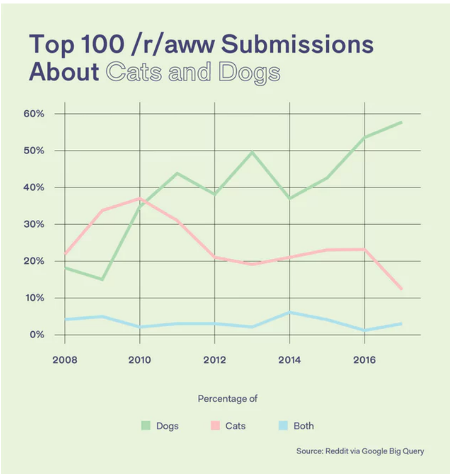
CSE 446/546: Machine Learning

Simon Du and Sewoong Oh

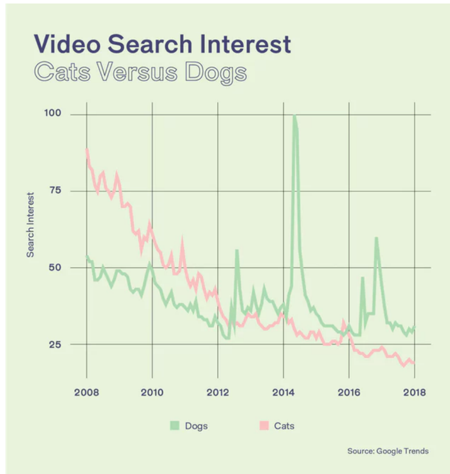
Traditional algorithms

Social media mentions of Cats vs. Dogs

Reddit



Google



Twitter?

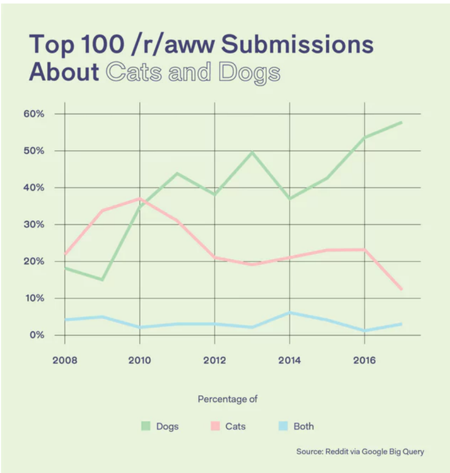
Write a program that sorts tweets into those containing “**cat**”, “**dog**”, or ***other***

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

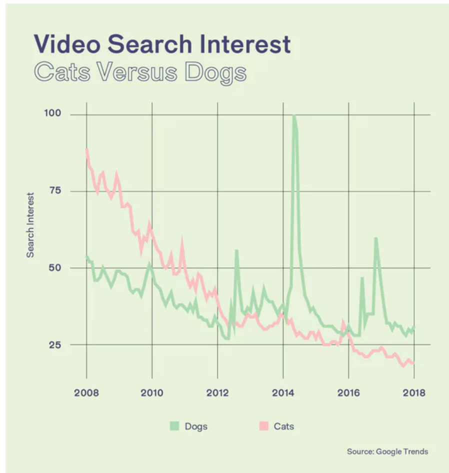
Traditional algorithms

Social media mentions of Cats vs. Dogs

Reddit



Google



Twitter?

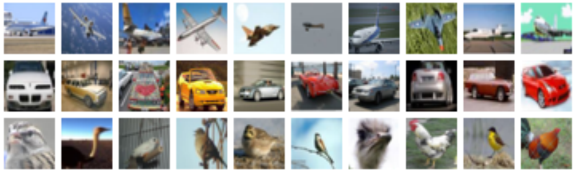
```
cats = []
dogs = []
other = []
for tweet in tweets:
    if "cat" in tweet:
        cats.append(tweet)
    elseif "dog" in tweet:
        dogs.append(tweet)
    else:
        other.append(tweet)
return cats, dogs, other
```

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.***

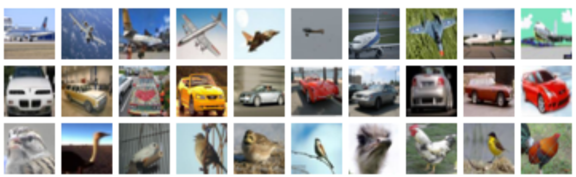


airplane
other
bird

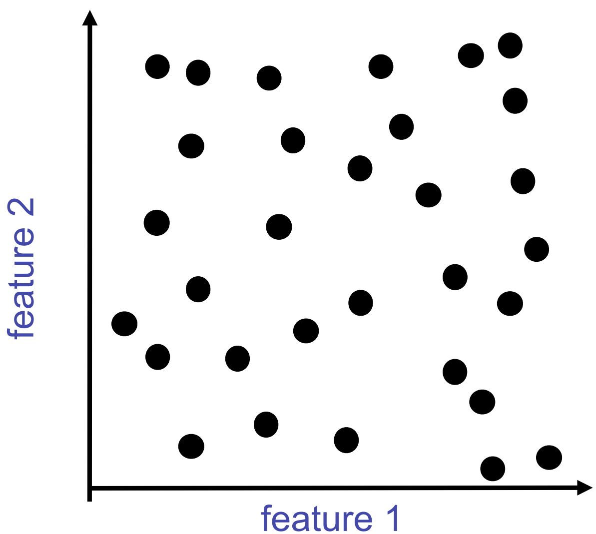
```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

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



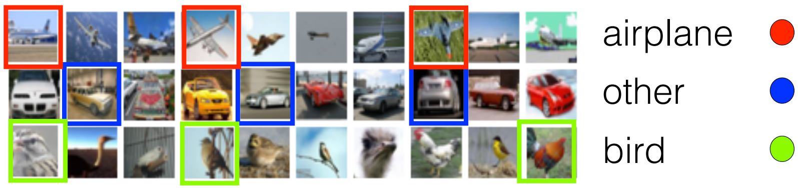
airplane
other
bird

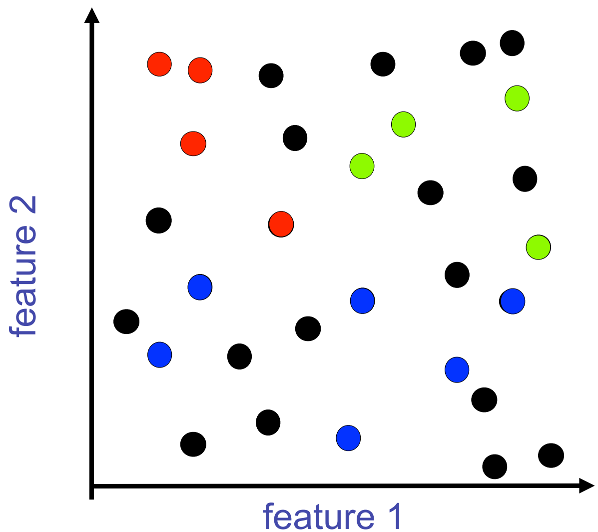



```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

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

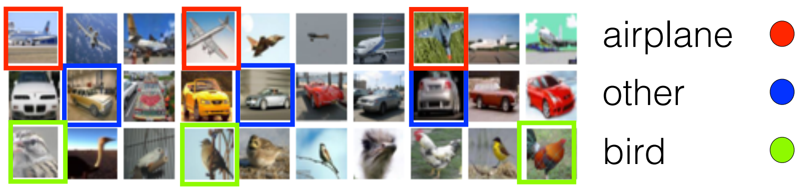


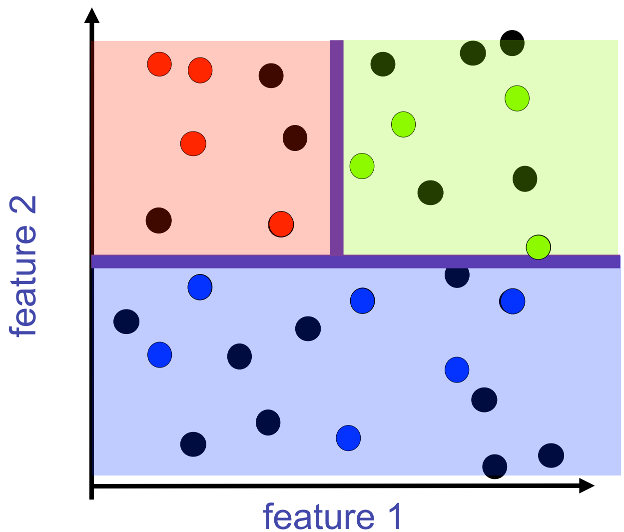



```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

Machine learning algorithms

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

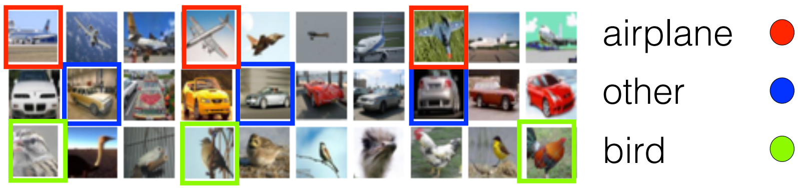


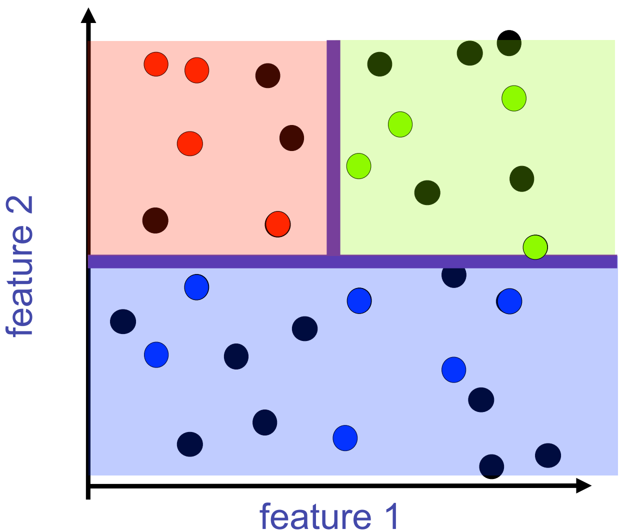



```
birds = []
planes = []
other = []
for image in images:
    if bird in image:
        birds.append(image)
    elseif plane in image:
        planes.append(image)
    else:
        other.append(tweet)
return birds, planes, other
```

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

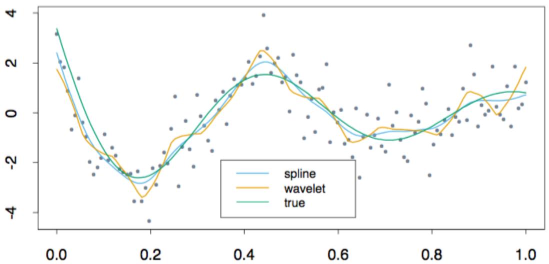
ML uses past data to make personalized predictions

You may also like…

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

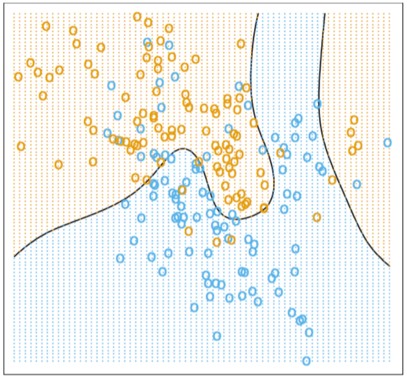
Learning objectives of this course:
- introduction to the fundamental concepts of machine learning
- analysis and implementation of machine learning algorithms
- knowing how to use machine learning responsibly and robustly

Flavors of ML



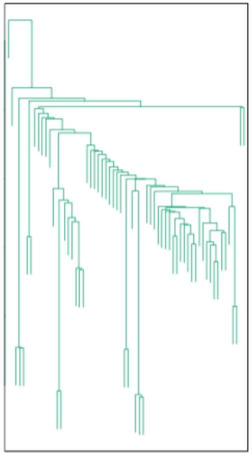
Regression

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)

CSE446/546: Machine Learning

Instructor: Simon Du and Sewoong Oh

Contact: cse446-staff@cs.washington.edu

Course Website: https://courses.cs.washington.edu/courses/cse446/21sp/

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 308, CSE 312, STAT 390 or equivalent
- Familiarity with:
  - Linear algebra
    - linear dependence, rank, linear equations, SVD
  - Multivariate calculus
  - 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 (99%=10%+20%+20%+20%+29%)*
  - *Each contains both theoretical questions and will have programming*
  - *Collaboration okay but must write who you collaborated with. You must write, submit, and understand your answers and code (which we may run)*
  - *Do not Google for answers.*
- *NO exams*
- *1% for submitting the proof of course evaluation*
- *We will assign random subgroups as PODs (when dust clears)*

Homework

- HW 0 is out (**Due next Monday Apr 5th 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
- Regrade requests on Gradescope
- **There is no credit for late work, 5 late days**

Homework

- HW 0 is out (**Due next Monday Apr 5th 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
- Regrade requests on Gradescope
- **There is no credit for late work, 5 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.

Homework & Grading

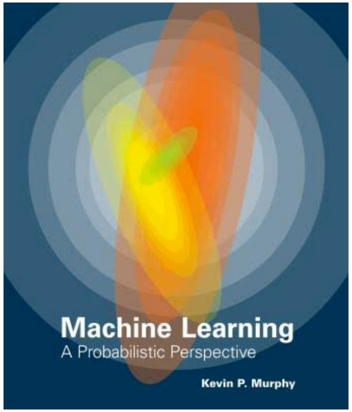
- **CSE 446:**
  - Just do A problems
  - Doing B problems will not get higher grades
  - Grade is on 4.0 scale (relative to students in 446)
- **CSE 546:**
  - If just do A problems, grade is up to 3.8
  - B problems are for 0.2
  - Final grade = A grade + B grade (relative students in 546)

Communication Chanels

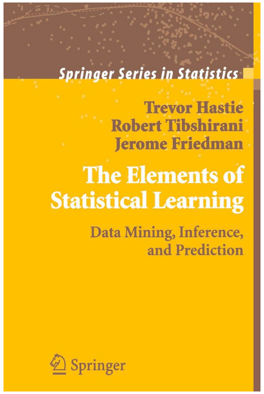
- **Announcements, questions about class, homework help**
  - EdStem (invitation sent, contact TAs if you need access)
  - Weekly Section
  - Office hours (starts tomorrow)
- **Regrade requests**
  - Directly to Gradescope
- **Personal concerns**
  - Email: cse446-staff@cs.washington.edu
- **Anonymous feedback**
  - See website for link

Textbooks

- 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



Addcodes

- Email: Elle Brown (ellean@cs.washington.edu) for addcodes

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

HHTTH ?

Data

- *You*: The probability is: $\frac{3}{5}$
- *Client:* Why? What is the principle behind your prediction?

hypothesis / model $\rightarrow$ Data.

Modelling Coin Flips: Binomial Distribution

- **Data**: sequence $\mathcal{D} = (H, H, T, H, T, \ldots)$
  - **k heads** out of **n flips**
- **Hypothesis:**/assumption
  - Flips are i.i.d. (independent and identically distributed):
    - Independent events $P(A \text{ and } B) = P(A) \times P(B)$
    - Identically distributed according to Bernoulli distribution
      - P(Heads) = θ, P(Tails) = $1 - \theta$
        for some unknown ***parameter*** $\theta \in [0,1]$
- **Generative model:**

$$P(\mathcal{D}|\theta) = P(HHTHT|\theta)$$

$$\overset{\text{indep.}}{=} P(H|\theta)\,P(H|\theta)\,P(T|\theta)\,P(H|\theta)\,P(T|\theta)$$

$$\theta \cdot \theta \cdot (1-\theta) \cdot \theta \cdot (1-\theta)$$

$$P(D;\theta) = \theta^k \cdot (1-\theta)^{n-k}$$

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**:

$$\overbrace{P(\mathcal{D}|\theta)}^{\text{likelihood}} = \theta^k (1-\theta)^{n-k}$$

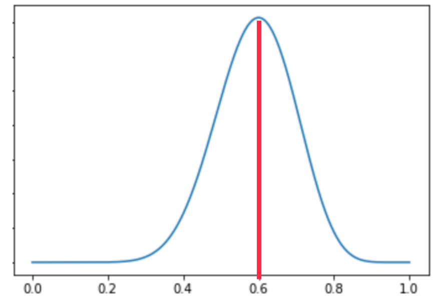
- **Maximum likelihood estimation** (MLE): Choose θ that maximizes the probability of observed data:

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

Your first learning algorithm

$$\begin{cases} P(HHTHT) = \theta^3(1-\theta)^2 \\ P(3\ \text{Heads}) = \binom{5}{3}\theta^3(1-\theta)^2 \end{cases}$$

$P(\mathcal{D}|\theta)$



$\widehat{\theta}_{MLE}$ θ

$$\begin{aligned}\widehat{\theta}_{MLE} &= \arg\max_{\theta}\ \log P(\mathcal{D}|\theta) \\ &= \arg\max_{\theta}\ \log \theta^k(1-\theta)^{n-k}\end{aligned}$$

$$= k\log\theta + (n-k)\log(1-\theta)$$

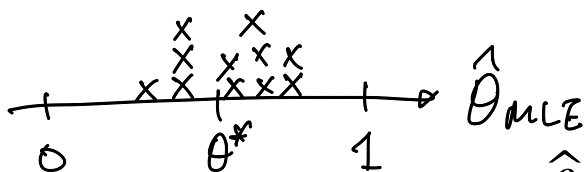
- Use the fact that derivative is zero at maxima (and also minima)
- Set derivative to zero, and find θ satisfying: $$\boxed{\frac{d}{d\theta}\log P(\mathcal{D}|\theta) = 0}$$

$$= \frac{k}{\hat\theta} + \left(-\frac{n-k}{1-\hat\theta}\right) = \frac{k - \cancel{k\hat\theta} - n\hat\theta + \cancel{k\hat\theta}}{\hat\theta(1-\hat\theta)} = 0$$

$$\boxed{\hat\theta = \frac{k}{n}}$$

How good is MLE?

- We treat MLE $\widehat{\theta}_{\mathrm{MLE}}$ 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 $\widehat{\theta}_{\mathrm{MLE}}$?
- First good property of MLE for Binomial: **unbiased**
  - Definition: **bias** of our MLE is

$$\mathrm{Bias}(\widehat{\theta}_{\mathrm{MLE}}) := \underset{\mathcal{D}\sim P_{\theta^*}}{\mathbb{E}}\left[\widehat{\theta}_{\mathrm{MLE}}\right] - \theta^* = \underbrace{\mathbb{E}\left[\frac{k}{n}\right]}_{=\theta^*} - \theta^* = 0$$

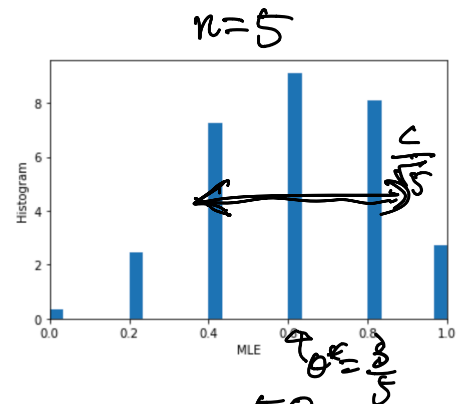
- **Expectation** describes how the estimator behaves *on average*

How many flips do I need?

$$|\theta^* - \hat{\theta}_{MLE}| \le \sqrt{\mathrm{Var}(\hat{\theta})} \leq \sqrt{\frac{c}{4n}}$$

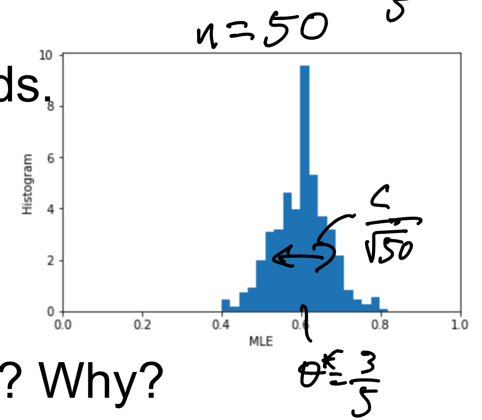
changes each time :D

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



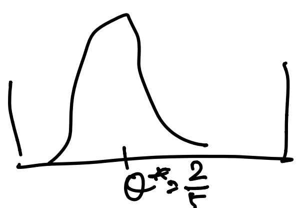
- *Client*: I flipped the coin 5 times and got 2 heads.

$$\hat{\theta}_{MLE} = \frac{2}{5}$$



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

$$\hat{\theta}_{MLE} = \frac{30}{50}$$



- *Client:* they are both unbiased, which one is right? Why?

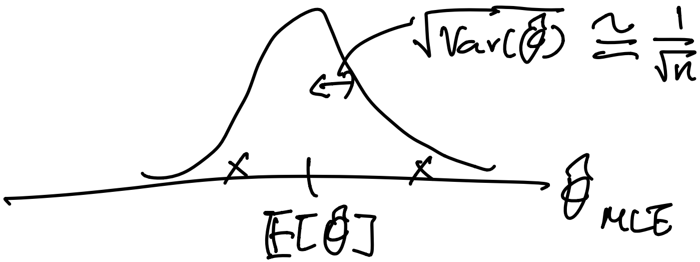
Quantifying Uncertainty

- 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

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



$Var(\hat{\theta}) = f(\theta^*, n)$

$\geq$

$Var(\hat{\theta}_{MLE}) = \mathcal{I}(\theta^*, n)$

as $n \uparrow$

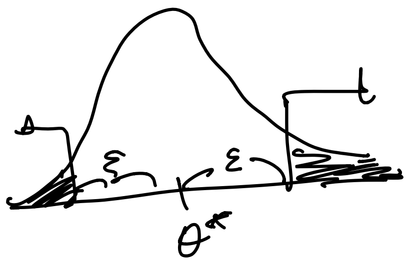
- Second good property of MLE: **minimum (asymptotic) variance**

$n \uparrow \infty$

- **Exercise**: compute the $\text{Variance}(\widehat{\theta}_{\text{MLE}})$

Expectation versus High Probability

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



Confidence interval.

Markov's inequality → chebyshev's ineq.

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 $X = |\widehat{\theta}_{MLE} - \theta^*|^2$)

Maximum Likelihood Estimation

Observe $X_1, X_2, \ldots, X_n$ drawn IID from $f(x;\theta)$ for some "true" $\theta = \theta_*$

Likelihood function $L_n(\theta) = \prod_{i=1}^{n} f(X_i;\theta)$

Log-Likelihood function $l_n(\theta) = \log(L_n(\theta)) = \sum_{i=1}^{n} \log(f(X_i;\theta))$

Maximum Likelihood Estimator (MLE) $\widehat{\theta}_{MLE} = \arg\max_{\theta} L_n(\theta)$