https://courses.cs.washington.edu/courses/cse446/20sp

Machine Learning CSE446/546

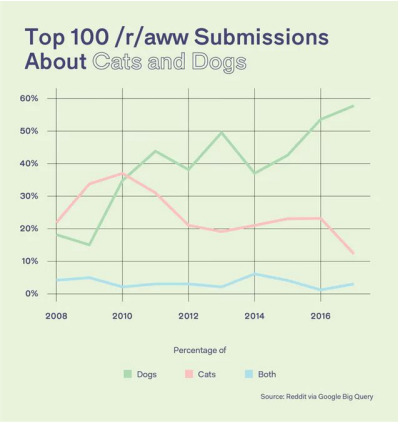
Kevin Jamieson and Jamie Morgenstern
University of Washington

March 30, 2020

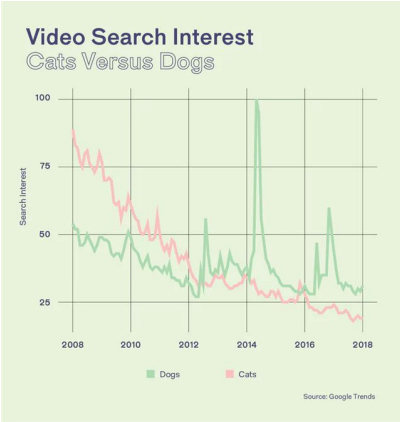
Traditional algorithms

Social media mentions of Cats vs. Dogs

Reddit



Google



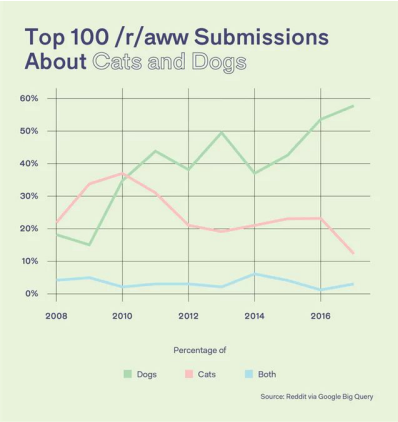
Twitter?

Graphics courtesy of https://theoutline.com/post/3128/dogs-cats-internet-popularity?zd=1

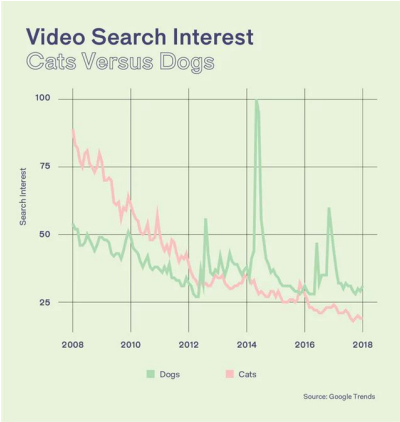
Traditional algorithms

Social media mentions of Cats vs. Dogs

Reddit



Google



Twitter?

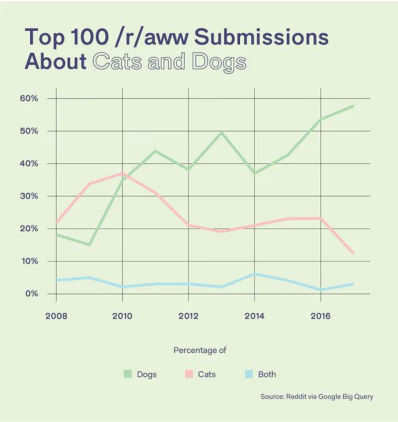
Write a program that sorts tweets into those containing "**cat**", "**dog**", or ***other***

Graphics courtesy of https://theoutline.com/post/3128/dogs-cats-internet-popularity?zd=1

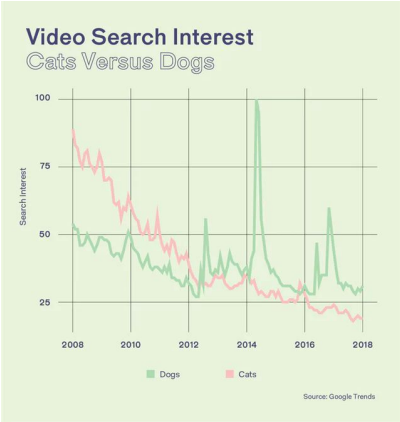
Traditional algorithms

Social media mentions of Cats vs. Dogs

Reddit



Google



Twitter?

```
cats = []
dogs = []
other = []
for tweet in tweets:
    if "cat" in tweet:
        cats.append(tweet)
    elseif "dog" in tweet:
        dogs.append(tweet)
    else:
        other.append(tweet)
return cats, dogs, other
```

Write a program that sorts tweets into those containing "**cat**", "**dog**", or ***other***

Graphics courtesy of https://theoutline.com/post/3128/dogs-cats-internet-popularity?zd=1

Machine learning algorithms

Write a program that sorts images into those containing “**birds**”, “**airplanes**”, or ***other.***

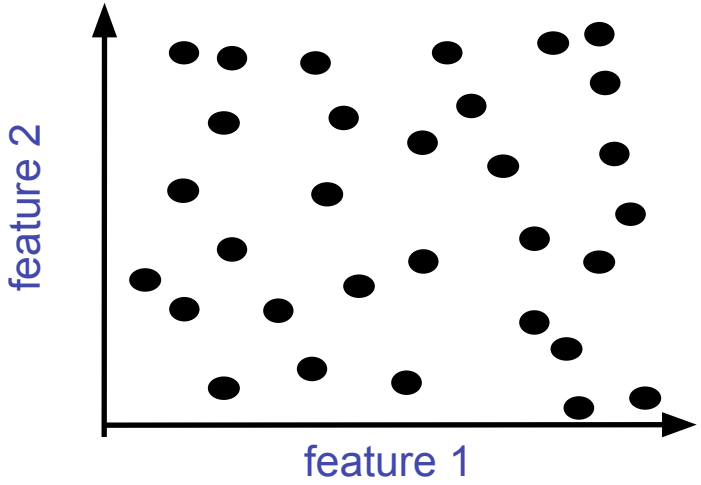
Machine learning algorithms

Write a program that sorts image
into those containing "**birds**",
"**airplanes**", or ***other.***

```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

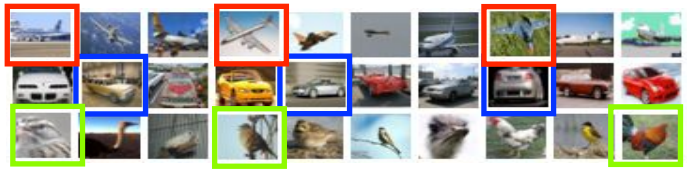
Write a program that sorts image
into those containing "**birds**",
"**airplanes**", or ***other.***

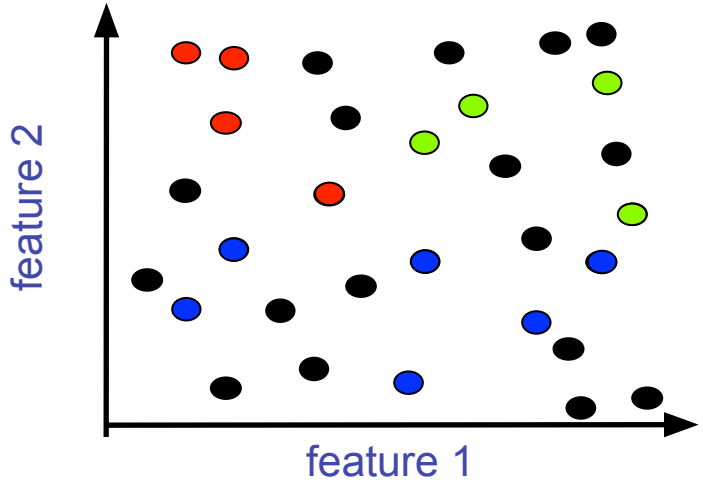



```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

Write a program that sorts image
into those containing "**birds**",
"**airplanes**", or ***other.***

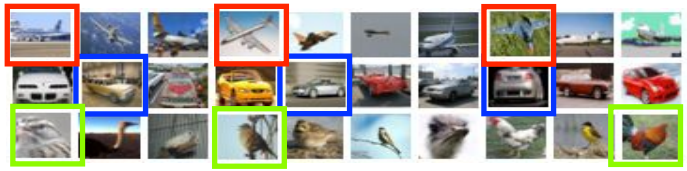


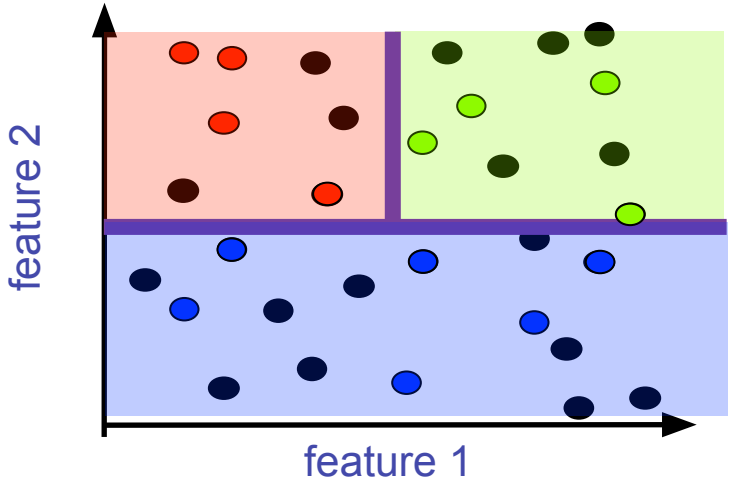



```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

Write a program that sorts image
into those containing "**birds**",
"**airplanes**", or ***other.***

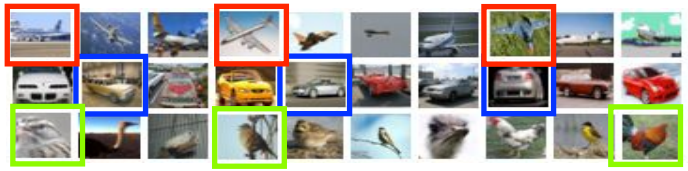


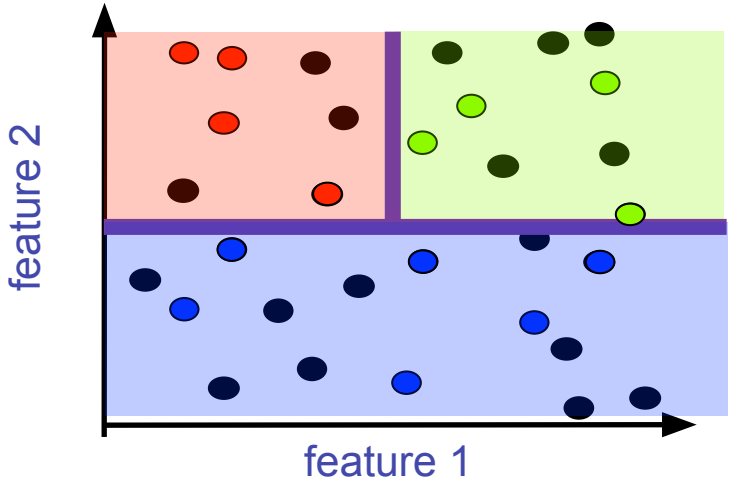



```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

Write a program that sorts image
into those containing "**birds**",
"**airplanes**", or ***other.***






```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

The decision rule of

```
if "cat" in tweet:
```

is **hard coded by expert.**

The decision rule of

```
if bird in image:
```

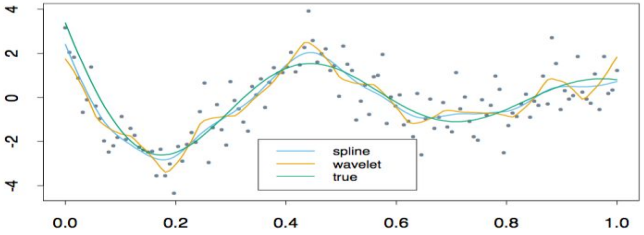
is **LEARNED using DATA**

Machine Learning Ingredients

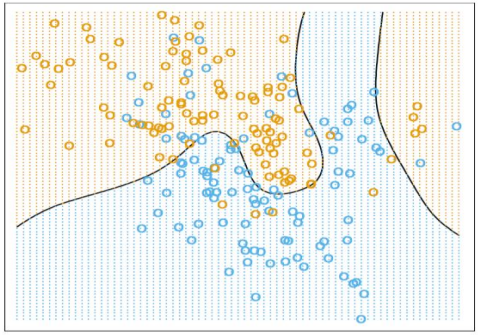
- **Data**: past observations
- **Hypotheses/Models**: devised to capture the patterns in data
- **Prediction**: apply model to forecast future observations

ML uses past data to make personalized predictions

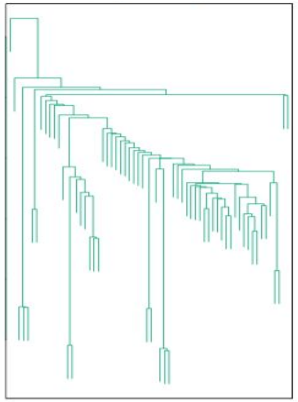
Flavors of ML



Predict continuous value:
ex: stock market, credit score, temperature, Netflix rating



Predict categorical value:
loan or not? spam or not? what disease is this?



Predict structure:
tree of life from DNA, find similar images, community detection, “trending” topics

Mix of statistics (theory) and algorithms (programming)

CSE446/546: Machine Learning

Lecture: Monday, Wednesday 9:00-10:20
Instructor: Kevin Jamieson and Jamie Morgenstern
Contact: cse446-staff@cs.washington.edu
Website: https://courses.cs.washington.edu/courses/cse446/20sp/

What this class is:

- **Fundamentals of ML:** bias/variance tradeoff, overfitting, optimization and computational tradeoffs, supervised learning (e.g., linear, boosting, deep learning), unsupervised models (e.g. k-means, EM, PCA)
- **Preparation for further learning:** the field is fast-moving, you will be able to apply the basics and teach yourself the latest

What this class is not:

- **Survey course:** laundry list of algorithms, how to win Kaggle
- **An easy course:** familiarity with intro linear algebra and probability are assumed, homework will be time-consuming

Prerequisites

- Formally:
  - MATH 126, CSE 312, MATH 308, STAT 390 or equivalent
- Familiarity with:
  - Linear algebra
    - linear dependence, rank, linear equations
  - Multivariate calculus
  - Probability and statistics
    - Distributions, densities, marginalization, moments
  - Algorithms
    - Basic data structures, complexity
- "Can I learn these topics concurrently?"
- Use HW0 to judge skills
- **See website for review materials!**

Which course again?

- 446 **undergraduate** versus 546 **graduate**
  - 446 is is the undergraduate version of machine learning
    - Only type “A” problems on the homeworks
  - 546 is the graduate version
    - Both type “A” and type “B” problems on the homework
      - Will be curved/weighted so that a perfect score on “A” problems (and no “B” problems) -> 3.8
      - For grades > 3.8, need to do “B” problems

Grading

- 5 homeworks
- Each contains both theoretical questions and will have programming
  - Collaboration okay. You must write, submit, and understand your answers and code (which we may run)
  - Do not Google for answers.
  - Do not share code.
- No midterm or final

Honor Code

- Collaboration okay.
  - Do not Google for answers.
  - Do not share code.
- **You must cite every collaborator and source you used/found for your homeworks**
- **Do not write up or code your homeworks together**
  - **Word-for-word or line-by-line similarities of solutions will be checked for and not be tolerated (0% on assignment for first offense, escalation for second offense)**
- **Do not ask for help on your homeworks on stackexchange/Q+A sites.**

Homeworks

- HW 0 is out (**Due next Wednesday Midnight**)
  - Short *review*
  - Work individually, treat as barometer for readiness
- HW 1,2,3,4
- They are not easy or short. Start early.
- Submit to Gradescope

1. All code must be written in Python
2. All written work must be typeset (e.g., LaTeX)

See course website for tutorials and references.

Communication Channels

- **Announcements, questions about class, homework help**
  - Piazza (get code on Canvas)
  - Section
  - Office hours (start tomorrow)
- **Regrade requests**
  - Directly to Gradescope
- **Personal concerns**
  - Email: cse446-staff@cs.washington.edu
- **Anonymous feedback**
  - See website for link

Office hours

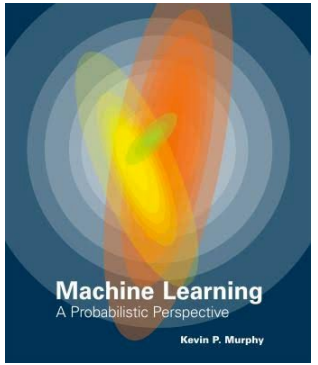
- 13 Experienced TAs, lots of office hours (see website)
- Romain Camilleri, Anirudh Canumalla, Eric Chan, An-Tsu Chen, Wenjun Chen, Yao Dou, Zian Fu, Joshua Gardner, Rishi Dev Jha, Sam Lee, Zhichao Lei, Zeyu Liu, Swati Padmanabhan, Tobias Rohde, Luxi Wang, Chengda Xu, Zhanhao Zhang
- Office hours are *not* recorded
- First half hour of office hours reserved for **A problems only,** second half can be A or B problems

Section

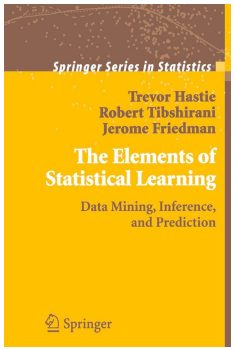
- **Section is intended for students in CSE446 only**
- Section is ***not*** recorded
- If you are in CSE546, you are welcome to attend section but please act as “a fly on the wall” rather than an active participant. We are trying to provide as many resources to the enrolled members of CSE446.

Text Books

- Required Textbook:
  - *Machine Learning: a Probabilistic Perspective*; **Kevin Murphy**



- Optional Books (free PDF):
  - *The Elements of Statistical Learning: Data Mining, Inference, and Prediction;* Trevor Hastie, Robert Tibshirani, Jerome Friedman



Auditing/Add code requests

- We will give out as many add codes to fulfill demand
- Add-code list for CSE 446: https://www.cs.washington.edu/academics/ugrad/courses/petition
- Add-code list for CSE 546: http://tinyurl.com/UWCSE5XX
- By FERPA guidelines, you can only attend Zoom lecture if you are officially registered for the course (you can officially audit)

Zoom etiquette

- Lecture, section, office hours all on zoom (see Canvas for links)
- Please mute yourself when not speaking
- Please keep your video on as much as possible
- To ask a question, in the Zoom chat window either
  - Type your question
  - Type “hand” and we will call on you (preferred)
    - Please do not use Zoom’s “raise hand” feature, it doesn’t track ordering

Virtual Course

Teaching a course virtually is new to everyone, including the instructors. We will do our very best. But expect (on both ends):

- Hiccups in lecture including audio/video drop, yelling kids or dogs
- Potentially different instructions to watch lecture (different zoom links, something new, etc.)
- We will periodically send out a feedback form to find ways to improve your experience.

Privacy

- For correspondences regarding grades, health, or any other personal matters, email the instructors directly. Do **not** use Piazza or lecture/section for these matters
- Lecture is ***recorded*.** Not public, but be aware.
- Office hours are ***not recorded***
- Section is ***not recorded*** but materials online.

Enjoy!

- ML is becoming ubiquitous in science, engineering and beyond
- It's one of the hottest topics in industry today
- This class should give you the basic foundation for applying ML and developing new methods
- The fun begins…