CSE 446: Machine Learning

Sewoong Oh

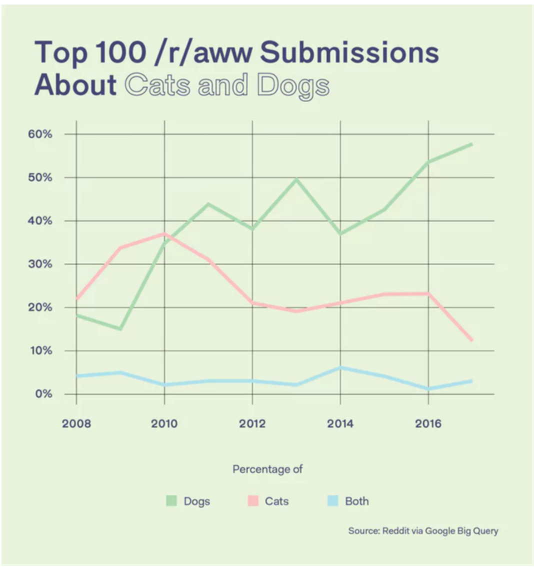
Traditional algorithms

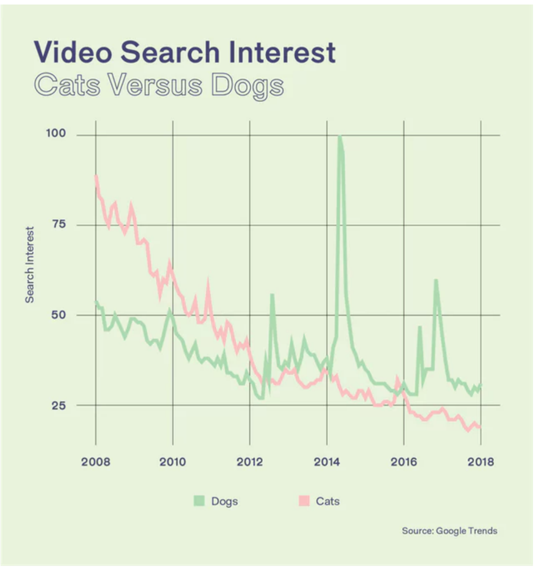
Social media mentions of Cats vs. Dogs

Reddit

Google

Twitter?





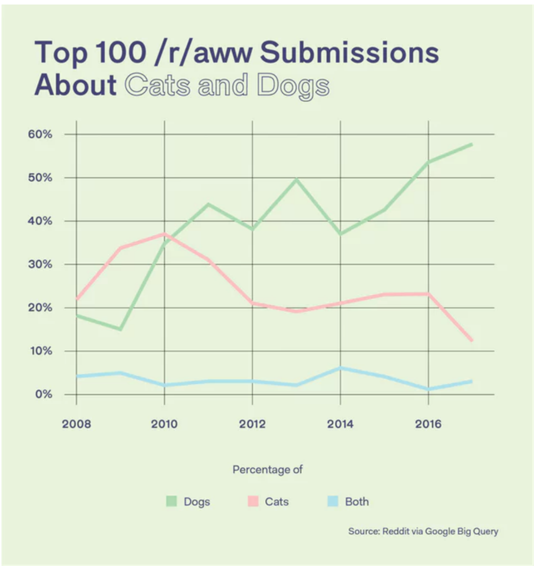
Write a program that sorts tweets into those containing "**cat**", "**dog**", or ***other***

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

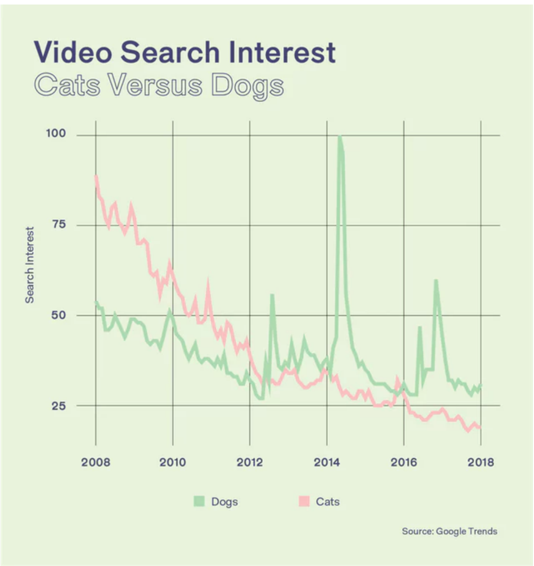
Traditional algorithms

Social media mentions of Cats vs. Dogs

Reddit



Google



Twitter?

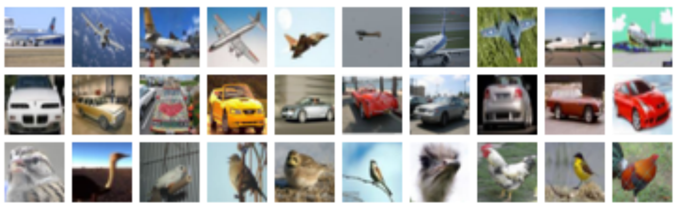
```
cats = []
dogs = []
other = []
for tweet in tweets:
    if "cat" in tweet:
        cats.append(tweet)
    elseif "dog" in tweet:
        dogs.append(tweet)
    else:
        other.append(tweet)
return cats, dogs, other
```

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.***

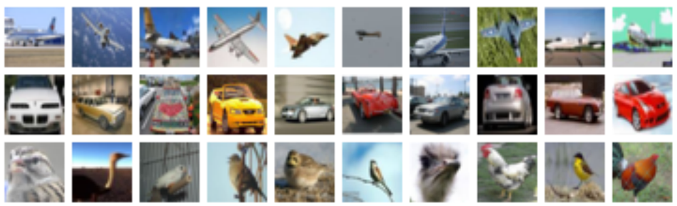


airplane
other
bird

```
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
```

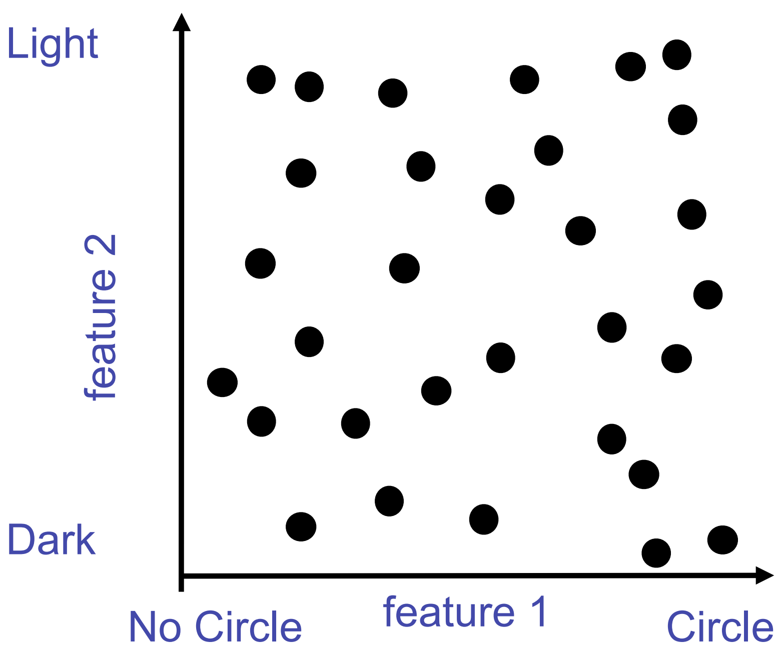
Machine learning algorithms

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



airplane
other
bird

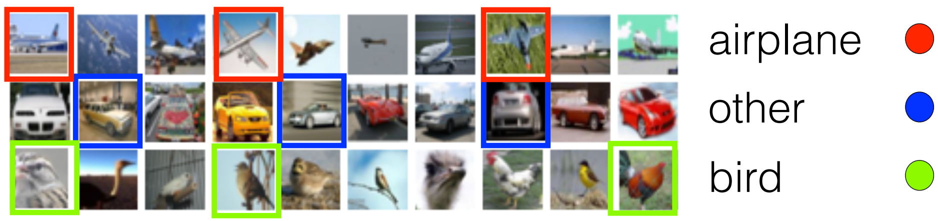
```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```



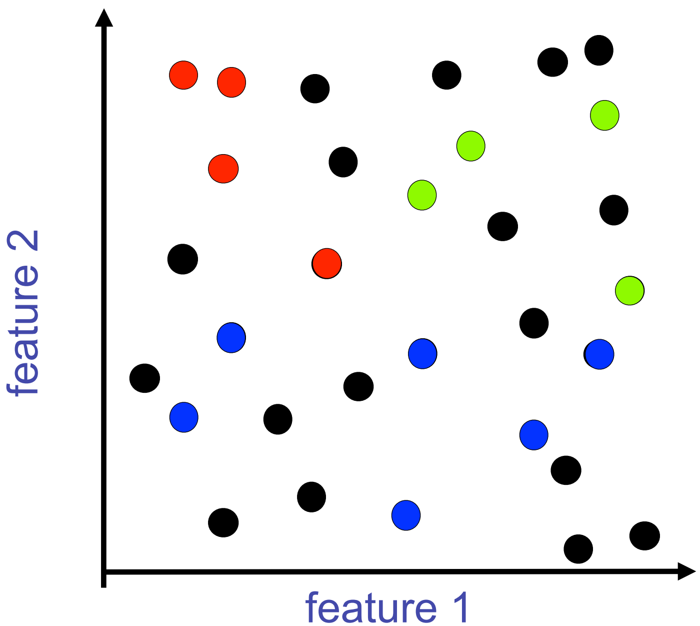
1. Find appropriate representation of the data

Machine learning **algorithms**

Write a program that sorts images 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
```

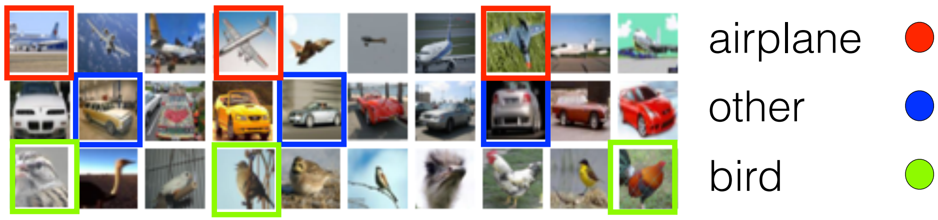


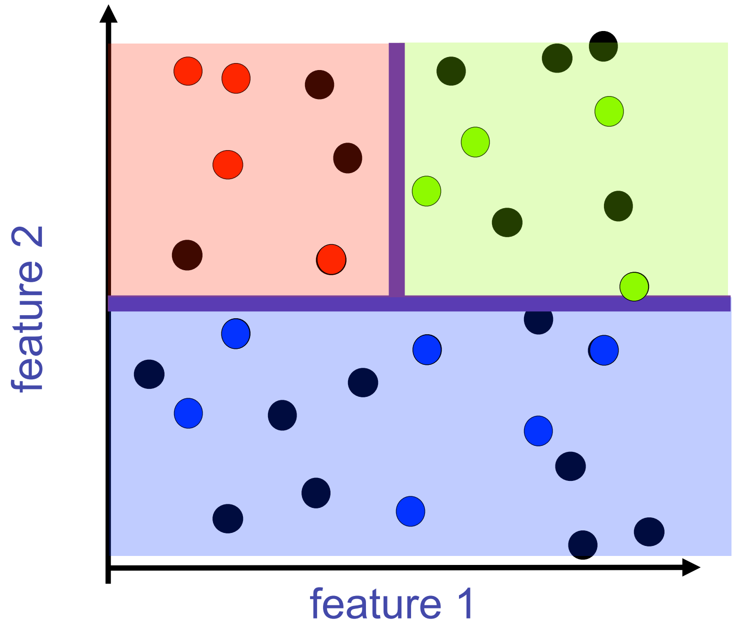
1. Find appropriate representation of the data
2. Crowdsource some samples to get labels

Machine learning **algorithms**

Write a program that sorts images 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
```

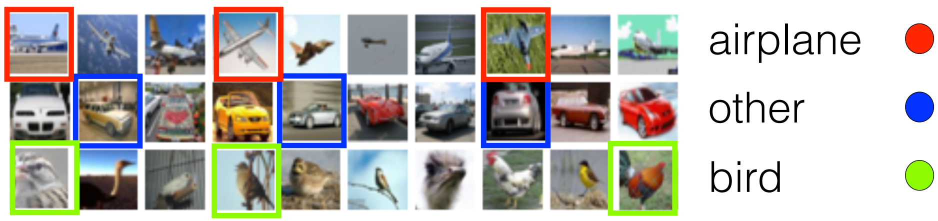


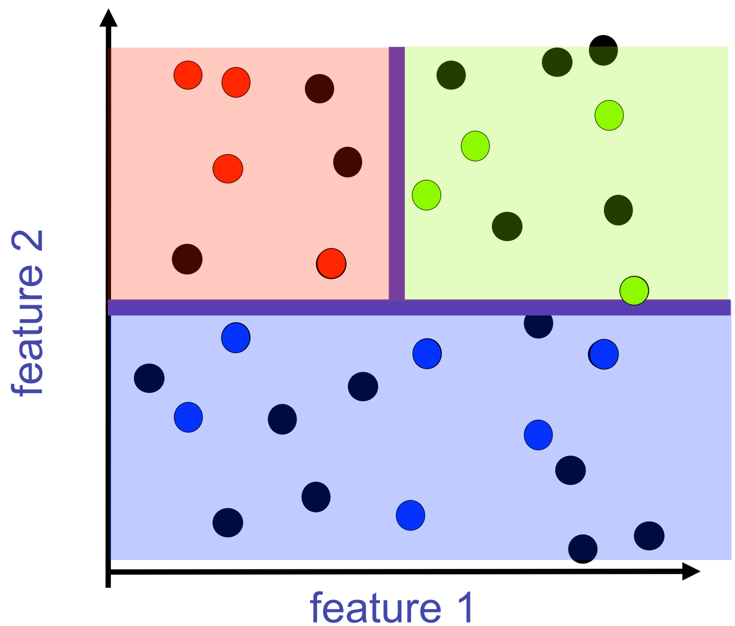


1. Find appropriate representation of the data
2. Crowdsource some samples to get labels
3. Run a machine learning algorithm
   to find decision boundaries

Machine learning algorithms

Write a program that sorts images 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.**

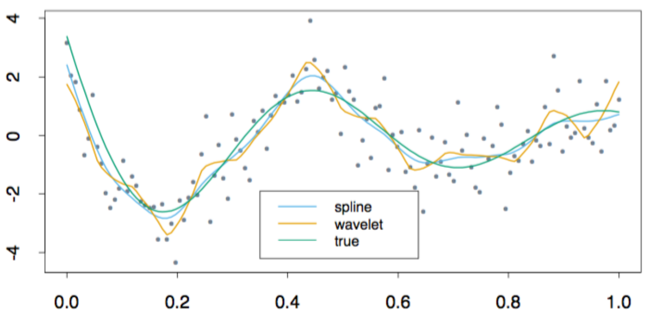
The decision rule of
`if bird in image:`
is **LEARNED using DATA**

Machine learning is incredibly powerful and can have significant (unintended) negative consequences on society through targeting, excluding, and misusing.

Learning objectives of this course:

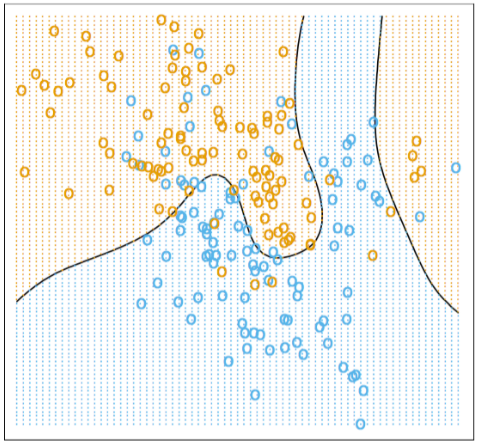
- introduction to the fundamental concepts of machine learning
- analysis and implementation of machine learning algorithms
- knowing how to use machine learning responsibly and robustly

Flavors of ML



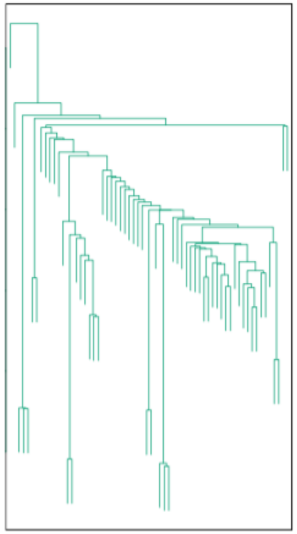
Regression

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



Classification

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



Unsupervised Learning

Predict structure:
tree of life from DNA, find similar images, community detection

Mix of statistics (theory) and algorithms (programming)

CSE446: Machine Learning

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

Course Logistics

- All the information can be found at Course Website:
  https://courses.cs.washington.edu/courses/cse446/22wi/

- **All zoom links are on Canvas**
  - First week lectures 1-3
  - First week sections
  - OHs

- **Instructor**: Sewoong Oh
- **9 amazing TAs**: Jakub Filipek, Joshua Gardner, Thai Quoc Hoang, Chase King, Tim Li, Pemi Nguyen, **Hugh Sun**, Yuhao Wan, Kyle Zhang
- **Lectures**: MWF 9:30-10:20 (first week on Zoom)
- **Questions/announcements/discussions**: EdStem, link on website
- **Personal questions**: cse446-staff@cs.washington.edu
- **Anonymous feedback**: link on website
- **Office hours**: starts on Tuesday, schedule on the website

Prerequisites

- Formally:
  - Linear algebra in MATH 308
  - Algorithm complexity in CSE 312
  - Probability in STAT 390 or equivalent

- Familiarity with:
  - Linear algebra
    - linear dependence, rank, linear equations, SVD
  - Multivariate calculus
    - Differentiate a multi-variate function
  - Probability and statistics
    - Distributions, marginalization, moments, conditional expectation
  - Algorithms
    - Basic data structures, complexity

- “Can I learn these topics concurrently?”
  - Use HW0 to judge skills
  - See website for review materials!

Grading

- 5 homework (100%=12%+22%+22%+22%+22%)
  - Collaboration is okay but must write who you collaborated with.
  - You can spend an arbitrary amount of time discussing and working out a solution with your listed collaborators, but **do not take notes, photos, or other artifacts of your collaboration**. Erase the board you were working on, and once you're alone, write up your answers yourself.
- NO exams
- Extra credit for submitting the proof of course evaluation in the end
- We will assign random subgroups as PODs to collaborate/discuss (when dust clears)

Homework

- HW 0 is out (**Due next Tuesday Jan 11th 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 (instructions on the website)
- Regrade requests on Gradescope
  - within 7 days of release of the grade
- **There is no credit for late work, you get 5 late days**
  - if HW1 is late by 23 hours, then you used 1 late day
  - If HW1 is late by 25 hours, then you used 2 late days

Homework

- HW 0 is out (**Due next Tuesday Jan 11th 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 (instructions on the website)
- Regrade requests on Gradescope
  - within 7 days of release of the grade
- **There is no credit for late work, you get 5 late days**
  - if HW1 is late by 23 hours, then you used 1 late day
  - If HW1 is late by 25 hours, then you used 2 late days

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.

Weekly Sections

- Everyone is enrolled in a 50 minutes in-person section on Thursday.
  - Except for week 1
- Taught by very talented TAs.
- You are not required to attend.
- There is no attendance or quiz.
- It is meant to help you understand the lectures better and deeper.

Weekly Sections

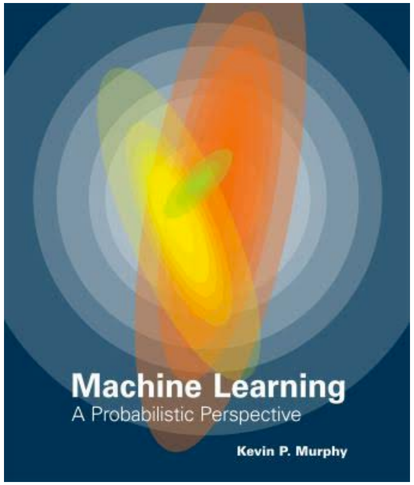
- Previously, We have seen steep decline in attendance in morning sections.
- This time, we have decided to cancel the two morning sections, and instead offer more office hours and dedicate more resources to responding on EdStem

    - Section AA (8:30-9:20): cancelled
    - Section AB (9:30-10:20): cancelled
    - Section AC (10:30-11:20): Chase King, LOW 105
    - Section AD (11:30-12:20): Kyle Zhang, LOW 105
    - Section AE (12:30-1:20): Yuhao Wan, CDH 110B
    - Section AF (1:30-2:20): Jakub Filipek, FSH 107 0

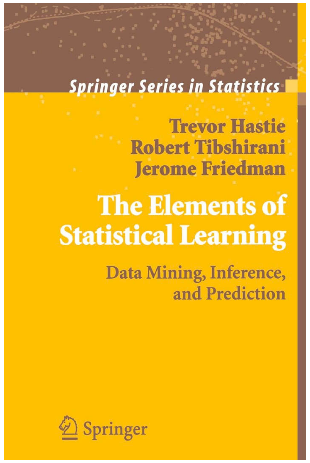
- We ask those registered in AA and AB to attend other sections
- If this is an issue, please contact sewoong@cs.washington.edu

Textbooks

- Required Textbook (optional):
  - ***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



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…

Maximum Likelihood Estimation

Your first consulting job

- *Client*: I have a special coin, if I flip it, what's the probability it will be heads?
- *You*: I need to collect ***data***.

- *You*: The probability is:

- *Client:* Why? What is the principle behind your prediction?

Modelling Coin Flips: Binomial Distribution

- **Data**: sequence $\mathscr{D} = (H, H, T, H, T, \ldots)$
  - **k heads** out of **n flips**
- **Hypothesis:**
  - Flips are i.i.d. (independent and identically distributed):
    - Independent events
    - Identically distributed according to Bernoulli distribution
      - P(Heads) = θ, P(Tails) = $1 - \theta$ for some unknown ***parameter*** $\theta \in [0,1]$
- **Generative model:**
  - Probability that the data $\mathscr{D}$ is generated by hypothesis θ is $P(\mathscr{D}; \theta) \ =$

Maximum Likelihood Estimation

- **Data**: sequence $\mathscr{D} = (H, H, T, H, T, \ldots),$
  - **k heads** out of **n flips**
- **Hypothesis:** P(Heads) = θ, P(Tails) = $1 - \theta$
- **Likelihood**:

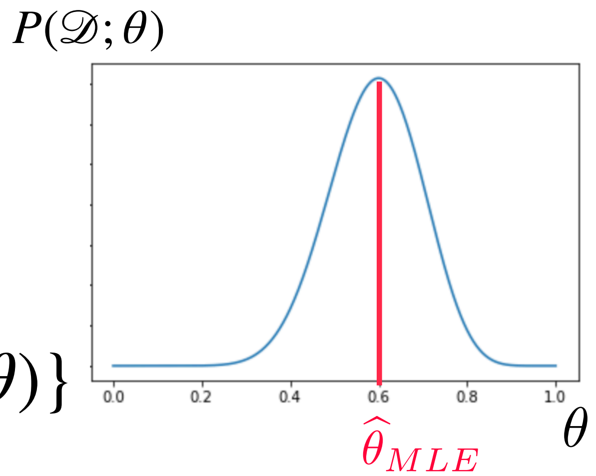
$$\underbrace{P(\mathscr{D};\theta)}_{\textbf{likelihood}} = \theta^k (1-\theta)^{n-k}$$

- **Maximum likelihood estimation** (MLE): Choose θ that maximizes the probability of observed data:

$$\begin{aligned}\widehat{\theta}_{\text{MLE}} &= \arg\max_{\theta} P(\mathscr{D};\theta) \\ &= \arg\max_{\theta} \log P(\mathscr{D};\theta)\end{aligned}$$

Your first learning algorithm

$$\widehat{\theta}_{\mathrm{MLE}} = \arg\max_{\theta} \log P(\mathscr{D};\theta)$$
$$= \arg\max_{\theta} \log\{\theta^k(1-\theta)^{n-k}\}$$
$$= \arg\max_{\theta} k\log\theta + (n-k)\log(1-\theta)\}$$



- Use the fact that derivative is zero at maxima (and also minima)
- Set derivative to zero,

  and find θ satisfying: $$\frac{d}{d\theta}\log P(\mathscr{D};\theta) = 0$$

How good is MLE?

- We treat MLE $\widehat{\theta}_{\text{MLE}}$ as a random variable, where there is a ground truth parameter θ^* that generates the data $\mathscr{D} = (HHTTH\ldots)$ of a fixed size n

- What can we say about this random variable $\widehat{\theta}_{\text{MLE}}$?
- First good property of MLE for Binomial: **unbiased**
  - Definition: **bias** of our MLE is

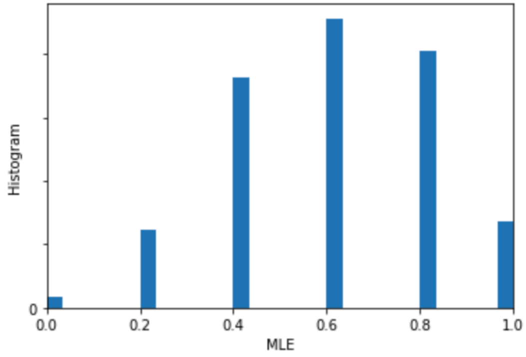
$$\text{Bias}(\widehat{\theta}_{\text{MLE}}) := \mathbb{E}_{\mathscr{D}\sim P_{\theta^*}}\left[\hat{\theta}_{\text{MLE}}\right] - \theta^* =$$

- **Expectation** describes how the estimator behaves *on average*

How many flips do I need?

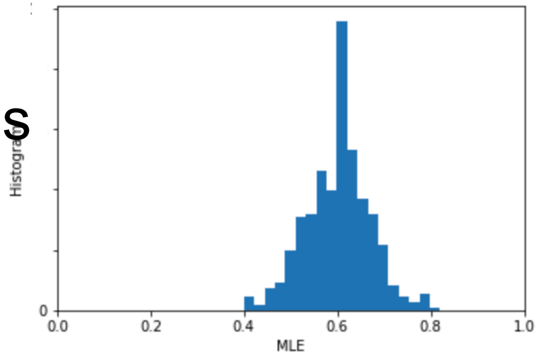
- Consider running many experiments with $\theta^* = \frac{3}{5}$, and observe many instances of the random variable

$$\widehat{\theta}_{MLE} = \frac{k}{n}$$



- *Client*: I flipped the coin 5 times and got 2 heads.

$$\widehat{\theta}_{MLE} =$$



- *Client*: I flipped the coin 50 times and got 30 heads

$$\widehat{\theta}_{MLE} =$$

- *Client:* they are both unbiased, which one is right? Why?
- The width of typical uncertainty is about $\sqrt{\mathrm{Var}(\widehat{\theta}_{\mathrm{MLE}})} = \sqrt{\frac{\theta^*(1-\theta^*)}{n}}$

Quantifying Uncertainty

- The **Variance** is the expected squared deviation from the mean:

$$\text{Variance}(\widehat{\theta}_{MLE}) := \mathbb{E}\left[\left(\widehat{\theta}_{MLE} - \mathbb{E}[\widehat{\theta}_{MLE}]\right)^2\right]$$

- As a rule of thumb

$$\widehat{\theta}_{\text{MLE}} \simeq \mathbb{E}[\widehat{\theta}_{\text{MLE}}] \pm \sqrt{\text{Variance}(\widehat{\theta}_{\text{MLE}})}$$

- Second good property of MLE: **minimum (asymptotic) variance** i.e, for all estimators $\widehat{\theta}$, $\lim_{n\to\infty} \text{Var}(\widehat{\theta}_{\text{MLE}}) \leq \lim_{n\to\infty} \text{Var}(\widehat{\theta})$

Expectation versus High Probability

- Tail bound of a random variable
- For any $\epsilon > 0$ can we bound $\mathbb{P}(|\widehat{\theta}_{MLE} - \mathbb{E}[\widehat{\theta}_{MLE}]| \geq \epsilon)$?

Markov's inequality
For any $t > 0$ and non-negative random variable X

$$\mathbb{P}(X \geq t) \leq \frac{\mathbb{E}[X]}{t}$$

- **Exercise**: Apply Markov's inequality to obtain bound.
(Hint: set $X = \left| \widehat{\theta}_{\mathrm{MLE}} - \mathbb{E}[\widehat{\theta}_{\mathrm{MLE}}] \right|^2$)

Maximum Likelihood Estimation

- **Observe** $X_1, X_2, \ldots, X_n$ drawn i.i.d. from $P(X_i; \theta)$ for some true $\theta = \theta^*$
- **Likelihood function**: $L_n(\theta) = \prod_{i=1}^{n} P(X_i; \theta)$
- **Log-likelihood function**: $\ell_n(\theta) = \log L_n(\theta) = \sum_{i=1}^{n} \log P(X_i; \theta)$
- **Maximum Likelihood Estimator (MLE)**: $\widehat{\theta}_{\text{MLE}} = \arg\max_{\theta} \ell_n(\theta)$

Questions?

Lecture 2: MLE for Gaussian and linear regression

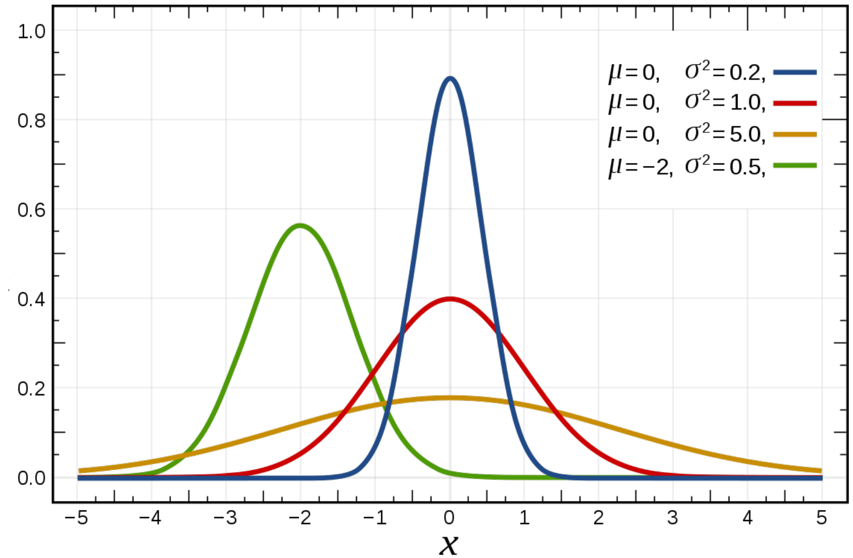
Recap: Maximum Likelihood Estimation

- **Observe** $X_1, X_2, \ldots, X_n$ drawn i.i.d. from $P(X_i; \theta)$ for some true $\theta = \theta^*$
- **Likelihood function**: $L_n(\theta) = \prod_{i=1}^{n} P(X_i; \theta)$
- **Log-likelihood function**: $\ell_n(\theta) = \log L_n(\theta) = \sum_{i=1}^{n} \log P(X_i; \theta)$
- **Maximum Likelihood Estimator (MLE)**: $\widehat{\theta}_{\mathrm{MLE}} = \arg\max_{\theta} \ell_n(\theta)$

What about continuous variables?

- *Client*: What if I am measuring a **continuous variable**?
- ***You*: Let me tell you about Gaussians…**
  - A Gaussian random variable is written as $X \sim \mathcal{N}(\mu, \sigma^2)$ with mean $\mu \triangleq \mathbb{E}[X]$ and variance $\sigma^2 \triangleq \mathbb{E}\left[(X - \mathbb{E}[X])^2\right]$
  - The p.d.f. (Probability Density Function) of X is

$$P(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



Some properties of Gaussians

- affine transformation
  (multiplying by scalar and adding a constant)
  - $X \sim \mathcal{N}(\mu, \sigma^2)$
  - $Y = aX + b \implies Y \sim \mathcal{N}(a\mu + b, a^2\sigma^2)$

- Sum of Gaussians
  - $X \sim \mathcal{N}(\mu_X, \sigma_X^2)$
  - $Y \sim \mathcal{N}(\mu_Y, \sigma_Y^2)$
  - $Z = X + Y \implies Z \sim \mathcal{N}(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2)$

MLE for Gaussian

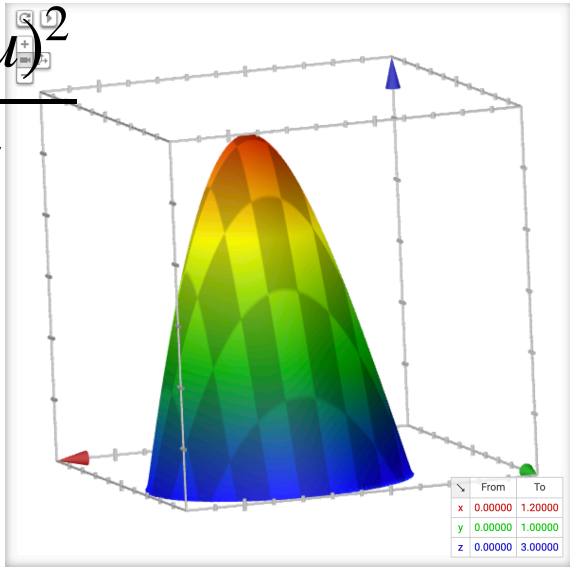
- **Hypothesis**: i.i.d. samples $\mathcal{D} = \{x_1, x_2, \ldots, x_n\}$ from $\mathcal{N}(\mu, \sigma^2)$

$$\begin{aligned} P(\mathcal{D};\mu,\sigma^2) &= P(x_1,\ldots,x_n;\mu,\sigma^2) \\ &= P(x_1;\mu,\sigma^2) \times P(x_2;\mu,\sigma^2) \times \cdots P(x_n;\mu,\sigma^2) \\ &= \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x_i-\mu)^2}{2\sigma^2}} \end{aligned}$$

- **Log-likelihood** of data:

$$\log P(\mathcal{D};\mu,\sigma^2) = -n\log(\sigma\sqrt{2\pi}) - \sum_{i=1}^{n} \frac{(x_i-\mu)^2}{2\sigma^2}$$

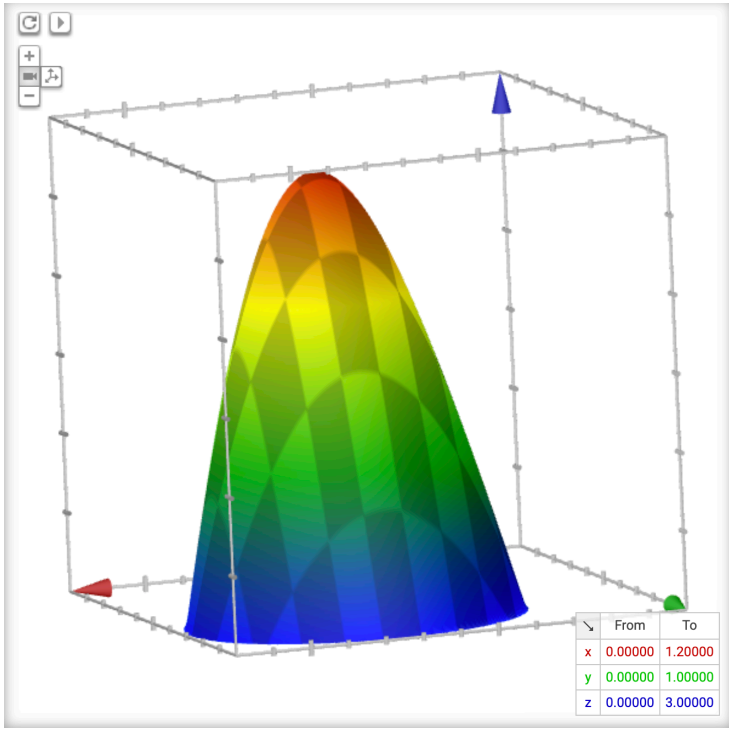
- What is $\widehat{\theta}_{\mathrm{MLE}}$ for $\theta = (\mu, \sigma^2)$?



Your second learning algorithm: MLE for mean of a Gaussian

- What's MLE for mean?

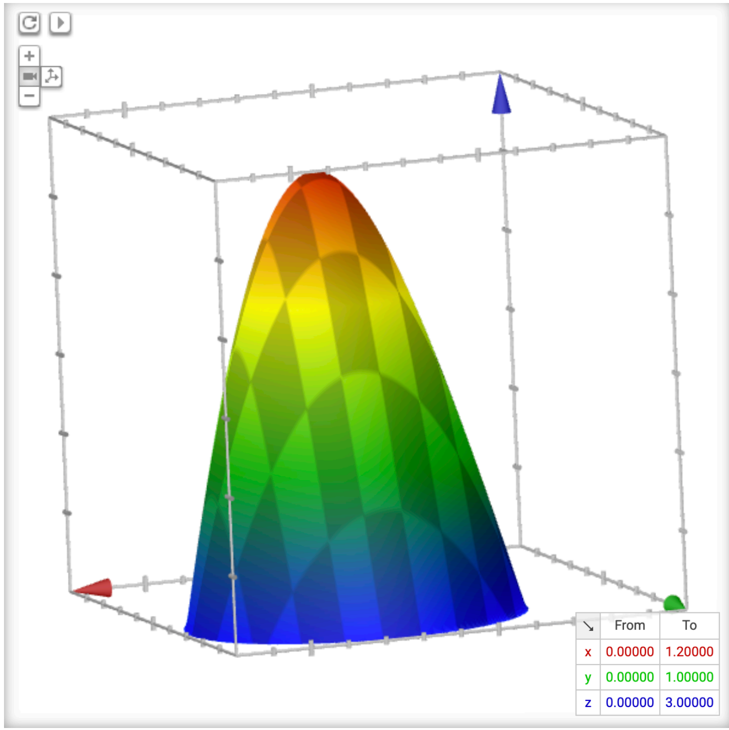
$$\frac{d}{d\mu}\log P(\mathcal{D};\mu,\sigma^2) = \frac{d}{d\mu}\left[-n\log(\sigma\sqrt{2\pi}) - \sum_{i=1}^{n}\frac{(x_i-\mu)^2}{2\sigma^2}\right]$$



MLE for variance

- Again, set derivative to zero:

$$\frac{d}{d\sigma}\log P(\mathscr{D};\mu,\sigma^2) = \frac{d}{d\sigma}\Big[-n\log(\sigma\sqrt{2\pi}) - \sum_{i=1}^{n}\frac{(x_i-\mu)^2}{2\sigma^2}\Big]$$



What can we say about the MLE?

- MLE:
  - $\widehat{\mu}_{\text{MLE}} = \dfrac{1}{n}\sum_{i=1}^{n} x_i$
  - $\widehat{\sigma}^2_{\text{MLE}} = \dfrac{1}{n}\sum_{i=1}^{n} (x_i - \widehat{\mu}_{\text{MLE}})^2$

- MLE for the mean of a Gaussian is **unbiased**
- MLE for the variance of a Gaussian is **biased**
  - $\mathbb{E}[\widehat{\sigma}^2_{\text{MLE}}] \neq \sigma^2$

- Unbiased variance estimator:
  - $\widehat{\sigma}^2_{\text{unbiased}} = \dfrac{1}{n-1}\sum_{i=1}^{n} (x_i - \widehat{\mu}_{\text{MLE}})^2$

Maximum Likelihood Estimation

- **Observe** $X_1, X_2, \ldots, X_n$ drawn i.i.d. from $P(X_i;\theta)$ for some true $\theta = \theta^*$
- **Likelihood function**: $L_n(\theta) = \prod_{i=1}^{n} P(X_i;\theta)$
- **Log-likelihood function**: $\ell_n(\theta) = \log L_n(\theta) = \sum_{i=1}^{n} \log P(X_i;\theta)$
- **Maximum Likelihood Estimator (MLE)**: $\widehat{\theta}_{\mathrm{MLE}} = \arg\max_{\theta} \ell_n(\theta)$

Properties (under benign regularity conditions—smoothness, identifiability, etc.):

- Asymptotically consistent and normal: $\frac{\widehat{\theta}_{MLE} - \theta_*}{\widehat{se}} \sim \mathcal{N}(0,1)$
- Asymptotic Optimality, minimum variance (see Cramer-Rao lower bound)

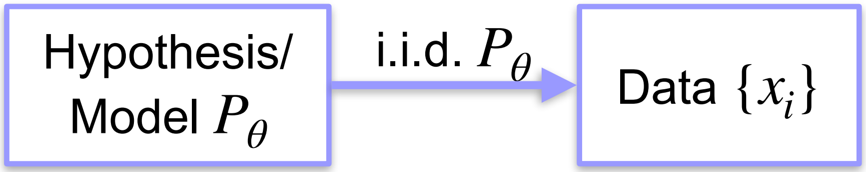
Recap

- Learning is…
  - Collect some data
    - E.g., coin flips

Data $\{x_i\}$

Recap

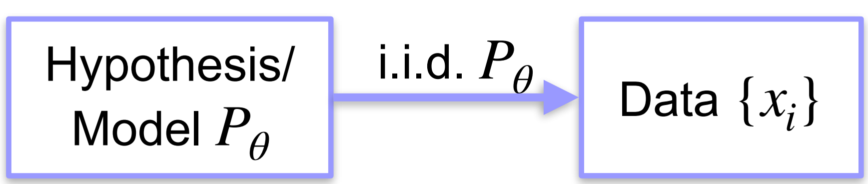
- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial



Recap

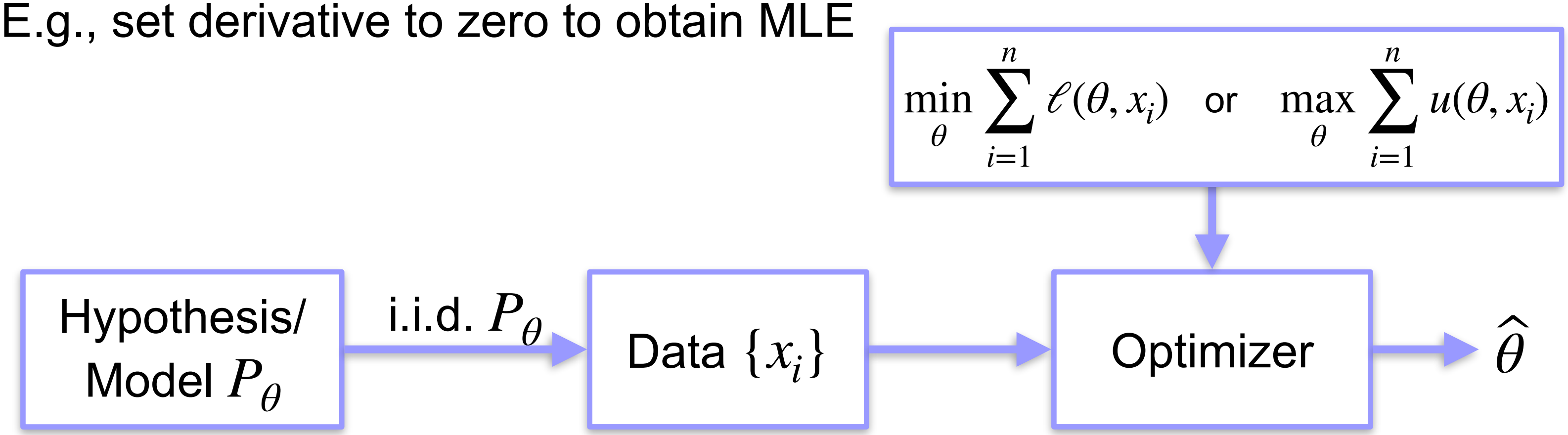
- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial
  - Choose a loss function
    - E.g., data likelihood

$$\min_{\theta} \sum_{i=1}^{n} \ell(\theta, x_i) \quad \text{or} \quad \max_{\theta} \sum_{i=1}^{n} u(\theta, x_i)$$



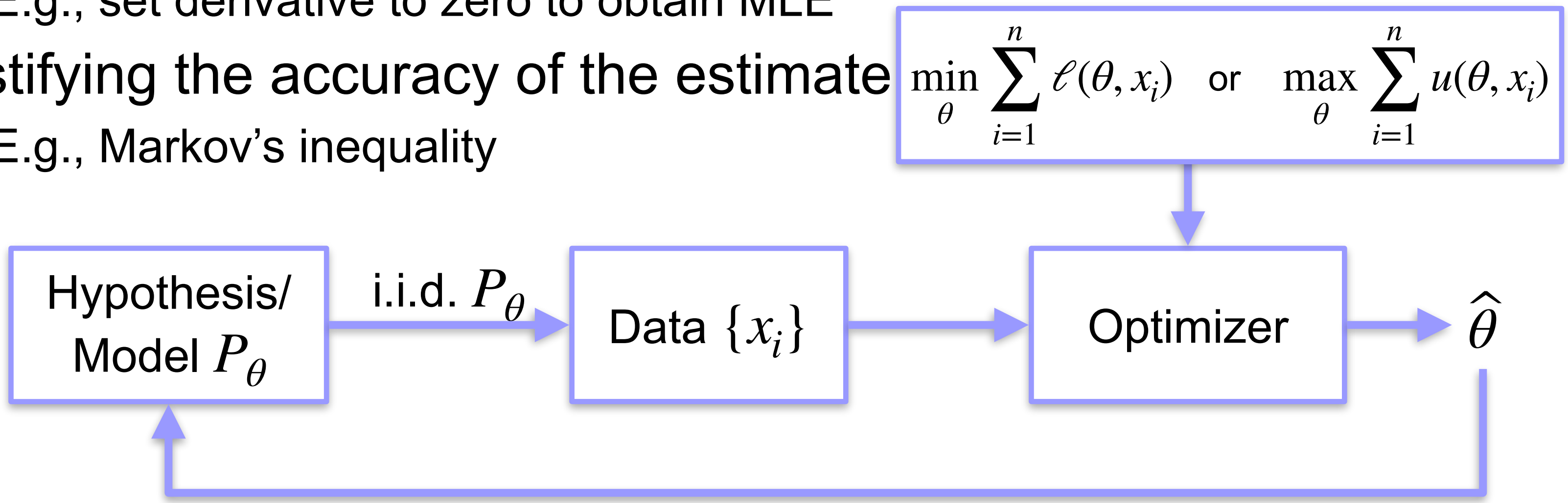
Recap

- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial
  - Choose a loss function
    - E.g., data likelihood
  - Choose an optimization procedure
    - E.g., set derivative to zero to obtain MLE



Recap

- Learning is…
  - Collect some data
    - E.g., coin flips
  - Choose a hypothesis class or model
    - E.g., binomial
  - Choose a loss function
    - E.g., data likelihood
  - Choose an optimization procedure
    - E.g., set derivative to zero to obtain MLE
  - Justifying the accuracy of the estimate
    - E.g., Markov's inequality



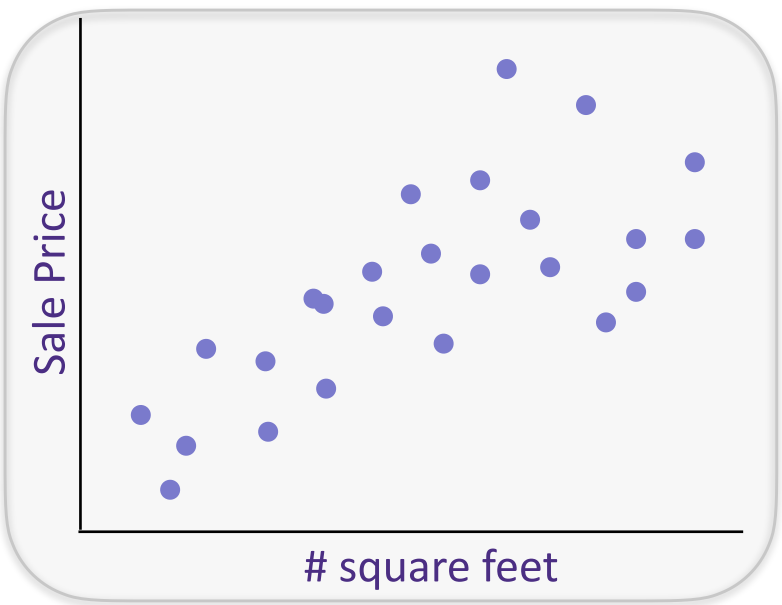
Linear Regression

The regression problem, 1-dimensional

You want to sell your house that is 2,500 sq.ft.
Q. What is the right price?

Collect past sales data on zillow.com:

y = **House sale price** *and* x = **{# sq. ft.}**



Training Data: $x_i \in \mathbb{R}$ $y_i \in \mathbb{R}$

$$\{(x_i, y_i)\}_{i=1}^{n}$$

Process

1. Decide on a **model/hypothesis class**

assume house sale price is a linear function of square feet.

2. Find the function/model/hypothesis which explains/fits the data best

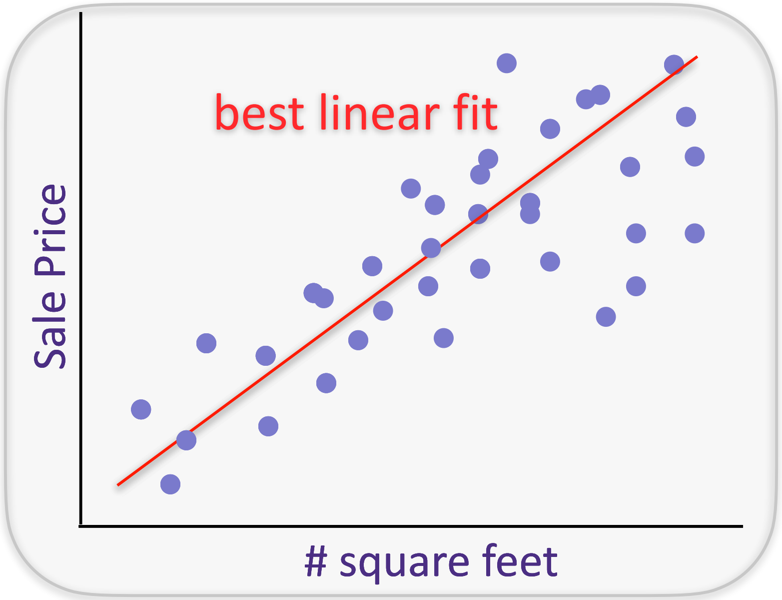
3. Use function to make prediction on new examples
   How much should you put your house on the market?

Fit a function to our data, 1-dimension

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft.}**



1. Training Data: $\{(x_i, y_i)\}_{i=1}^n$ $\quad x_i \in \mathbb{R}$, $y_i \in \mathbb{R}$

2. Hypothesis/Model: linear

$$y_i = w \cdot x_i + \epsilon_i$$

3. Measure of good fit: ℓ_2-loss

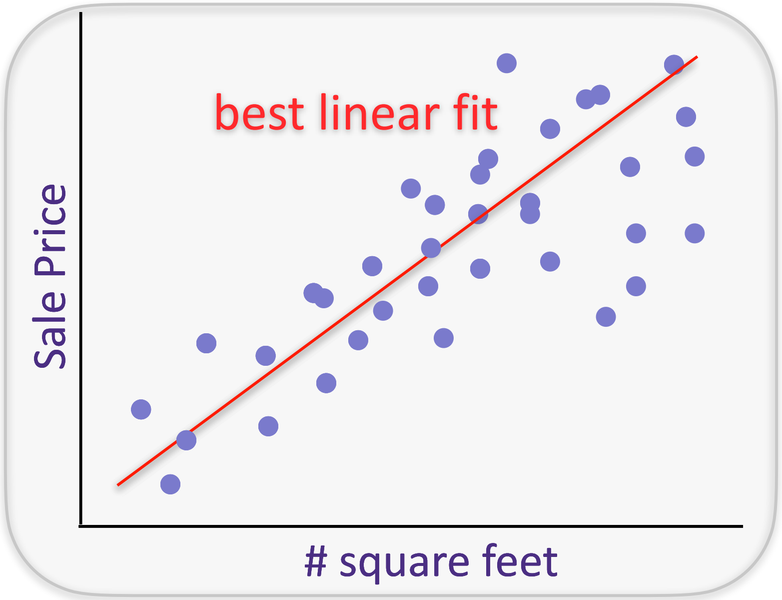
$$\min_{w \in \mathbb{R}} \sum_{i=1}^n (y_i - w x_i)^2 = \sum_{i=1}^n \varepsilon_i^2$$

The regression problem, d-dimensions

Given past sales data on zillow.com, predict:

y = **House sale price** *from*

x = **{# sq. ft., zip code, date of sale, etc.}**



1. Training Data: $\{(x_i, y_i)\}_{i=1}^n$ $\quad x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$

2. Hypothesis/Model: linear

$$y_i = w^T x_i + \epsilon_i$$

3. Measure of good fit: ℓ_2-loss

$$\min_{w \in \mathbb{R}^d} \sum_{i=1}^n (y_i - w^T x_i)^2 = \sum_{i=1}^n \varepsilon_i^2$$

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input

n : # of examples/datapoints

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \quad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input

n : # of examples/datapoints

Linear Model:

$$y_1 = x_1^T w + \epsilon_1$$

$$y_2 = x_2^T w + \epsilon_2$$

$$\vdots$$

$$y_n = x_n^T w + \epsilon_n$$

$$\mathbf{y} = \mathbf{X} w + \epsilon$$

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input

n : # of examples/datapoints

Linear Model:

$$y_1 = x_1^T w + \epsilon_1$$
$$y_2 = x_2^T w + \epsilon_2$$
$$\vdots$$
$$y_n = x_n^T w + \epsilon_n$$

$$\mathbf{y} = \mathbf{X}w + \epsilon$$

ℓ_2-norm of a vector:
(also known as Euclidean norm)

$$\|\epsilon\|_2 = \sqrt{\epsilon_1^2 + \epsilon_2^2 + \cdots + \epsilon_d^2}$$

it follows that

$$\sum_{i=1}^{d} \epsilon_i^2 \;=\; \|\epsilon\|_2^2 \;=\; \epsilon^T \epsilon$$

ℓ_2-Loss: $\widehat{w}_{\text{LS}} \;=\; \arg\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - x_i^T w)^2$

this is also known as **Least Squares** solution

The regression problem in matrix notation

Data:

$$\mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} x_1^T \\ \vdots \\ x_n^T \end{bmatrix}$$

d : # of features/size of the input

n : # of examples/datapoints

Linear Model:

$$y_1 = x_1^T w + \epsilon_1 \qquad \mathbf{y} = \mathbf{X}w + \epsilon$$

$$y_2 = x_2^T w + \epsilon_2$$

$$\vdots$$

$$y_n = x_n^T w + \epsilon_n$$

ℓ_2**-norm of a vector:**

$$\|\epsilon\|_2 = \sqrt{\epsilon_1^2 + \epsilon_2^2 + \cdots + \epsilon_d^2}$$

it follows that

$$\sum_{i=1}^{d} \epsilon_i^2 = \|\epsilon\|_2^2 = \epsilon^T \epsilon$$

ℓ_2**-Loss:** $\widehat{w}_{LS} = \arg\min_{w \in \mathbb{R}^d} \sum_{i=1}^{n} (y_i - x_i^T w)^2 = \arg\min_{w} \|\mathbf{y} - \mathbf{X}w\|_2^2$

$$= \arg\min_{w} (\mathbf{y} - \mathbf{X}w)^T (\mathbf{y} - \mathbf{X}w)$$

The regression problem in matrix notation

$$\widehat{w}_{LS} = \arg\min_{w \in \mathbb{R}^d} (\mathbf{y} - \mathbf{X}w)^T(\mathbf{y} - \mathbf{X}w)$$

Set gradient w.r.t. w to zero to find the minima:

A few reminders on vector calculus

- Gradient of a function:

$$\nabla_w f(w) = \begin{bmatrix} \frac{df(w)}{dw_1} \\ \frac{df(w)}{dw_2} \\ \vdots \\ \frac{df(w)}{dw_d} \end{bmatrix}$$

- Example:

$$\mathcal{L}(w) = w^T w \implies \nabla_W f(w) = 2w$$

$$\mathcal{L}(w) = (Aw)^T(Aw) \implies \nabla_W f(w) = 2AA^T w$$

$$\mathcal{L}(w) = (Aw + b)^T(Aw + b) \implies \nabla_W f(w) = 2A^T(Aw + b)$$

The regression problem in matrix notation

$$\begin{aligned}\widehat{w}_{LS} &= \arg\min_w ||\mathbf{y} - \mathbf{X}w||_2^2 \\ &= \arg\min_w (\mathbf{y} - \mathbf{X}w)^T(\mathbf{y} - \mathbf{X}w) \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}\end{aligned}$$

"Closed form" solution!

Questions?

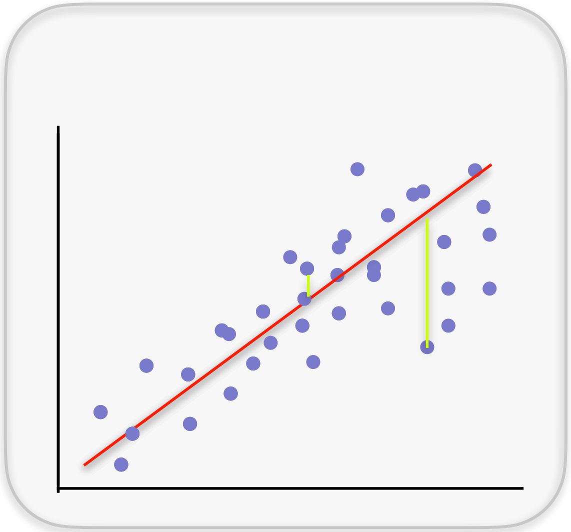
Lecture 3: Linear regression (continued)

The regression problem in matrix notation

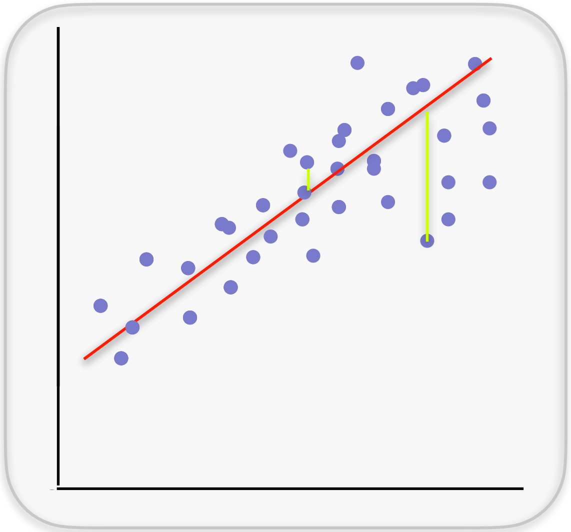
Linear model: $y_i = x_i^T w + \epsilon_i$

Least squares solution:

$$\widehat{w}_{LS} = \arg\min_w ||\mathbf{y} - \mathbf{X}w||_2^2$$
$$= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

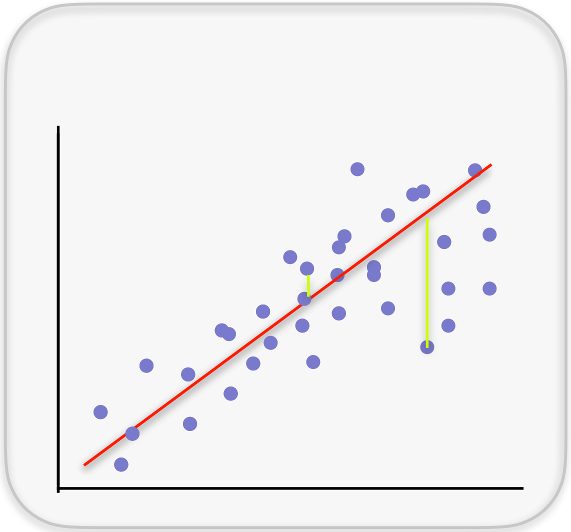


What about an offset (a.k.a intercept)?



The regression problem in matrix notation

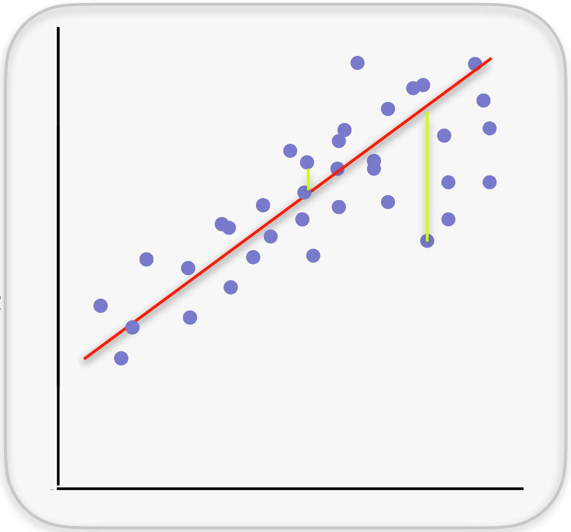
Linear model: $y_i = x_i^T w + \epsilon_i$



Least squares solution:

$$\begin{aligned}\widehat{w}_{LS} &= \arg\min_w \|\mathbf{y} - \mathbf{X}w\|_2^2 \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}\end{aligned}$$

Affine model: $y_i = x_i^T w + b + \epsilon_i$



Least squares solution:

$$\begin{aligned}\widehat{w}_{LS}, \widehat{b}_{LS} &= \arg\min_{w,b} \sum_{i=1}^{n} \left(y_i - (x_i^T w + b)\right)^2 \\ &= \arg\min_{w,b} \|\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)\|_2^2\end{aligned}$$

Dealing with an offset

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

Set gradient w.r.t. w and b to zero to find the minima:

Dealing with an offset

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

$$\mathbf{X}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{X}^T\mathbf{1} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{1}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{1}^T\mathbf{1} = \mathbf{1}^T\mathbf{y}$$

If $\mathbf{X}^T\mathbf{1} = 0$, if the features have zero mean,

$$\widehat{w}_{LS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

$$\widehat{b}_{LS} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

Dealing with an offset

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

$$\mathbf{X}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{X}^T\mathbf{1} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{1}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{1}^T\mathbf{1} = \mathbf{1}^T\mathbf{y}$$

If $\mathbf{X^T1} = 0$,

$$\widehat{w}_{LS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

$$\widehat{b}_{LS} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

In general, when $\mathbf{X}^T\mathbf{1} \neq 0$,

Dealing with an offset

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

$$\mathbf{X}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{X}^T\mathbf{1} = \mathbf{X}^T\mathbf{y}$$

$$\mathbf{1}^T\mathbf{X}\widehat{w}_{LS} + \widehat{b}_{LS}\mathbf{1}^T\mathbf{1} = \mathbf{1}^T\mathbf{y}$$

If $\mathbf{X^T1} = 0$,

$$\widehat{w}_{LS} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

$$\widehat{b}_{LS} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

In general, when $\mathbf{X}^T\mathbf{1} \neq 0$,

$$\mu = \frac{1}{n}\mathbf{X}^T\mathbf{1}$$

$$\widetilde{\mathbf{X}} = \mathbf{X} - \mathbf{1}\mu^T$$

$$\widehat{w}_{\text{LS}} = (\widetilde{\mathbf{X}}^T\widetilde{\mathbf{X}})^{-1}\widetilde{\mathbf{X}}^T\mathbf{y}$$

$$\widehat{b}_{\text{LS}} = \frac{1}{n}\sum_{i=1}^{n} y_i - \mu^T\widehat{w}_{\text{LS}}$$

Process

Decide on a **model:** $y_i = x_i^T w + b + \epsilon_i$

Choose a loss function - least squares

Pick the function which minimizes loss on data

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} \sum_{i=1}^{n} \left(y_i - (x_i^T w + b)\right)^2$$

Use function to make prediction on new examples

$$\hat{y}_{\text{new}} = x_{\text{new}}^T \hat{w}_{LS} + \hat{b}_{LS}$$

Another way of dealing with an offset

$$\widehat{w}_{LS}, \widehat{b}_{LS} = \arg\min_{w,b} ||\mathbf{y} - (\mathbf{X}w + \mathbf{1}b)||_2^2$$

reparametrize the problem as $\overline{\mathbf{X}} = [\mathbf{X}, \mathbf{1}]$ and $\overline{w} = \begin{bmatrix} w \\ b \end{bmatrix}$

$$\overline{\mathbf{X}}\,\overline{w} =$$

Why is least squares a good loss function?

$$\widehat{w}_{LS} = \arg\min_w ||\mathbf{y} - \mathbf{X}w||_2^2$$
$$= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{y}$$

Consider $\quad y_i = x_i^T w + \epsilon_i \quad$ where $\quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2)$

$$\implies y_i \sim$$

$$\implies P(y_i; x_i, w, \sigma) =$$

Why is least squares a good loss function?

Maximum Likelihood Estimator:

$$\begin{aligned}\widehat{w}_{\text{MLE}} &= \arg\max_w \; \log P(\{y_i\}_{i=1}^n; \{x_i\}_{i=1}^n, w, \sigma) \\ &= \arg\max_w \; -n \log(\sigma\sqrt{2\pi}) + \sum_{i=1}^n -\frac{(y_i - x_i^T w)^2}{2\sigma^2}\end{aligned}$$

Why is least squares a good loss function?

Maximum Likelihood Estimator:

$$\begin{aligned}\widehat{w}_{\text{MLE}} &= \arg\max_w \log P(\{y_i\}_{i=1}^n; \{x_i\}_{i=1}^n, w, \sigma) \\ &= \arg\max_w -n\log(\sigma\sqrt{2\pi}) + \sum_{i=1}^n -\frac{(y_i - x_i^T w)^2}{2\sigma^2} \\ &= \arg\min_w \sum_{i=1}^n (y_i - x_i^T w)^2\end{aligned}$$

Recall: $$\widehat{w}_{LS} = \arg\min_w \sum_{i=1}^n \left(y_i - x_i^T w\right)^2$$

$$\widehat{w}_{LS} = \widehat{w}_{MLE} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y}$$

Recap of linear regression

$$\text{Data } \{(x_i, y_i)\}_{i=1}^n$$

Minimize the loss
(Empirical Risk Minimization)

Choose a loss
e.g., $(y_i - x_i^T w)^2$

Solve $\widehat{w}_{\text{LS}} = \arg\min_w \sum_{i=1}^n (y_i - x_i^T w)^2$

Maximize the likelihood
(MLE)

Choose a Hypothesis class
e.g., $y_i = x_i^T w + \epsilon_i, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$

Maximize the likelihood,
$$\widehat{w}_{\text{MLE}} = \arg\max_w \left\{ -n \log(\sigma\sqrt{2\pi}) - \sum_{i=1}^n \frac{(y_i - x_i^T w)^2}{2\sigma^2} \right\}$$

Analysis of Error under additive Gaussian noise

$$\text{if} \quad y_i = x_i^T w + \epsilon_i \quad \text{and} \quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2) \qquad \mathbf{Y} = \mathbf{X}w + \epsilon$$

$$\begin{aligned} \widehat{w}_{MLE} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}w + \epsilon) \\ &= w + (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\epsilon \end{aligned}$$

Maximum Likelihood Estimator is unbiased:

Analysis of Error under additive Gaussian noise

$$\text{if} \quad y_i = x_i^T w + \epsilon_i \quad \text{and} \quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2) \qquad \mathbf{Y} = \mathbf{X}w + \epsilon$$

$$\begin{aligned} \widehat{w}_{MLE} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \\ &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}w + \epsilon) \\ &= w + (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\epsilon \end{aligned}$$

Covariance is:

Analysis of Error under additive Gaussian noise

if $\quad y_i = x_i^T w + \epsilon_i \quad$ and $\quad \epsilon_i \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2) \qquad \mathbf{Y} = \mathbf{X}w + \epsilon$

$$
\begin{aligned}
\widehat{w}_{MLE} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \\
&= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T(\mathbf{X}w + \epsilon) \\
&= w + (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\epsilon
\end{aligned}
$$

$$\mathbb{E}[\hat{w}_{\text{MLE}}] = w$$

$$\text{Cov}(\hat{w}_{\text{MLE}}) = \mathbb{E}[(\hat{w} - \mathbb{E}[\hat{w}])(\hat{w} - \mathbb{E}[\hat{w}])^T] = \sigma^2(\mathbf{X}^T\mathbf{X})^{-1}$$

$$\hat{w}_{\text{MLE}} \sim \mathcal{N}(w, \sigma^2(\mathbf{X}^T\mathbf{X})^{-1})$$

Questions?