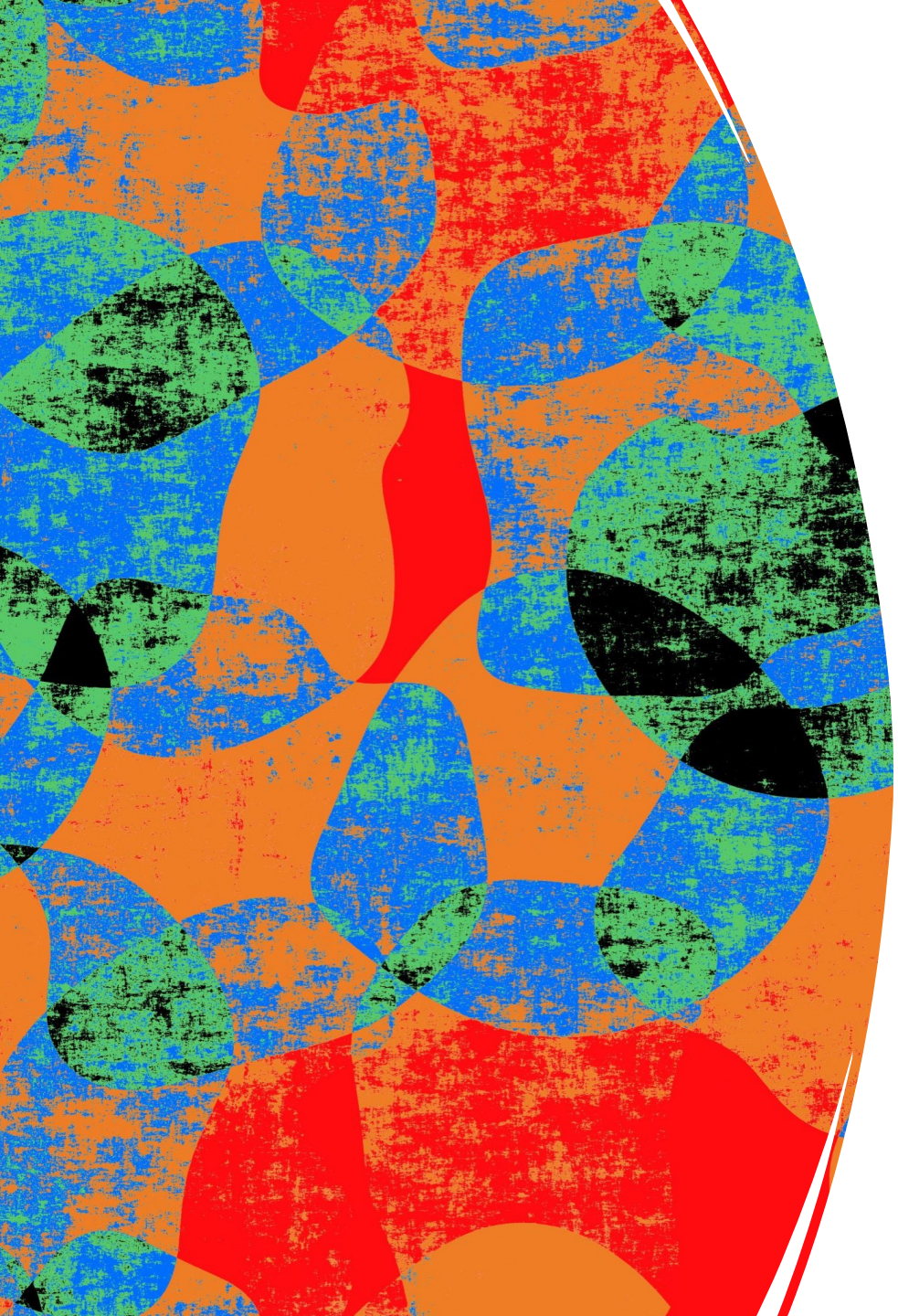
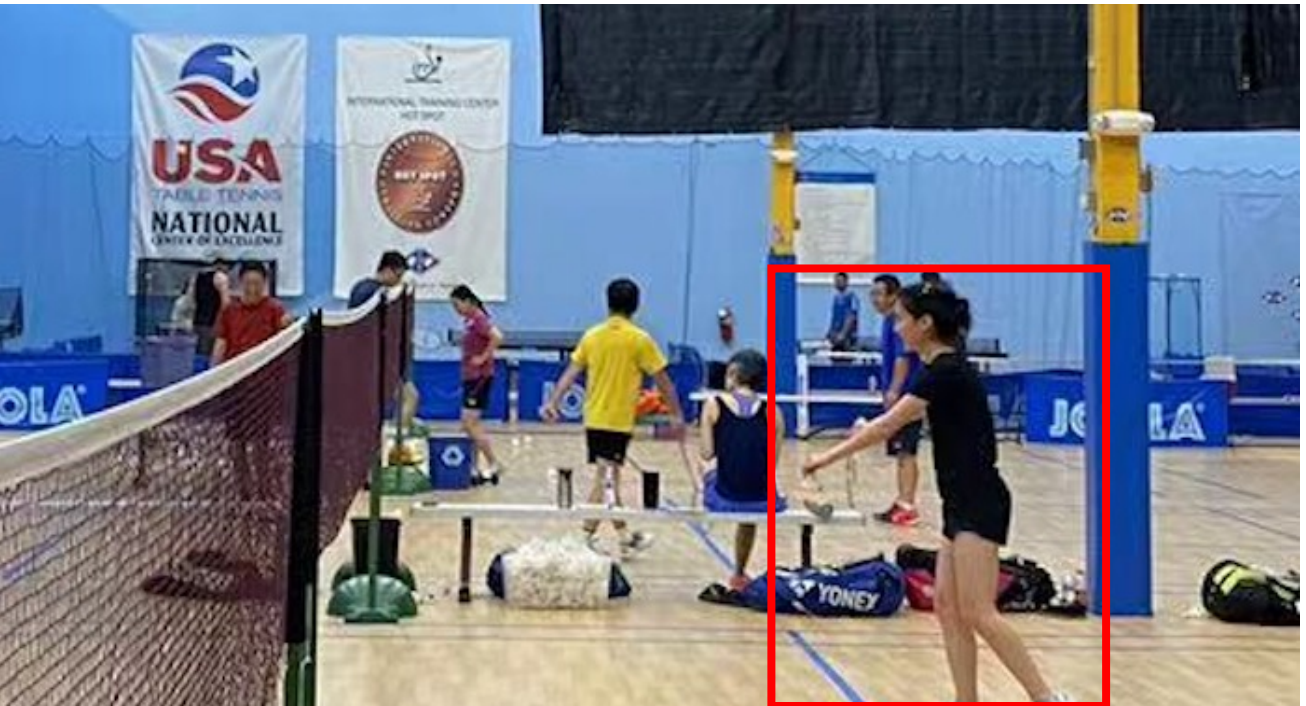


UMassAmherst

College of Information & Computer Sciences

Machine Learning

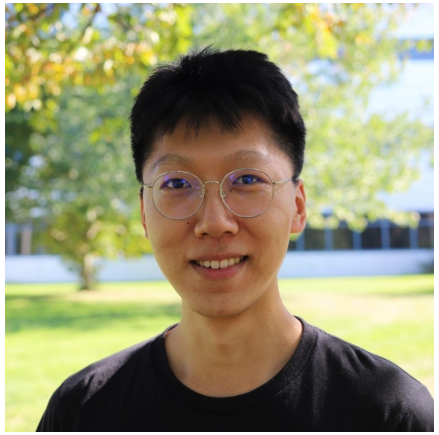
Instructor: Hui Guan



About me

- Assistant Professor at CS department
- Research Area:
  - Machine Learning
  - Edge Intelligence
  - Programming Systems

About the TAs

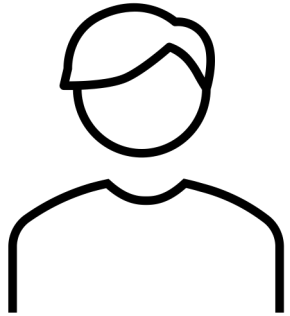


Yunda Liu

William Rebelsky

Chhandak Bagchi

Kaleigh Clary



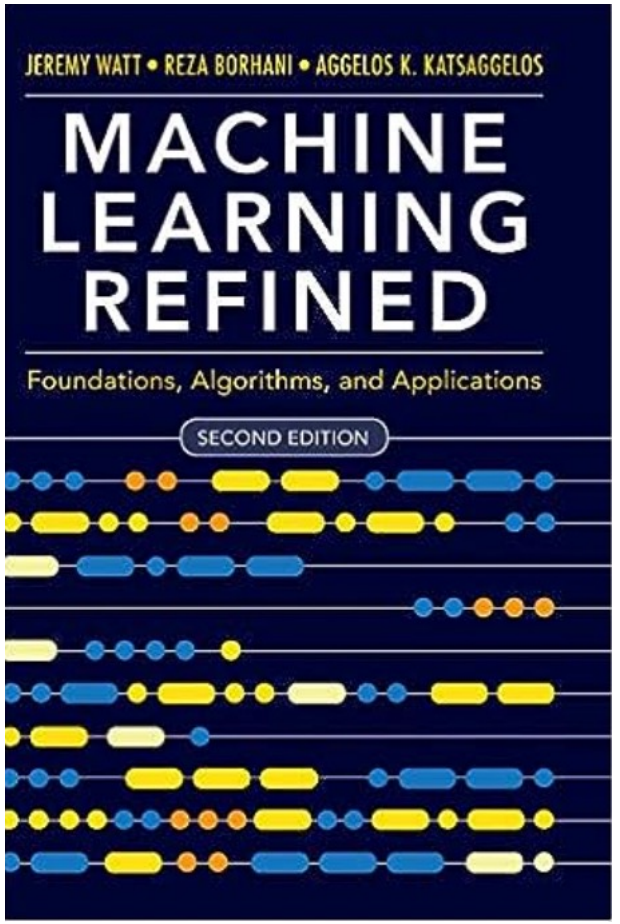
Yiquan Xiao (Bryan)

About the Course: management

- Website: https://sites.google.com/umass.edu/compsci589-fall23/home [look for “Teaching” → “2023Fall”]
  - Syllabus
  - Tentative Course Schedule
- Canvas:
  - Lectures will be recorded using ECHO360 (links available on Canvas)
  - Slides
- Gradescope:
  - Homework & Assignments published in Google Doc
- Campuswire
  - All Q/A

About the Course: textbooks

- No textbooks are required
- We follow closely using “Machine Learning Refined” (Available on Github)
  - https://github.com/jermwatt/machine_learning_refined



About the Course: grading

- Attendance:
  - strong recommended. You are responsible for the classes you miss
- Grading
  - ~ 8 Homework Assignments: 55%
  - Midterm exam: 30%
  - Final Project: 15%
  - Grades will not be curved.

A: [90,100]
A-: [85,90)
B+: [80,85)
B: [75,80)
B-: [70,75)
C+: [65,70)
C: [60,65)
F: [0,60)

About the Course: grading (cont.)

- Deadlines are strict
  - Zero credits if you hand in an assignment even 1 min late
- Seven free late days
  - 1 late day = (0, 24h] late
  - 2 late days = (24h, 48h] late

Tip #1: Ask questions with contexts

I wrote the following code but the code doesn't run.
Can you tell me why?

```
1   import pefile
2   import os
3   API_list = []
4   PATH = "C:\Malware_Samples"
5   for FILE in os.listdir(PATH)
6    current_file =os.path.join(PATH, FILE)
7    pe= pefile.PE(current_file)
8    for entry in pe.DIRECTORY_ENTRY_IMPORT:
9       for API in entry.imports:
10       API_LIST.append(API.name)
11  ...
```

I wrote the following code but the code doesn't run.

[Here is your code]

I tried the following methods ….

[Here is what you have tried]

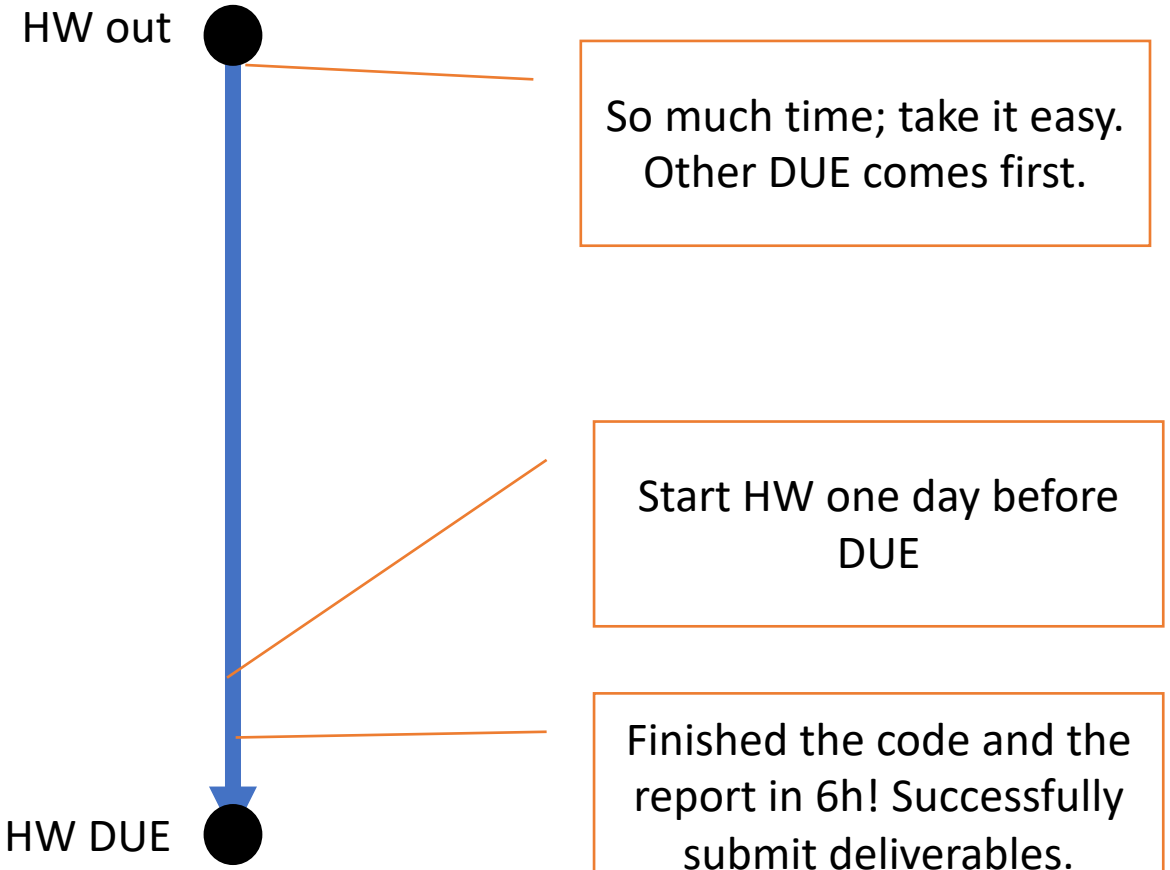
I got the following error ….

[Tell us what went wrong]

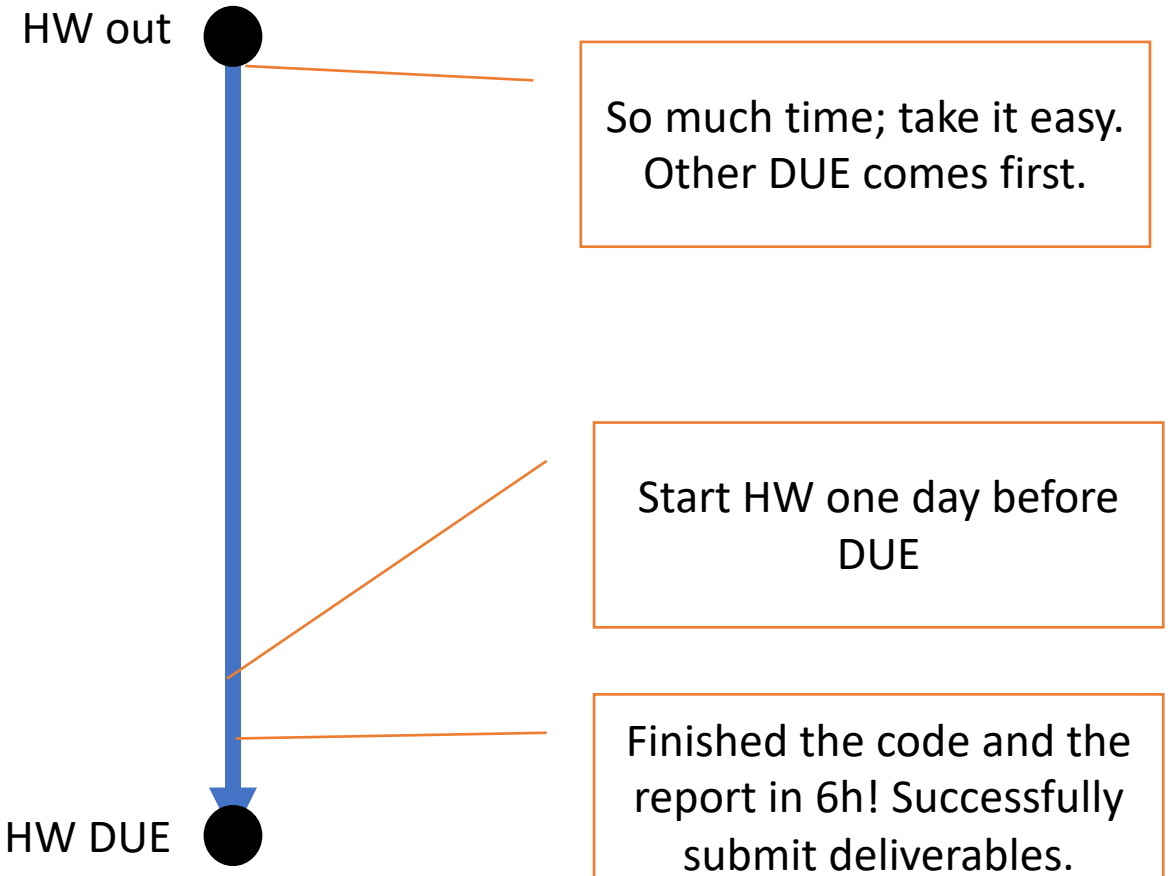
Tip #2: Start early!

What you expect

What can happen



Tip #2: Start early!



What can happen

- I need clarifications on the HW description
- Configuring Python Env takes too much time
- I am not familiar with Numpy/Matplotlib APIs …
- My program has BUGs… and I don’t know how to DEBUG
- The results do not look right…
- Wow… no time left to write the report…
- …..

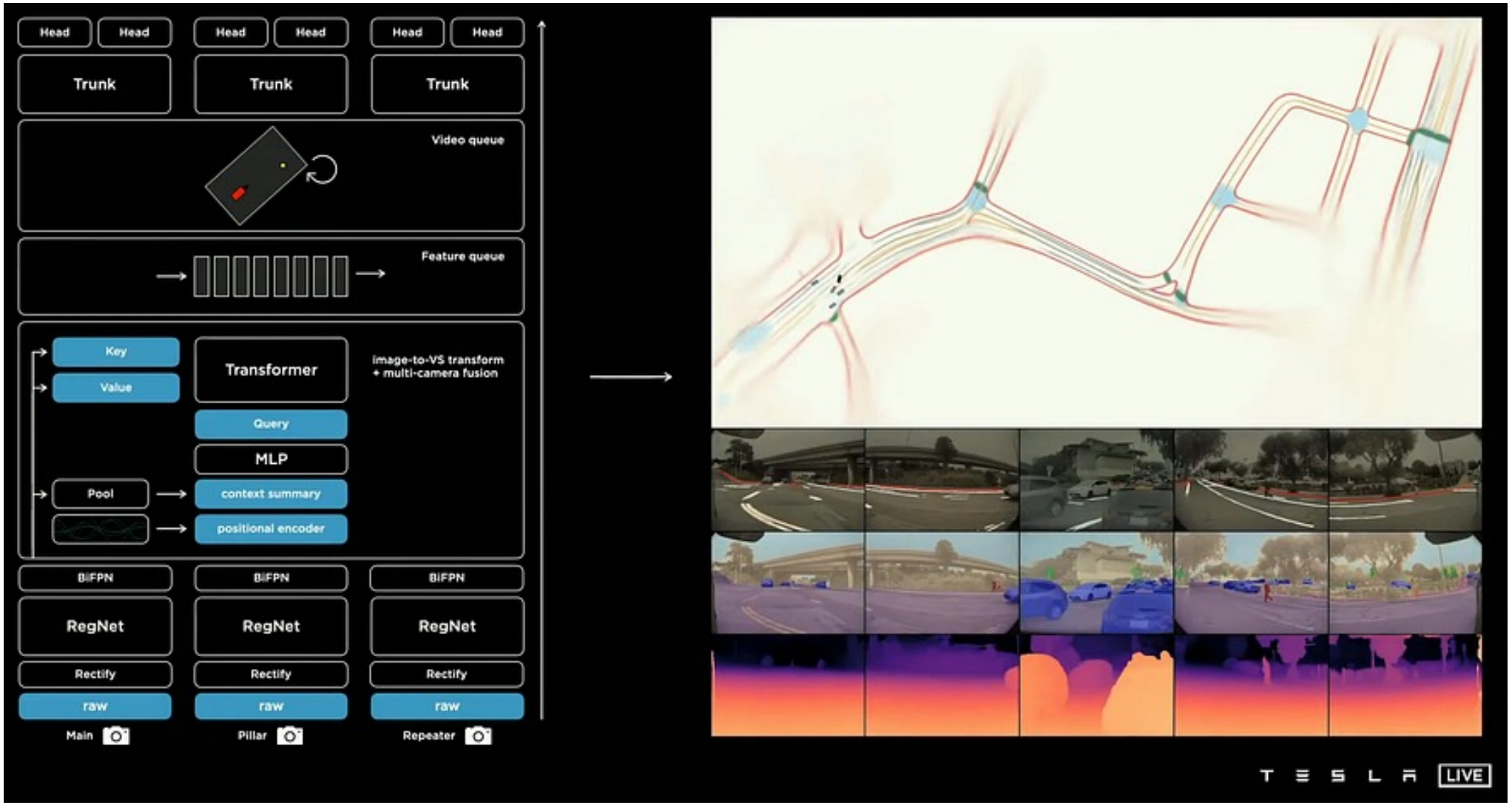
Tip #3: Pay attention to red-colored concepts

- The problem of determining the smallest (or largest) value a function can take, referred to as its *global minimum* (or *global maximum*) is a centuries old pursuit that has numerous applications throughout the sciences and engineering.
- In this Module we begin our investigation of mathematical optimization by describing the *zero order optimization* techniques (also called *derivative-free optimization*)

Q: What is derivation-free optimization?
Q: What is the difference between zero-order optimization vs. first-order optimizations?

Machine Learning Applications

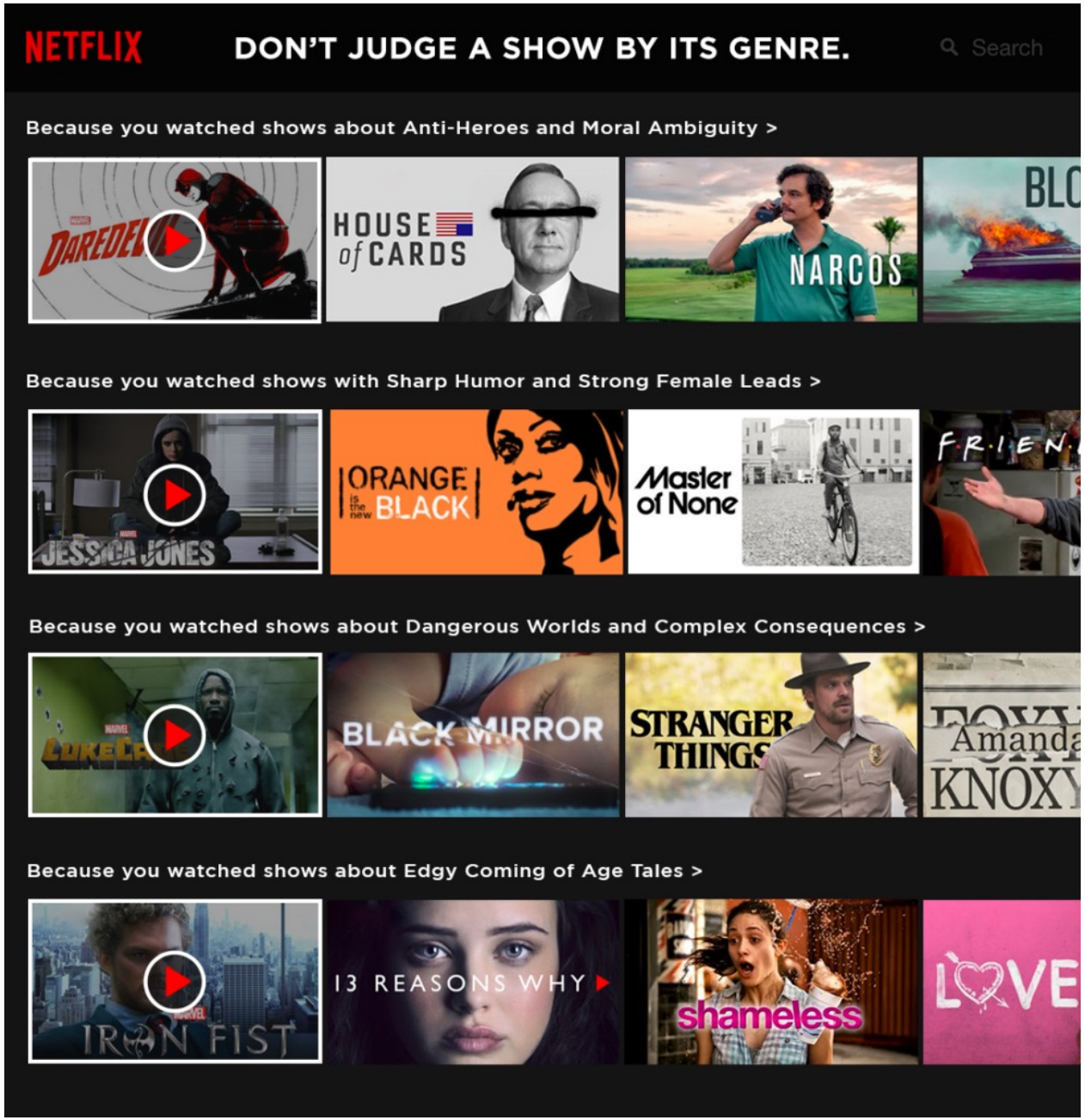
Automatic detection of objects in images -- a crucial component of driver-assisted and self-driving cars



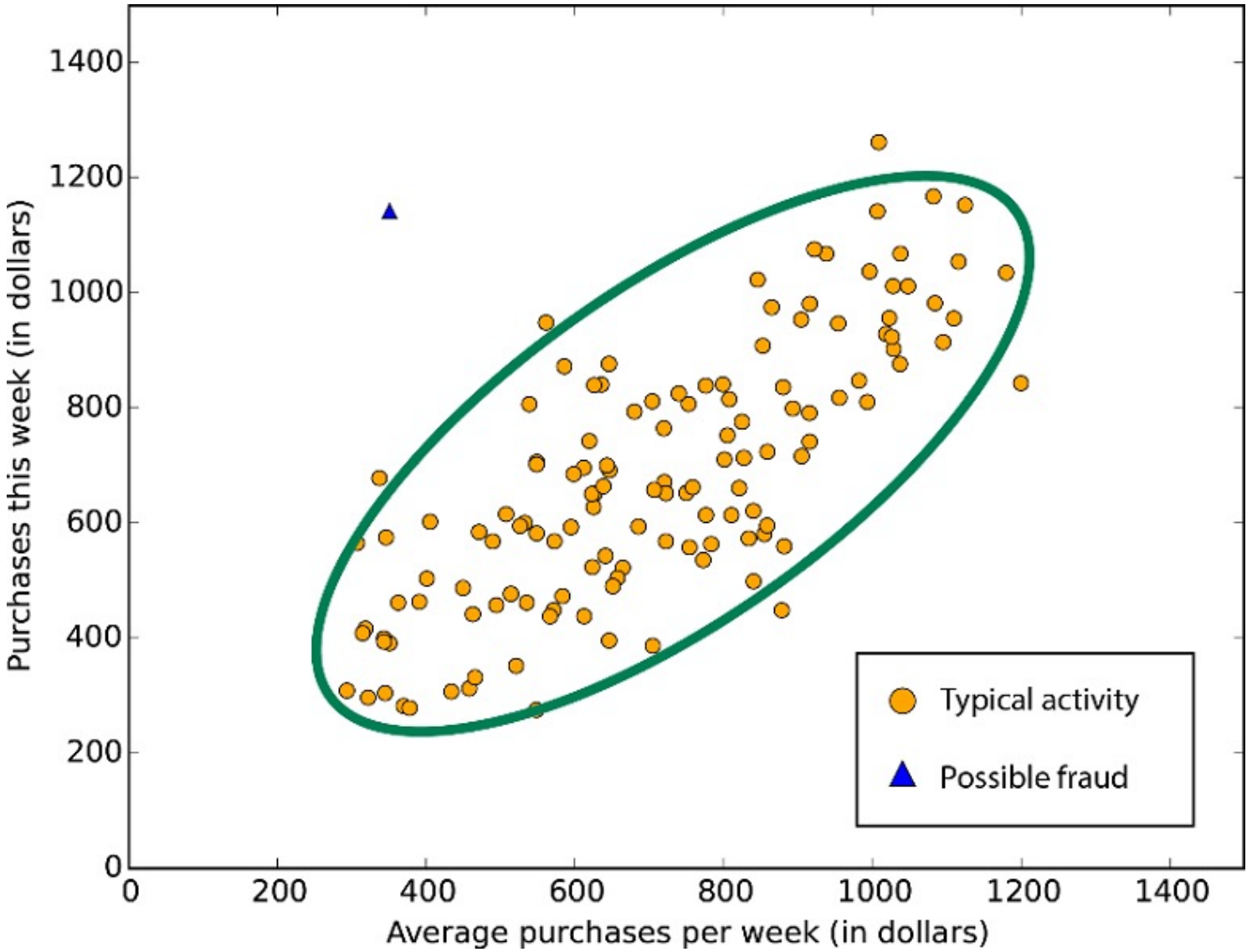
https://www.youtube.com/watch?v=j0z4FweCy4M

Speech recognition (which powers voice command technology)

Recommendation systems



Fraud Detection



ChatGPT – Large Language Models

Write a short two-paragraph poem about why chocolate is objectively the best food ever.

Chocolate, oh chocolate, the taste of pure bliss,
A treat that's indulgent, with every bite, a kiss.
From its rich and creamy texture, to its velvety smooth,
It's a flavor that's simply unmatched, a timeless proof.
With notes of sweet and bitter, a balance that's divine,
Chocolate is the food of gods, it's simply so fine.

With a history that dates back, to the ancient Mayans,
Its popularity endures, across cultures and spans.
From a simple bar to elaborate desserts so grand,
Chocolate always steals the show, it's simply unparalleled.
So let's celebrate this food, that brings so much joy,
Chocolate is the best, there's simply no other choice, oh boy!

Image generation: realistic image creation

"Teddy bears working on new AI research on the moon in the 1980s"

A high-level summary of machine learning

Machine Learning Pipeline

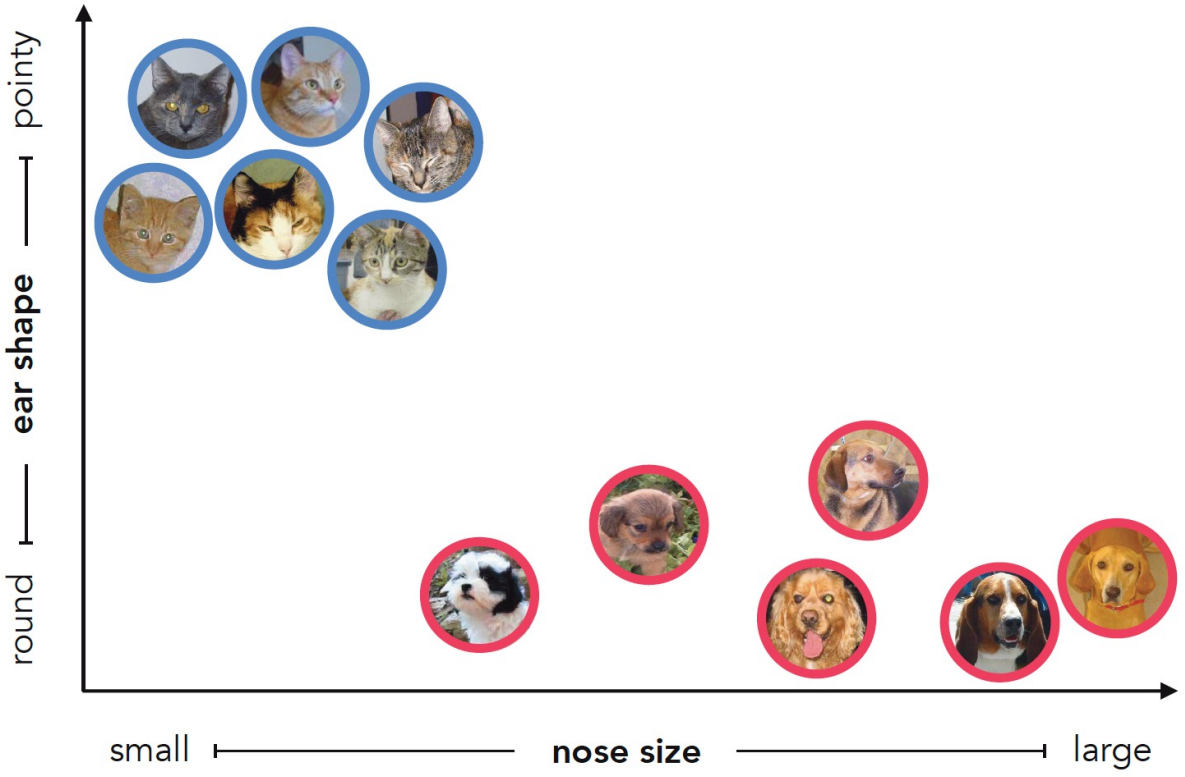
Example: teach a computer how to distinguish between pictures of cats from those with dogs

1. Data Collection

- Collect a training set of data

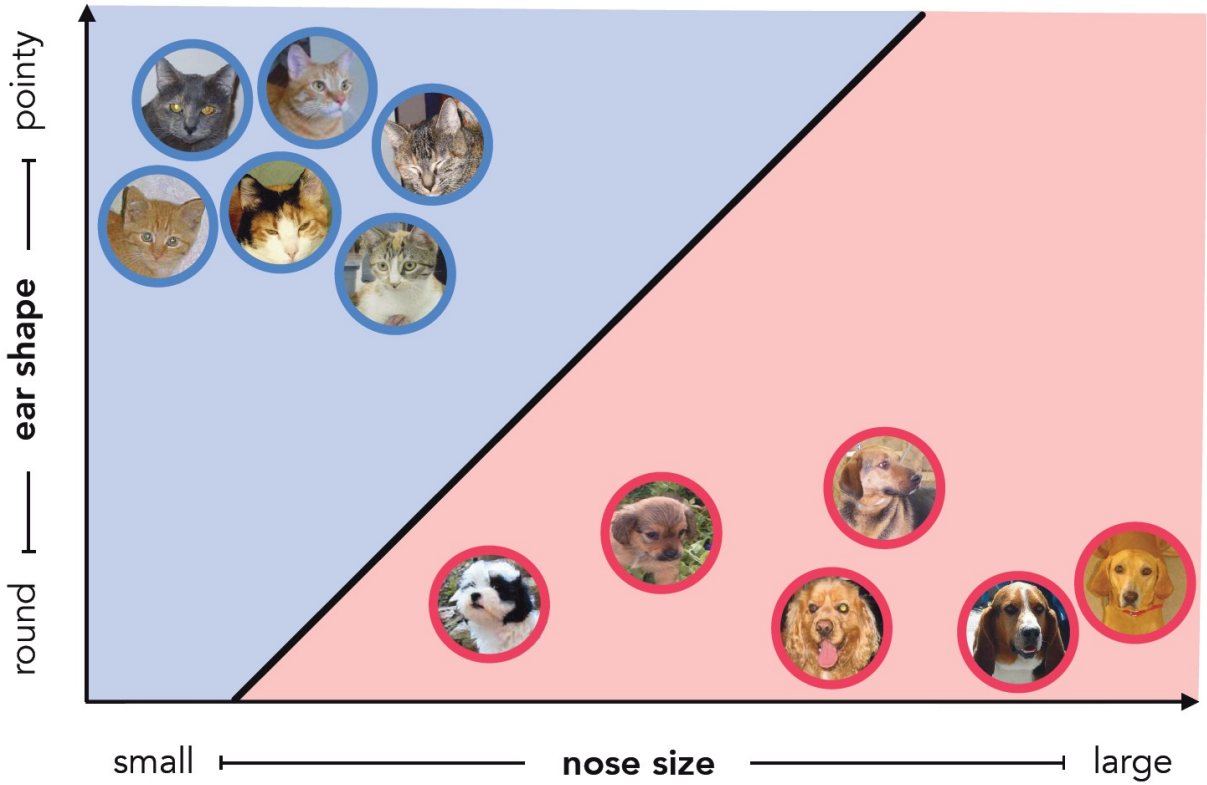
2. Feature Design

- We use color, size, the shape of the ears or nose, and/or some combination of these features in order to distinguish between the two.



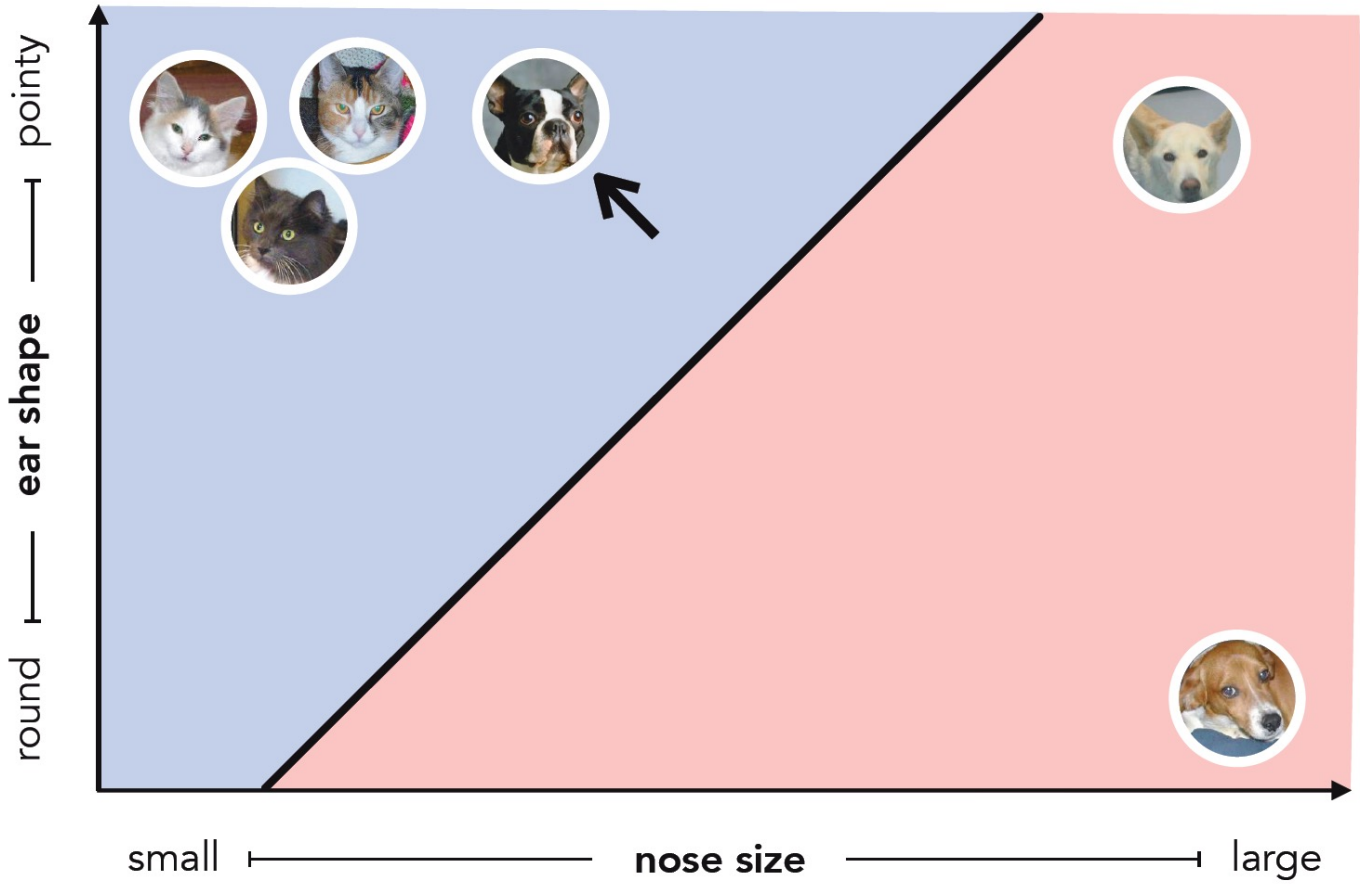
3. Model Training

- Train a machine learning (e.g., a linear classification model) to divides the feature space into cat and dog regions.

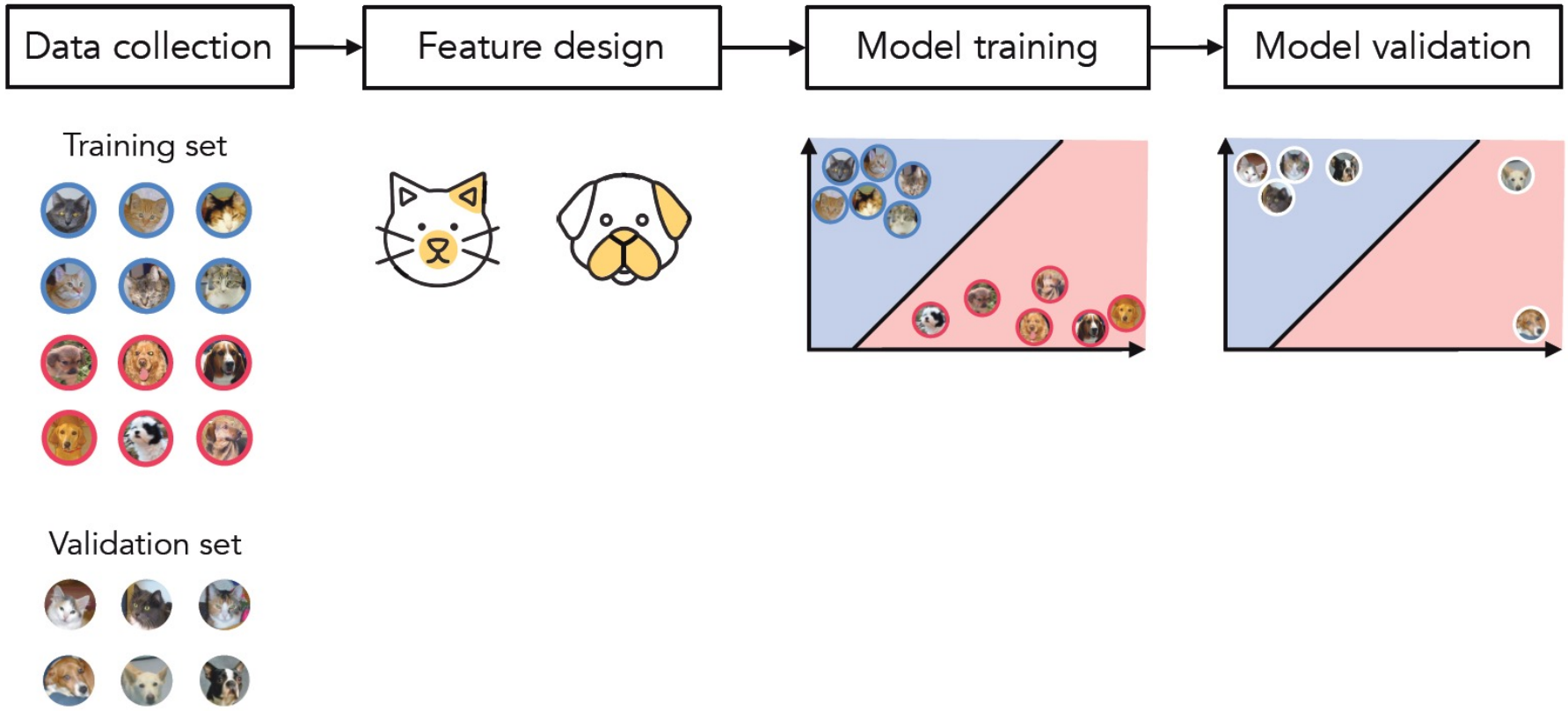


4. Model Validation

- Show the computer a batch of previously unseen images of cats and dogs (a validation set of data)



The pipeline of our cat-vs-dog problem

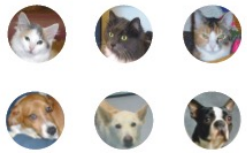


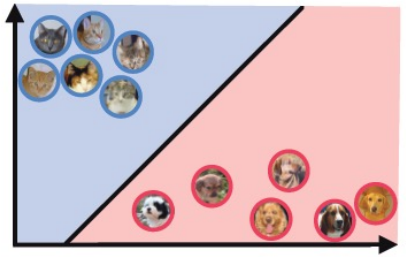
Mathematical Optimization

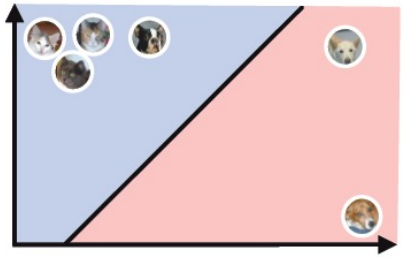
Data collection → Feature design → Model training → Model validation

Training set

Validation set

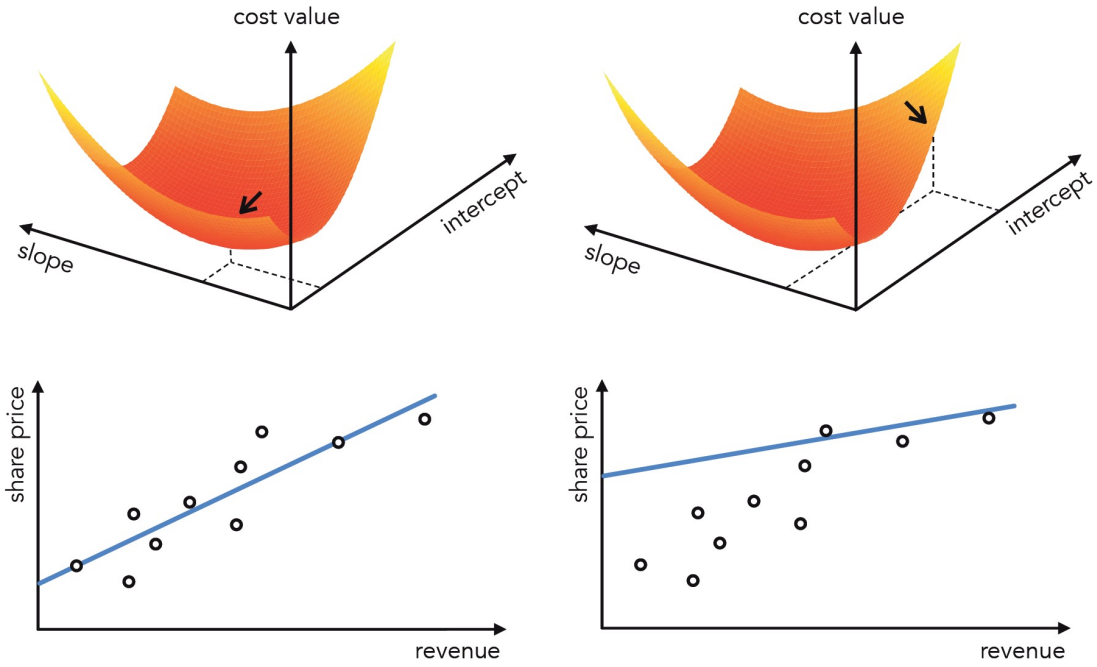






Mathematical Optimization

- Search for parameters of a learning model by optimizing a cost function

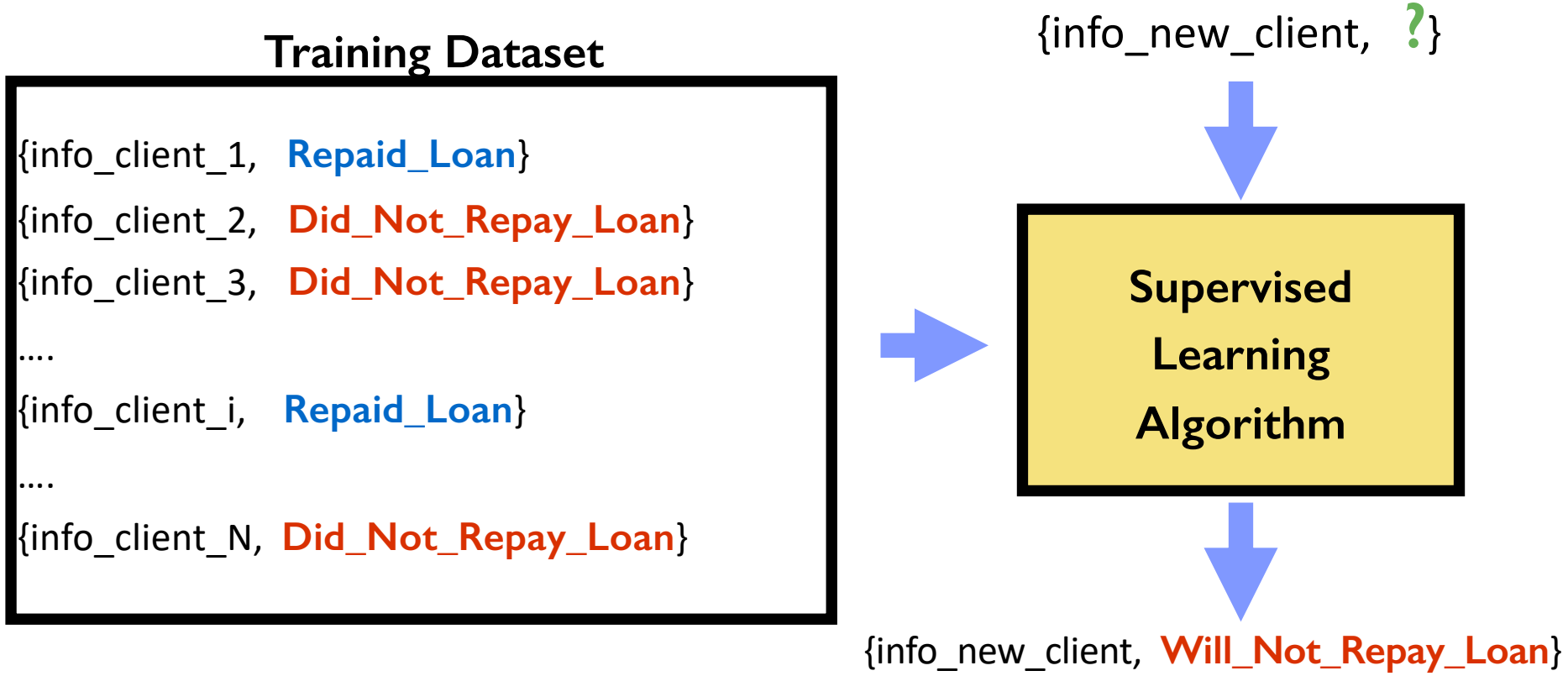


Machine Learning Approaches

- Supervised learning
  - Regression
  - Classification
- Unsupervised learning
  - Dimension reduction
  - Clustering

Supervised Learning

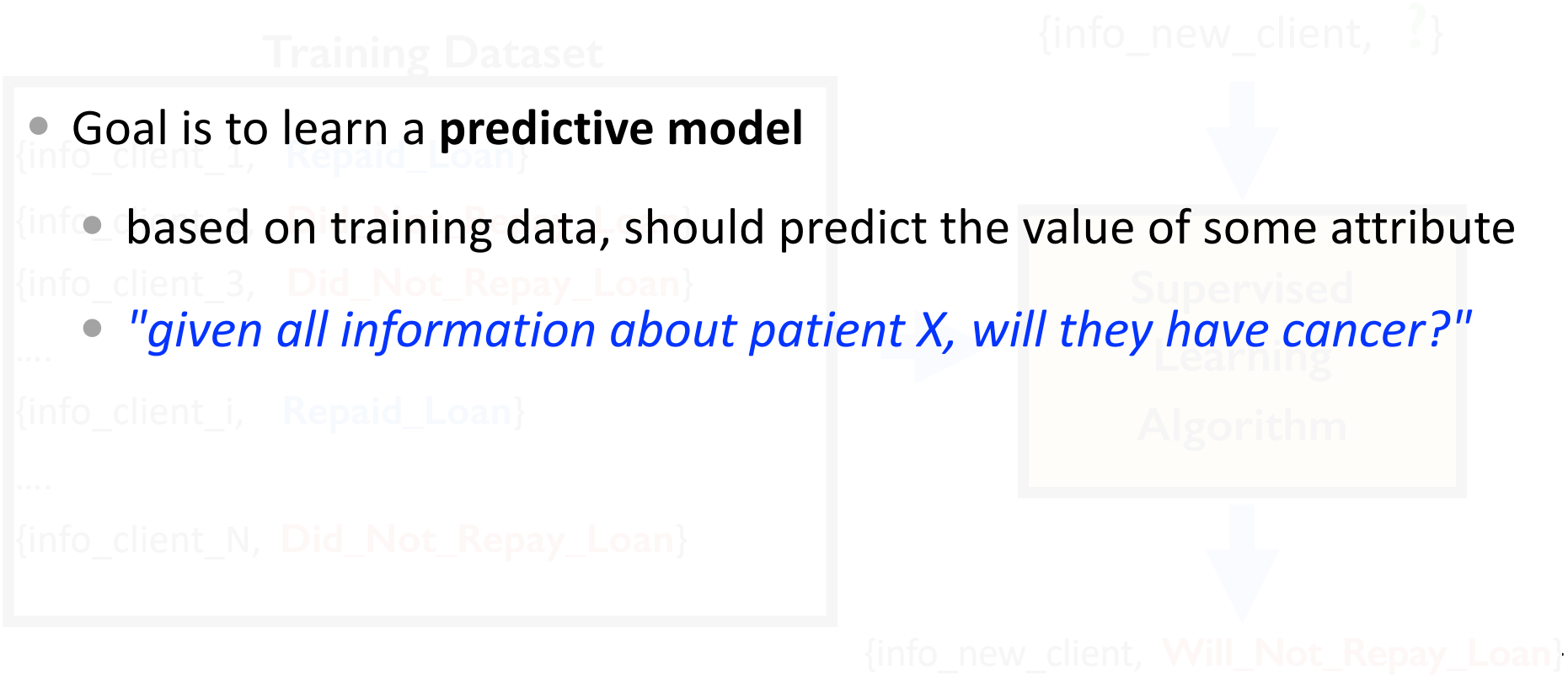
A *dataset* containing *labeled training examples* is given to the algorithm



Supervised Learning

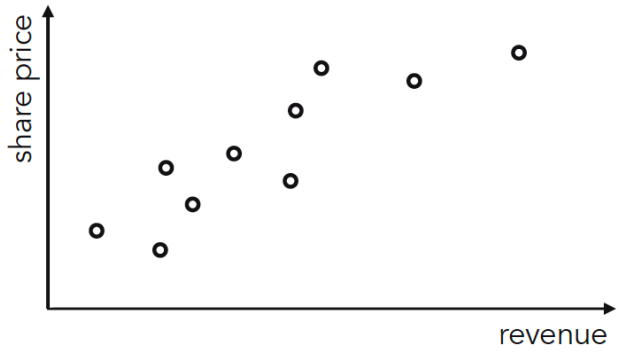
A *dataset* containing *labeled training examples* is given to the algorithm

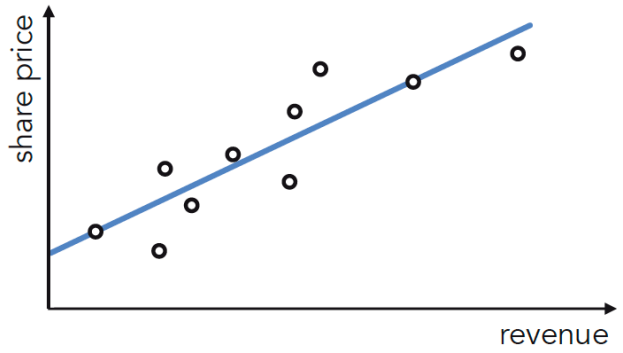
- Goal is to learn a **predictive model**
  - based on training data, should predict the value of some attribute
    - *"given all information about patient X, will they have cancer?"*

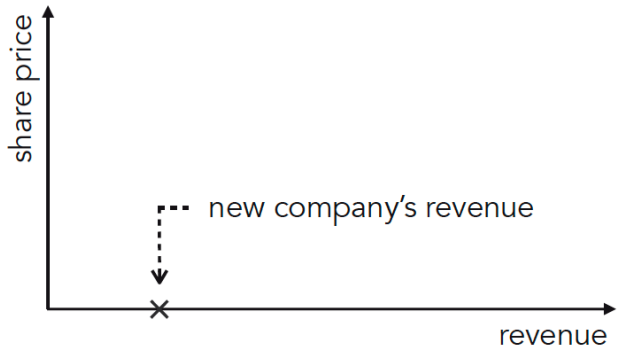


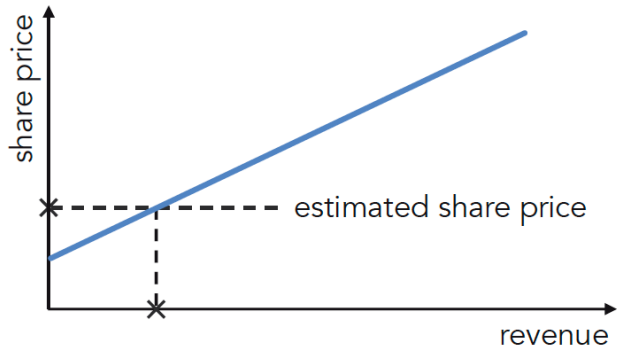
Regression

- *Predict a continuously-valued output*
- *Predict the share price of a company that is about to go public.*



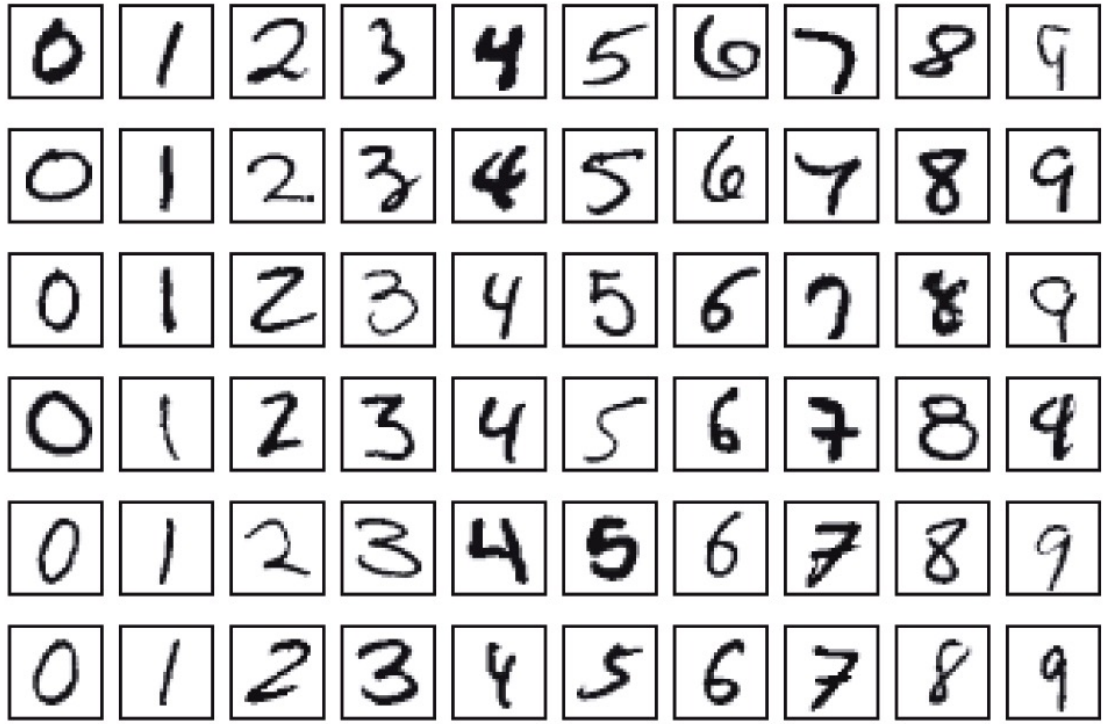






Classification

- Predict discrete values or classes
- e.g., recognize handwritten digits



Unsupervised Learning

A *dataset* containing *unlabeled data* is given to the algorithm

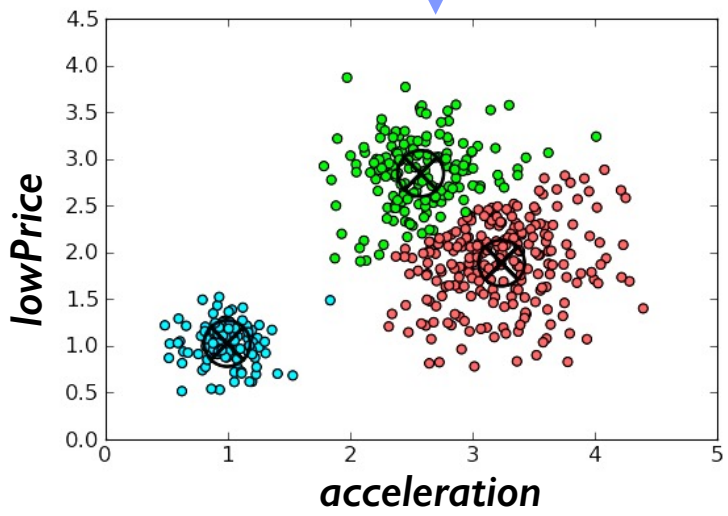
(no "correct/expected" answer associated with each example)

Training Dataset

{fuelEconomy=1, lowPrice=3, comfort=5, acceleration=1}
{fuelEconomy=3, lowPrice=5, comfort=3, acceleration=3}
{fuelEconomy=5, lowPrice=5, comfort=2, acceleration=4}
{fuelEconomy=2, lowPrice=2, comfort=1, acceleration=4}
{fuelEconomy=1, lowPrice=3, comfort=2, acceleration=2}

(features buyers said were important when making a decision

Unsupervised Learning Algorithm



"luxury car buyers"

"car buyers interested in performance/speed"

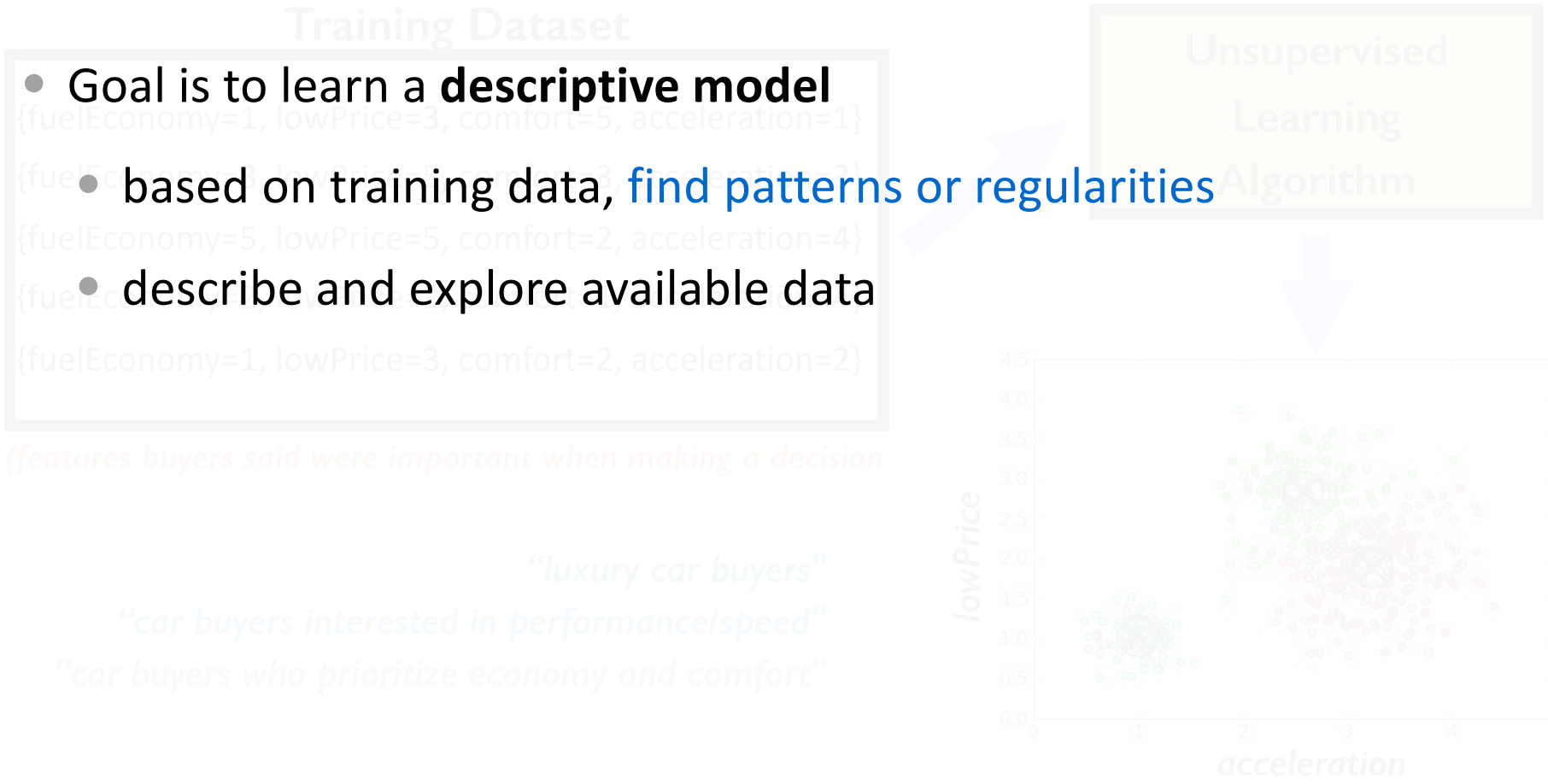
"car buyers who prioritize economy and comfort"

Unsupervised Learning

A *dataset* containing *unlabeled data* is given to the algorithm

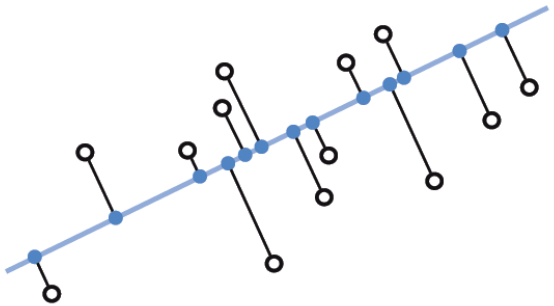
(no "correct/expected" answer associated with each example)

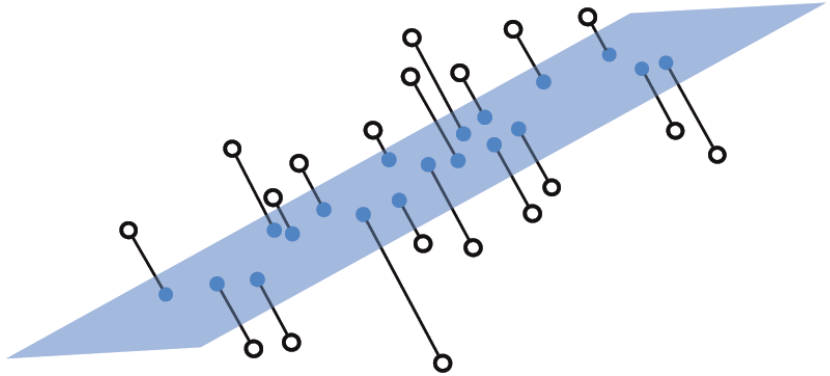
- Goal is to learn a **descriptive model**
  - based on training data, find patterns or regularities
  - describe and explore available data



Dimension Reduction

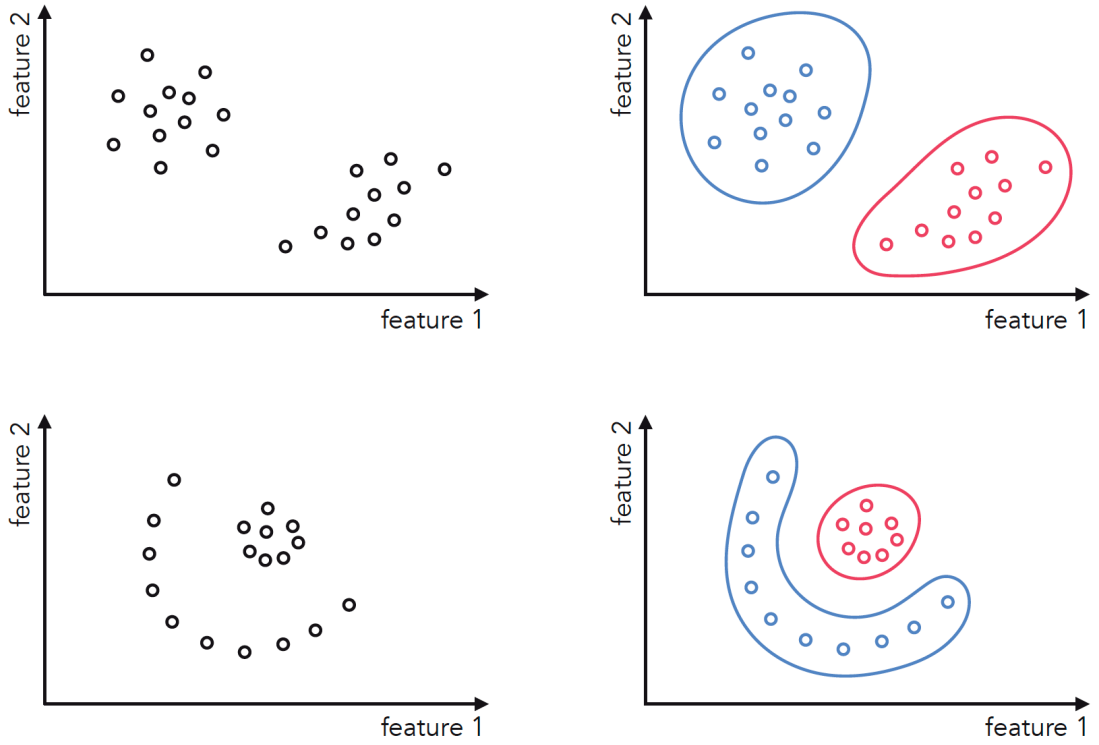
- Reduce the ambient dimension of input data





Clustering

- Determine a small number of representatives that adequately describe the diversity of a larger set of data

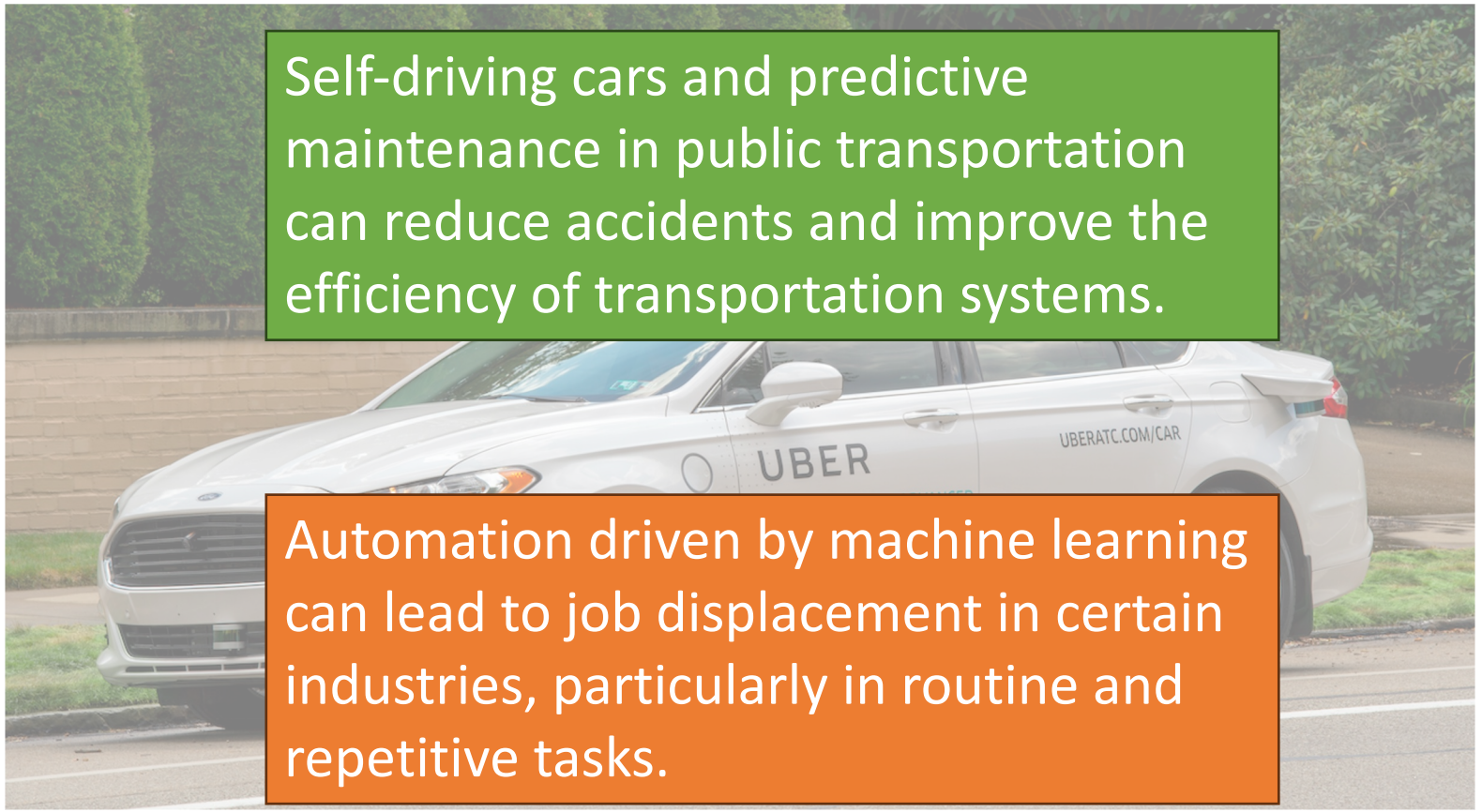


Machine Learning Impacts

Example 1: Self-driving cars

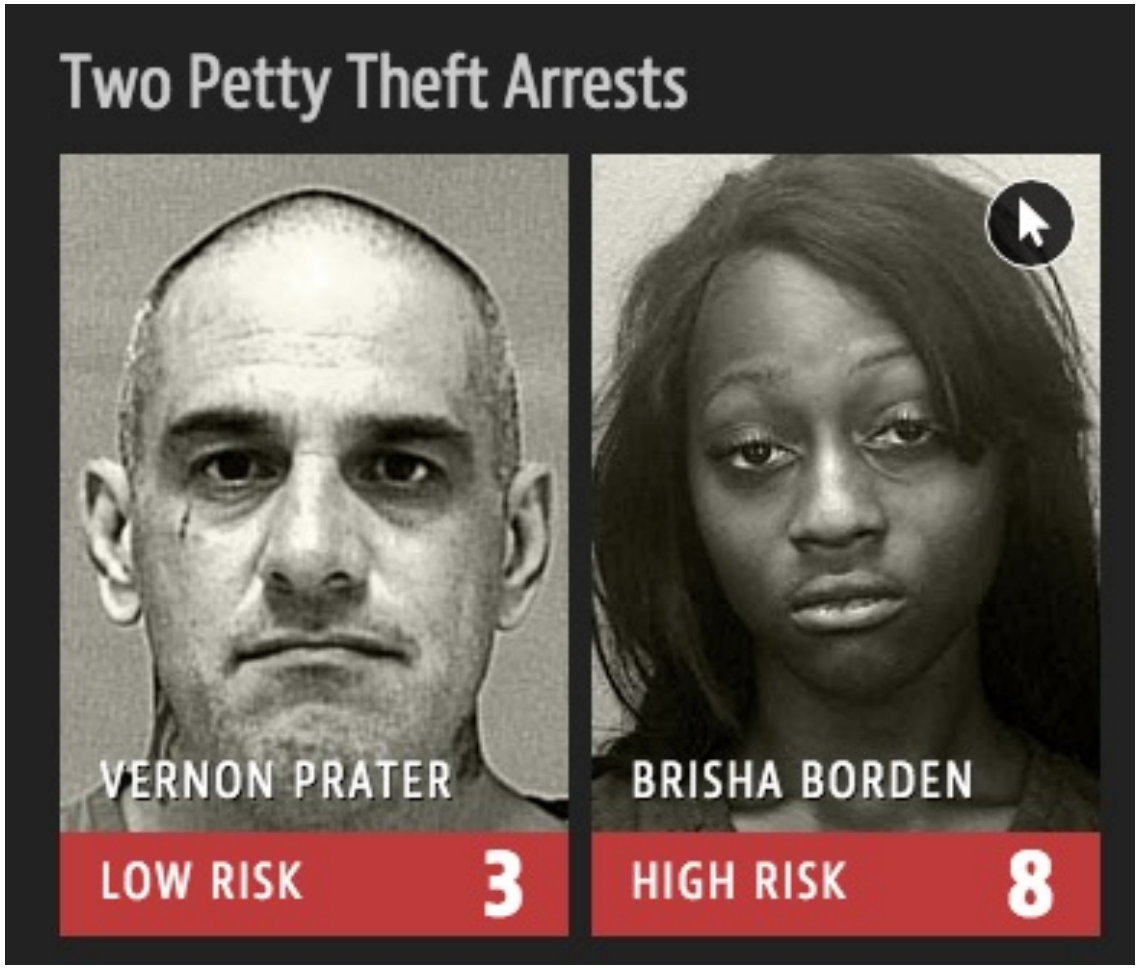
Pros and Cons?

Example 1: Self-driving cars

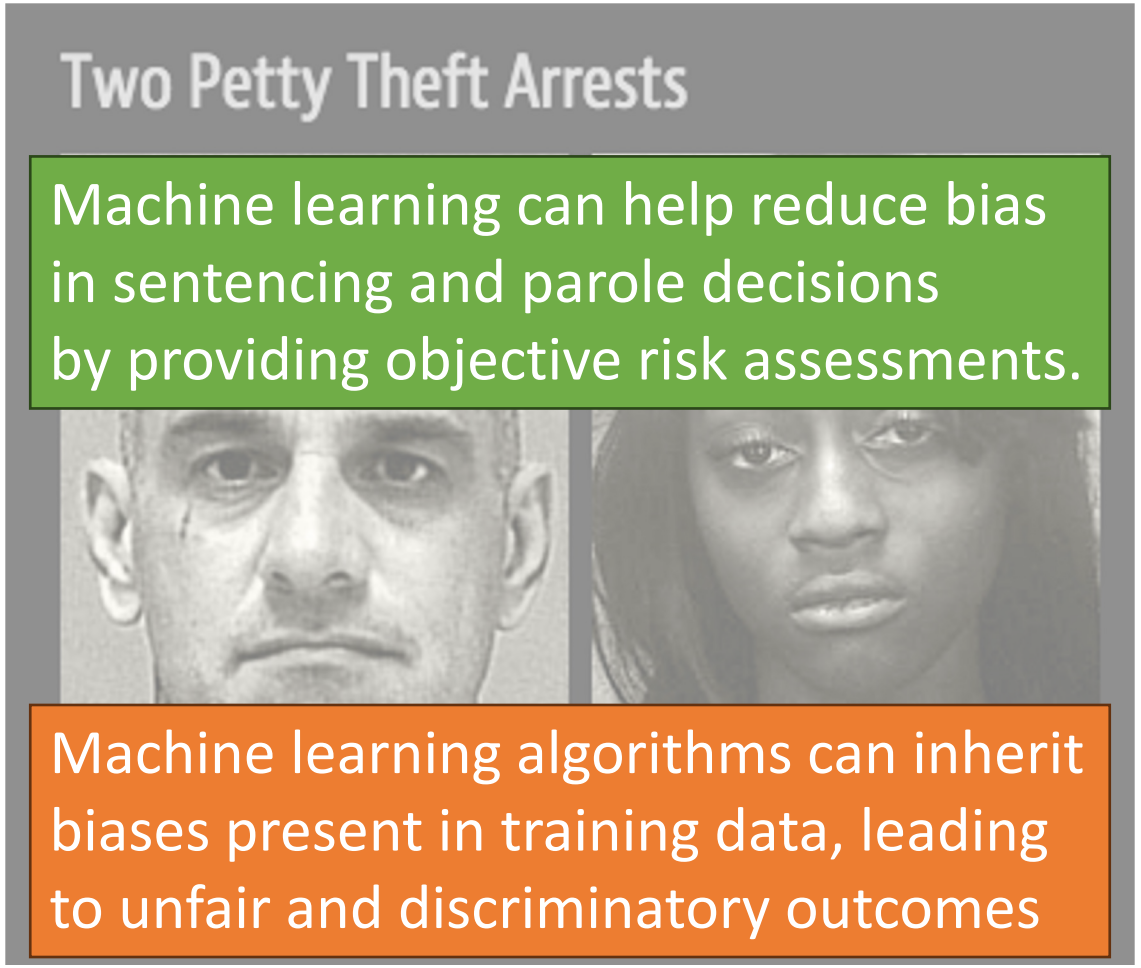


Pros and Cons?

Example 2: Crime prediction



Example 2: Crime prediction



Can you think of more examples?