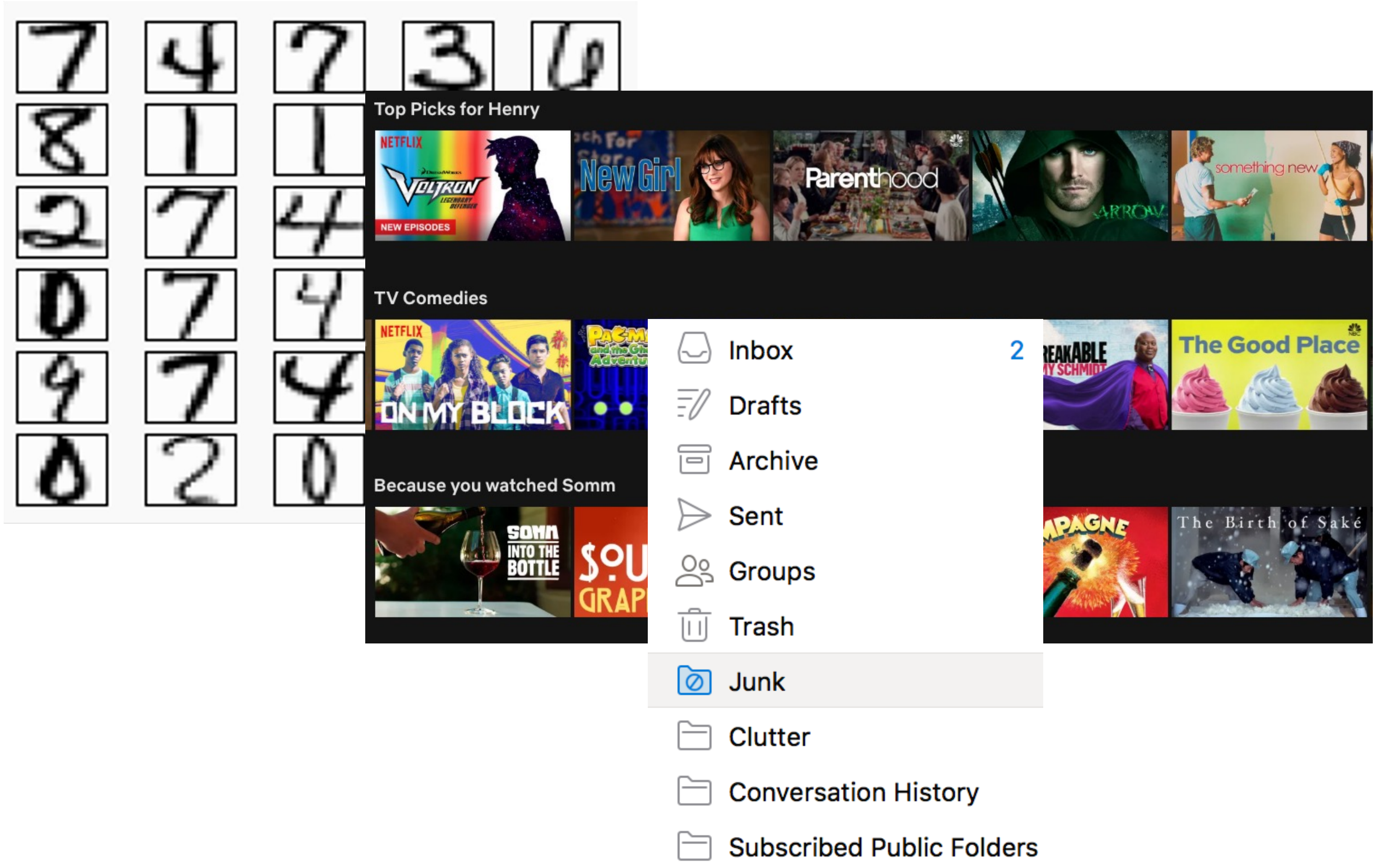
5/15/23

Front Matter

- Announcements:
  - PA0 released 5/15 (today!), due 5/18 at 11:59 PM
    - You must complete all assignments using LaTeX; see this Piazza post for details and a few LaTeX tutorials
  - General advice for the summer:
    - Start HWs early!
    - Go to office hours! Starting today, 5/15
      - MWThF (every weekday except Tuesday) from 5 – 6 PM in NSH 3002
- Recommended Readings:
  - None

What is Machine Learning?

Machine Learning (A long long time ago…)



Machine Learning
(A short time ago…)

Machine Learning (Now)

HC Henry: Hey Chad, how's it going?

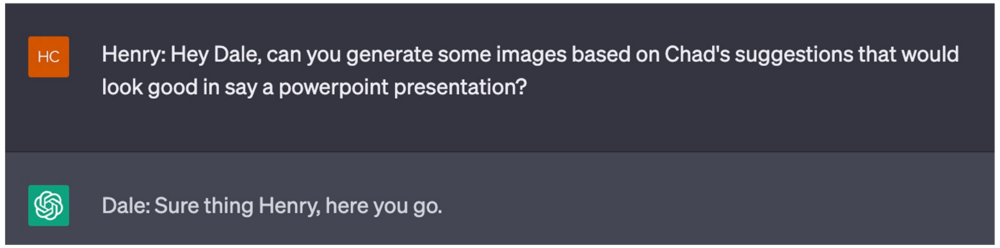
Chad: It's all good, man. Just living the dream, you know what I'm saying? How about you?

HC Henry: I'm good, thanks man. Can you tell me what you think are the three most exciting applications of machine learning in say the past year?

Chad: Sure thing, dude. There's been a ton of cool stuff happening in the machine learning space lately, but if I had to pick just three, I'd say:

1. GPT-3 and natural language processing (NLP) - The latest iteration of OpenAI's GPT series is pretty mind-blowing. It's got 175 billion parameters, which is orders of magnitude more than any other language model out there, and it's capable of some seriously impressive language generation and understanding. It's not perfect, of course, but the potential applications for this kind of technology are huge.
2. Image and video synthesis - There have been some really cool developments in generative adversarial networks (GANs) recently, which allow for the synthesis of incredibly realistic images and videos. This has a ton of potential applications, from creating realistic virtual environments to improving the quality of low-resolution images and videos.
3. Explainable AI (XAI) - As AI becomes more ubiquitous, the need for transparency and interpretability becomes increasingly important. XAI is all about developing models and techniques that allow us to understand how and why AI systems are making the decisions they do. This is crucial for ensuring that AI is used in a responsible and ethical way, and it's an exciting area of research right now.

Machine Learning (Now)



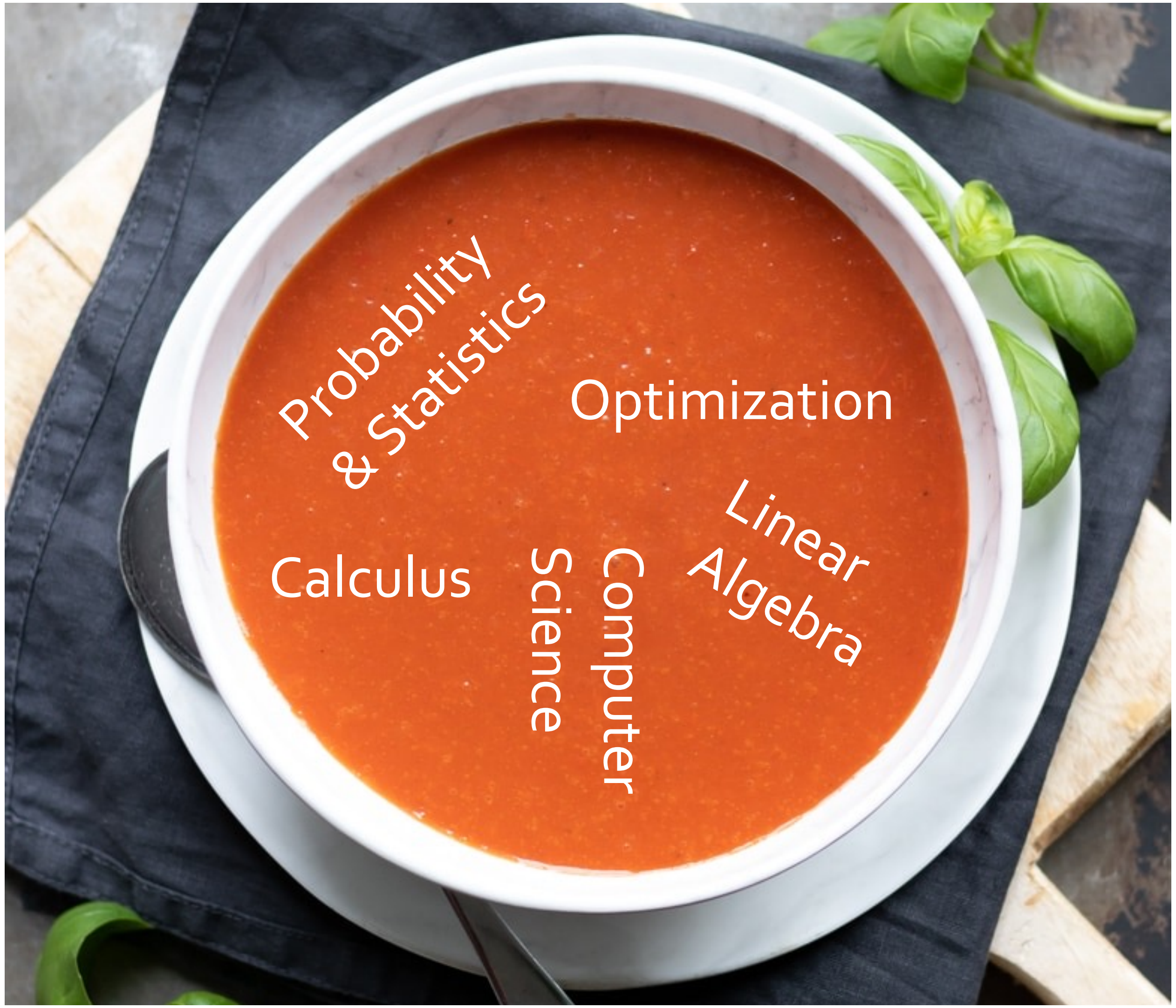
Source: https://www.bing.com/images/create?FORM=GERRLP

Source: https://chat.openai.com/

Premise of Machine Learning

- There exists some pattern/behavior of interest
- The pattern/behavior is difficult to describe
- There is data
- Use data to "learn" the pattern

What is Machine Learning?



Source: https://veggiedesserts.com/easy-tomato-soup-recipe/

Things Machine Learning Isn't

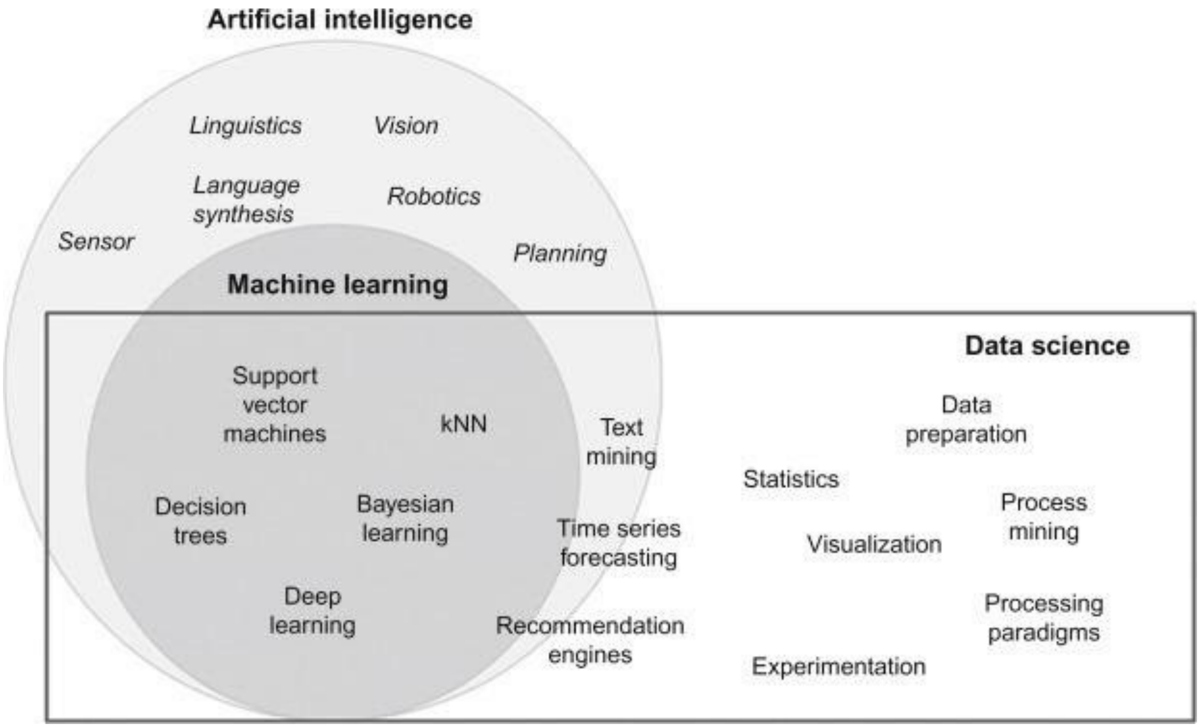
- Artificial intelligence
- Data science

Things Machine Learning Isn't

- Artificial intelligence: Creating machines that can mimic human behavior/cognition
- Data science

Things Machine Learning Isn't

- Artificial intelligence: Creating machines that can mimic human behavior/cognition
- Data science: Extracting knowledge/insights from noisy, unstructured data



Source: https://www.sciencedirect.com/topics/physics-and-astronomy/artificial-intelligence

What is ~~Machine Learning~~ 10-301/601?

- Supervised Models
  - Decision Trees
  - KNN
  - Naïve Bayes
  - Perceptron
  - Logistic Regression
  - Linear Regression
  - Neural Networks
- Unsupervised Models
  - K-means
  - PCA
- Ensemble Methods
- Graphical Models
  - Bayesian Networks
  - HMMs
- Learning Theory
- Reinforcement Learning
- Important Concepts
  - Feature Engineering
  - Regularization and Overfitting
  - Experimental Design

Defining a Machine Learning Task (Mitchell, 97)

- A computer program **learns** if its *performance*, *P*, at some *task*, *T*, improves with *experience*, *E*.
- Three components
  - Task, T
  - Performance metric, P
  - Experience, E

Defining a Machine Learning Task: Example

- Learning to approve loans/lines of credit

- Three components
  - Task, T

    Decide whether to extend someone a loan
  - Performance metric, P

    Number of people who default on their loan
  - Experience, E

    Interviews with loan officers

Defining a Machine Learning Task: Example

- Learning to approve loans/lines of credit

- Three components
  - Task, T

    Predict the probability someone defaults on a loan
  - Performance metric, P

    Amount of money (interest) made
  - Experience, E

    Historical data on loan defaults

Things Machine Learning Isn't

- Artificial intelligence: Creating machines that can mimic human behavior/cognition
- Data science: Extracting knowledge/insights from noisy, unstructured data
- Neutral?

Do you agree or disagree with the following sentence: "Because machine learning uses algorithms, math and data, it is inherently neutral or impartial."

- Agree
- Unsure
- Disagree

Things Machine Learning Isn't

- Artificial intelligence: Creating machines that can mimic human behavior/cognition
- Data science: Extracting knowledge/insights from noisy, unstructured data
- Neutral

Big Data: A Report on Algorithmic Systems, Opportunity, and Civil Rights

Executive Office of the President

May 2016

THE WHITE HOUSE
WASHINGTON

Source: https://obamawhitehouse.archives.gov/sites/default/files/microsites/ostp/2016_0504_data_discrimination.pdf

Things Machine Learning Isn't

- Artificial intelligence: Creating machines that can mimic human behavior/cognition
- Data science: Extracting knowledge/insights from noisy, unstructured data
- Neutral

OPPORTUNITIES AND CHALLENGES IN BIG DATA

The Assumption: Big Data is Objective

It is often assumed that big data techniques are unbiased because of the scale of the data and because the techniques are implemented through algorithmic systems. However, it is a mistake to assume they are objective simply because they are data-driven.[13]

The challenges of promoting fairness and overcoming the discriminatory effects of data can be grouped into the following two categories:

1) Challenges relating to ***data used as inputs*** to an algorithm; and
2) Challenges related to ***the inner workings of the algorithm itself***.

Source: https://obamawhitehouse.archives.gov/sites/default/files/microsites/ostp/2016_0504_data_discrimination.pdf

Defining a Machine Learning Task: Example

- Learning to

- Three components
  - Task, T
  - Performance metric, P
  - Experience, E

Defining a Machine Learning Task: Example

- Learning to

- Three components
  - Task, T
  - Performance metric, P
  - Experience, E

Our first Machine Learning Task

- Learning to diagnose heart disease

  as a **(supervised) binary classification task**

Family History	Resting Blood Pressure	Cholesterol	Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes

features

labels

data points

Our first Machine Learning Task

- Learning to diagnose heart disease

  as a **(supervised) binary classification task**

features			labels
Family History	Resting Blood Pressure	Cholesterol	Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes

data points

Our first Machine Learning Task

- Learning to diagnose heart disease

  as a **(supervised)** **binary classification** **task**

features | labels

data points

Family History	Resting Blood Pressure	Cholesterol	Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes

Our first Machine Learning Task

- Learning to diagnose heart disease

  as a **(supervised)** **<u>classification</u> task**

features

labels

data points

Family History	Resting Blood Pressure	Cholesterol	Risk
Yes	Low	Normal	Low Risk
No	Medium	Normal	Low Risk
No	Low	Abnormal	Medium Risk
Yes	Medium	Normal	High Risk
Yes	High	Abnormal	High Risk

Our first Machine Learning Task

- Learning to diagnose heart disease

  as a **(supervised)** **<u>regression</u> task**

Family History	Resting Blood Pressure	Cholesterol	Medical Costs
Yes	Low	Normal	\$0
No	Medium	Normal	\$20
No	Low	Abnormal	\$30
Yes	Medium	Normal	\$100
Yes	High	Abnormal	\$5000

Our first Machine Learning Classifier

- A **classifier** is a function that takes feature values as input and outputs a label
- Majority vote classifier: always predict the most common label in the   dataset

features

labels

data points

Family History	Resting Blood Pressure	Cholesterol	Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes

Is this a "good" Classifier?

- A **classifier** is a function that takes feature values as input and outputs a label
- Majority vote classifier: always predict the most common label in the dataset

features

labels

data points

Family History	Resting Blood Pressure	Cholesterol	Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes

Training vs. Testing

- A **classifier** is a function that takes feature values as input and outputs a label
- Majority vote classifier: always predict the most common label in the **training** dataset (Yes)

training dataset

Family History	Resting Blood Pressure	Cholesterol	Heart Disease?
Yes	Low	Normal	No
No	Medium	Normal	No
No	Low	Abnormal	Yes
Yes	Medium	Normal	Yes
Yes	High	Abnormal	Yes

Training vs. Testing

- A **classifier** is a function that takes feature values as input and outputs a label
- Majority vote classifier: always predict the most common label in the **training** dataset (Yes)
- A **test** dataset is used to evaluate a classifier's **predictions**

test dataset

Family History	Resting Blood Pressure	Cholesterol	Heart Disease?	**Predictions**
No	Low	Normal	No	Yes
No	High	Abnormal	Yes	Yes
Yes	Medium	Abnormal	Yes	Yes

- The **error rate** is the proportion of data points where the prediction is wrong

Training vs. Testing

- A **classifier** is a function that takes feature values as input and outputs a label
- Majority vote classifier: always predict the most common label in the **training** dataset (Yes)
- A **test** dataset is used to evaluate a classifier's **predictions**

test dataset

Family History	Resting Blood Pressure	Cholesterol	Heart Disease?	**Predictions**
No	Low	Normal	No	Yes
No	High	Abnormal	Yes	Yes
Yes	Medium	Abnormal	Yes	Yes

- The **test error rate** is the proportion of data points in the test dataset where the prediction is wrong ($1/3$)

A Typical (Supervised) Machine Learning Routine

- Step 1 – training
  - Input: a labelled training dataset
  - Output: a classifier
- Step 2 – testing
  - Inputs: a classifier, a test dataset
  - Output: predictions for each test data point
- Step 3 – evaluation
  - Inputs: predictions from step 2, test dataset labels
  - Output: some measure of how good the predictions are; usually (but not always) error rate

Key Takeaways

- Components of a machine learning problem
- Machine learning vs. artificial intelligence vs. data science
- Algorithmic bias
- Components of a labelled dataset for supervised learning
- Training vs. test datasets
- Majority vote classifier

Logistics: Course Website

https://www.cs.cmu.edu/~hchai2/courses/10601

Logistics: Course Syllabus

https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus

- This whole section is **required** reading

Logistics: Grading

https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus

- 30% programming assignments
- 25% in-class quizzes
- 20% midterm
- 20% final
- 5% participation
    - 5% (full credit) for 80% or greater poll participation
    - 3% for 65%-80% poll participation.
    - 1% for 50%-65% poll participation.
    - "Correctness" will not affect your participation grade
    - 50% credit for responses before the next lecture

Logistics: Programming Assignments

https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus

- 8 programming assignments throughout the semester
  - PA0 (out today!) is a self-assessment covering background/pre-requisite material
  - Each will have a programming component and some written, empirical questions
  - Your answers to the written questions must be typeset in LaTeX
    - To facilitate this, we will always provide a LaTeX starter template that you can just fill in with your answers.
  - You will submit your code and your answers to the written questions separately, both using Gradescope

Logistics: Late Policy

https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus

- 9 grace days for use across all programming assignments
- Only 3 grace days may be used per homework
- Late submissions w/o grace days:
  - 1 day late = 75% multiplicative penalty
  - 2 days late = 50% multiplicative penalty
  - 3 days late = 25% multiplicative penalty
- No submissions accepted more than 3 days late

Logistics: In-class Quizzes

https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus

- 10 weekly quizzes throughout the semester
  - Each quiz covers the previous week's content
  - The goal of these "frequent", low-stakes quizzes is to keep you up to date on the material and serve as regular check-ins for your understanding
  - To help you prepare:
    1. We will release a set of study questions at the end of each week
    2. Our TAs will go over some additional practice problems in recitation
  - **At least 75% of the points on the in-class quizzes will come from questions that are identical or nearly identical to questions from these sources**

Logistics: Collaboration Policy

https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus

- **On study materials and recitation handouts, you may collaborate freely, to any extent**
  - **However, you still have a duty to protect your work:** you may not post your solutions publicly/share your solutions with anyone outside of the course
- Collaboration on programming assignments is encouraged but must be documented
- **You must always write your own code/answers**
  - You may not re-use code/previous versions of the homework, whether your own or otherwise
- Good approach to collaborating on programming assignments:
  1. Collectively sketch pseudocode on an impermanent surface, then
  2. Disperse, erase all notes and start from scratch

Logistics: Technologies

https://www.cs.cmu.edu/~hchai2/courses/10601/#Syllabus

- Piazza, for course discussion: https://piazza.com/class/lh7wb71rd8z7ct/
- Gradescope, for submitting homework assignments: https://www.gradescope.com/courses/53741
- Polleverywhere, for in-class participation: https://pollev.com/301601polls
- Panopto, for lecture recordings: https://scs.hosted.panopto.com/Panopto/Pages/Sessions/List.aspx#folderID=%223c224789-15ee-41c1-a95f-affd012e5344%22

Logistics: Lecture Schedule

https://www.cs.cmu.edu/~hchai2/courses/10601/#Schedule

Schedule

Date	Topic	Slides	Readings/Resources
Mon, 5/15	Introduction: Notation & Problem Formulation		
Tue, 5/16	Decision Trees - Model Definition & Making Predictions		
Wed, 5/17	Decision Trees - Learning		
Mon, 5/22	Nearest Neighbors		
Tue, 5/23	Quiz 1: Decision Trees		
	Model Selection (Mini-lecture)		
Wed, 5/24	Perceptron		
Mon, 5/29	No Class (Memorial Day)		
Tue, 5/30	Quiz 2: KNN, Model Selection & Perceptron		
	Linear Regression (Mini-lecture)		
Wed, 5/31	Optimization for Machine Learning		

Logistics: Exam Schedule

https://www.cs.cmu.edu/~hchai2/courses/10601/#Schedule

Schedule

Date	Topic	Slides	Readings/Resources
⋮			
Fri, 6/23	Midterm Exam (Time and Location TBD)		
Mon, 6/26	No Class (Summer Break)		
⋮			
Fri, 8/11	Final Exam (Time and Location TBD)		

Logistics: Recitations

https://www.cs.cmu.edu/~hchai2/courses/10601/#Recitations

Recitations

Attendance at recitations is not required, but strongly encouraged. Recitations will be interactive and focus on problem solving; we strongly encourage you to actively participate. A problem sheet will usually be released prior to the recitation. If you are unable to attend one or you missed an important detail, feel free to stop by office hours to ask the TAs about the content that was covered. Of course, we also encourage you to exchange notes with your peers.

Date	Topic	Handout
Thu, 5/18	Recitation 1: Decision Trees	
Thu, 5/25	Recitation 2: KNN, Model Selection & Perceptron	
Thu, 6/01	Recitation 3: Linear Regression & Optimization	
Thu, 6/08	Recitation 4: MLE/MAP, Logistic Regression & Regularization	
Thu, 6/15	Recitation 5: Neural Networks	
Tue, 6/20	Midterm Practice Problem Review	
Thu, 6/22	Reading Day - Office Hours in lieu of Recitation	
Thu, 6/29	No Recitation (Summer Break)	
Thu, 7/06	Recitation 6: Deep Learning & Learning Theory	
Thu, 7/13	Recitation 7: Unsupervised Learning & Naïve Bayes	
Thu, 7/20	Recitation 8: Graphical Models	
Thu, 7/27	Recitation 9: Reinforcement Learning	
Thu, 8/03	Recitation 10: Ensemble Methods	
Tue, 8/08	Final Practice Problem Review	
Thu, 8/10	Reading Day - Office Hours in lieu of Recitation	

Logistics: Programming Assignments

https://www.cs.cmu.edu/~hchai2/courses/10601/#Assignments

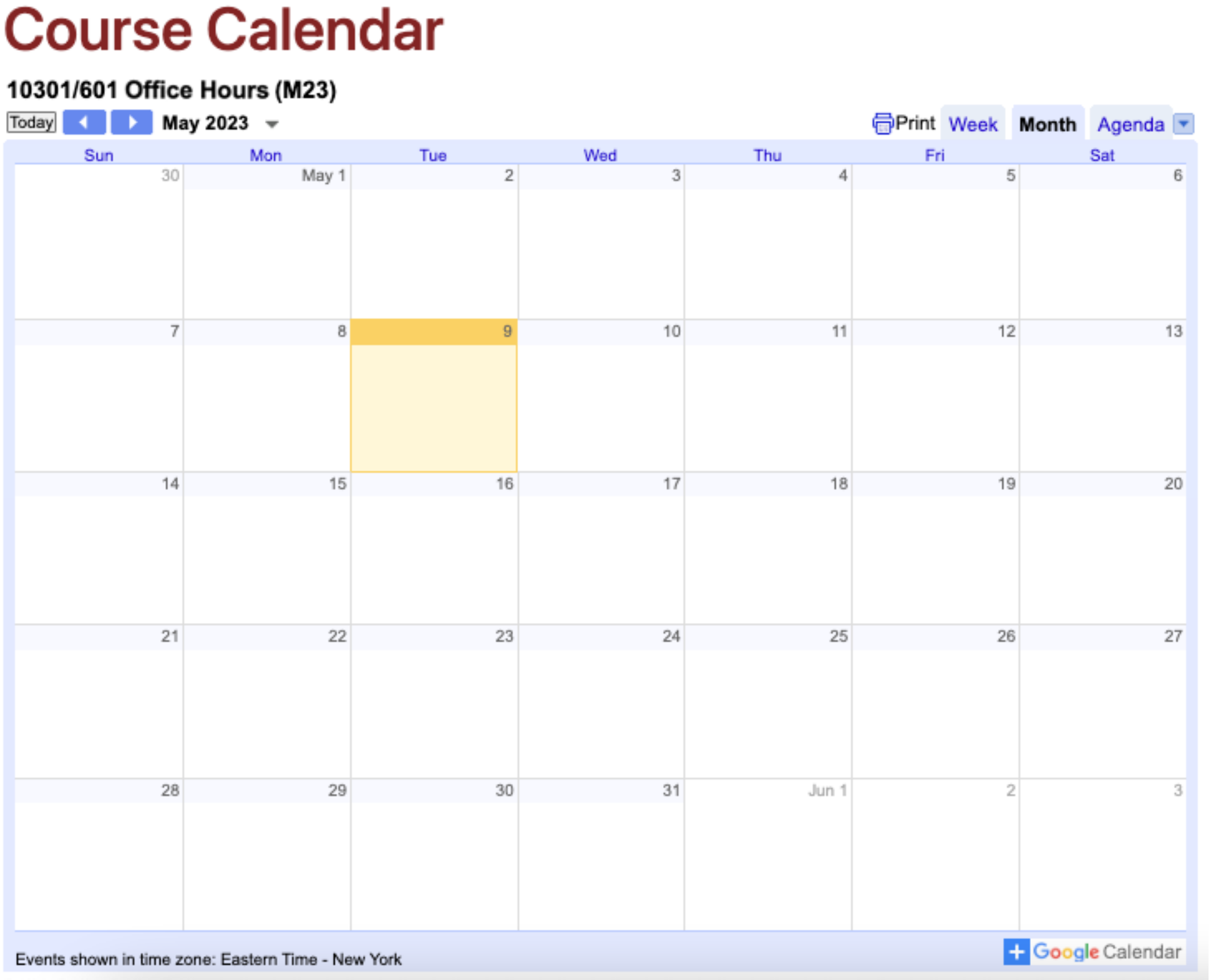
Programming Assignments

Our programming assignments are an opportunity for you all to build and experiment with some of the models that we introduce in class. All programming assignments must be completed in Python and the responses to the empirical questions must be written in LaTeX. You will submit both your code and your answers to the empirical questions using Gradescope; note that each assignment will have separate submissions for the code and the written portion.

Release Date	Topic	Files	Due Date
Mon, 5/15	PA0: Background Material		Thu, 5/18 at 11:59 PM
Thu, 5/18	PA1: Decision Trees		Thu, 5/25 at 11:59 PM
Thu, 5/25	PA2: KNN & Model Selection		Thu, 6/01 at 11:59 PM
Thu, 6/08	PA3: Logistic Regression		Thu, 6/15 at 11:59 PM
Thu, 6/15	PA4: Neural Networks		Thu, 7/13 at 11:59 PM
Thu, 7/13	PA5: Unsupervised Learning		Thu, 7/20 at 11:59 PM
Thu, 7/20	PA6: Graphical Models		Thu, 7/27 at 11:59 PM
Thu, 7/27	PA7: Reinforcement Learning		Thu, 8/03 at 11:59 PM

Logistics: Office Hours

https://www.cs.cmu.edu/~hchai2/courses/10601/#Calendar

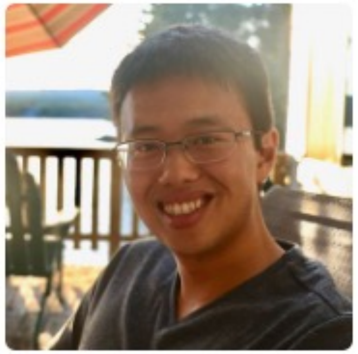


Logistics: Staff

https://www.cs.cmu.edu/~hchai2/courses/10601/#Staff

Instructor

Henry Chai



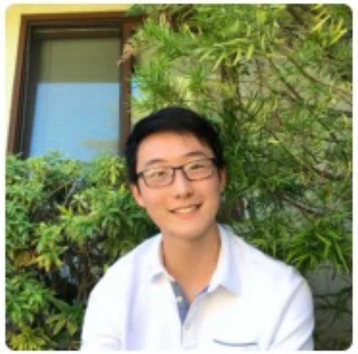
Education Associate

Joshmin Ray

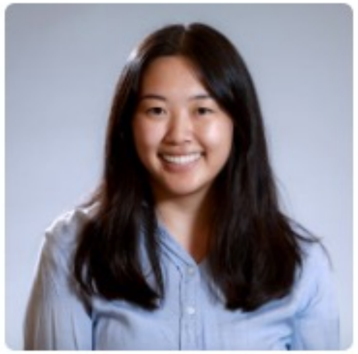
Teaching Assistants

Alex Xie

Andrew Wang



Sofia Kwok



Tara Lakdawala