About us

• Me: Geoff Gordon

• TAs:
  ▶ Aditya Paul
  ▶ Aishwarya Jadhav
  ▶ Xiaoyu Xu
Notes and reminders

• Location: CUC McKenna (when in person)

• Most weeks: two lectures, one “other”
  ▸ This week: virtual-only lectures today and Wednesday
  ▸ Lab 0 (optional) on Friday: Python review

• https://www.cs.cmu.edu/~ggordon/10606s22/syllabus-and-lecture-outline.html

• Ask questions, participate, help one another learn! https://xkcd.com/1053/
What is ML?

**Speech Recognition**
1. Learning to recognize spoken words

   **THEN**
   "...the SPHNN system (e.g. Lee, 1985) learns speaker-specific strategies for recognizing the attributes sounds (phonemes) and words from the observed speech signal... neural network methods... hidden Markov models..."

   (Mitchell, 1997)

   **NOW**
   Source: https://www.docstoccdn.com/blog/resources/learning/19222849.png

**Robotics**
2. Learning to drive an autonomous vehicle

   **THEN**
   "...the ALVINN system (Romaniuk, 1989) has used he learned strategies to drive standalone at 60 miles per hour in 96 miles on public highways among other cars..."

   (Mitchell, 1997)

   **NOW**
   waymo.com

**Games / Reasoning**
3. Learning to beat the masters at board games

   **THEN**
   "...the world’s top computer program for backgammon, TD-AMMON (Tempea, 1992, 1995), learned its strategy by playing over one trillion practice games against itself..."

   (Mitchell, 1997)

**Computer Vision**
4. Learning to recognize images

   **THEN**
   "...the recognizer is a convolution network that can be spatially replicated. From the network itself, a hidden Markov model produces word scores. The entire system is globally trained to minimize word-level errors..."

   (LeCun et al., 1990)

   **NOW**

   ![Image](https://example.com/image.png)

**Learning Theory**
5. In what cases and how well can we learn?

   1. How many examples do we need to learn?
   2. How do we quantify our ability to generalize to unseen data?
   3. Which algorithms are better suited to specific learning settings?

credit: Matt Gormley
Why this course?

Dual SVM – linearly separable case

- Dual problem (derivation):
  \[
  \max_{\alpha_j \geq 0} \sum_j d(\alpha_j) \quad \text{subject to} \quad \sum_j y_j \alpha_j = 0
  \]
  \[
  \Rightarrow w = \sum_j \alpha_j y_j x_j
  \]
  \[
  \Rightarrow \sum_j \alpha_j y_j = 0
  \]

\[
L(w, b, \alpha) = \frac{1}{2} w \cdot w - \sum_j \alpha_j \left( (w \cdot x_j + b) y_j - 1 \right)
\]
\[
\alpha_j \geq 0, \ \forall j
\]
\[
\alpha - \text{weights on training pts (n-dim problem)}
\]

credit: Aarti Singh (10-601)
This course is a bit odd

• Most courses:
  ▶ teach one semester’s worth of material in one semester
  ▶ go over enough examples of every topic that you can learn it from scratch with no trouble

• This course:
  ▶ several semester’s worth of material in one!
  ▶ assumption: you’ve seen at least some of it before, need to work on rest
  ▶ not enough examples on any one topic to learn from scratch: you must ask questions and seek out your own material

• Benefit: we cover a lot of ground, help build a strong base for ML courses
Formal systems

Objects & Data type

∩ API
Optimization

\[
\min_w L(w)
\]
Optimization from data

\[ P(x_i \mid w) \]

\[ P(w \mid X) \propto P(x \mid w) \frac{P(w)}{P(x)} \]

\[ L(w) \]

\[ \max_w P(w \mid X) \]

\[ P(y_i \mid x_i, w) \]

\[ P(w \mid x, y) \propto P(y \mid x, w) P(w \mid x) \]

\[ P(y \mid x, w) \]

\[ P(y_2 \mid x_2, w) \]

\[ \ldots P(y_T \mid x_T, w) \]
Example: classification

Classifier:
input $x \in \mathbb{R}^n$
output $y \in \{-1, 1\}$

$$L(w) = \prod_{i} P(y_i \mid x_i, w)$$

$$\begin{cases} x_1 \rightarrow \mathbb{R}^n \\ x_2 \rightarrow \mathbb{R} \\ y_i \rightarrow \pm 1 \end{cases}$$
Example: density estimation

\[ P(\omega) \propto \prod P(x_i | \omega) \]
Optimize: tries to follow steps of decreasing $L(w)$.
Gradient descent

Start at \( w^{(0)} \)

Repeat \( i = 1, 2, \ldots \)

\[
\begin{align*}
    w^{(i)} &= w^{(i-1)} - \eta \nabla L(w^{(i-1)}) \\
    \eta^{(i)} &= \frac{\eta^{(i-1)}}{2}
\end{align*}
\]

\( \eta^{(i)} \in \mathbb{R} \)

\( \eta^{(i)} > 0 \)
if $w \cdot x > 1$
predict $+1$

if $w \cdot x < 1$
predict $-1$

decision boundary:

$\exists x \mid w \cdot x = 1$

if $w$ is over here

we get this example right:

$\exists w \mid w \cdot x > 1$
Example

go to repl.it