CS109: Probability for Computer Scientists

Jerry Cain

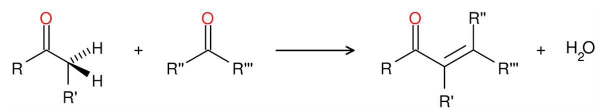
April 1, 2024

Lecture Discussion on Ed

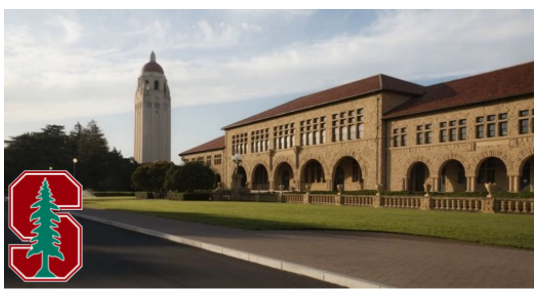
Welcome to CS109!

Jerry Cain

I went here from 1987 through 1991 and majored in chemistry.

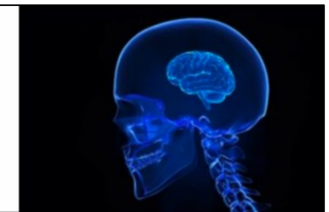


Then I came here for a PhD in chem, switched to CS



Received MSCS 1998
Lecturer: nearly 28 years

<u>My interests over time</u>

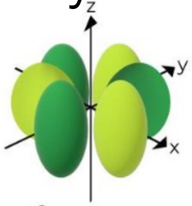
Chemistry and Physics	
Computer Science	
STEM Education	

Why Jerry likes probability

- I majored in chemistry and focused on physical chemistry—thermodynamics, quantum mechanics, etc.—and my undergraduate research was rooted in surface science and statistical mechanics.
- When I switched to CS as a grad student here, I focused on CS theory and all the beautiful mathematics that comes with it.
- Probability has revived parts of AI and information theory that were thought to be borderline dead when I was getting my MSCS degree here during the 90's.

1974

1996



$f_{x(x^2-3y^2)}$

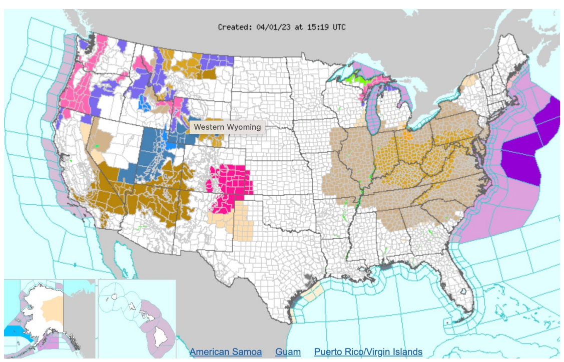
$$PV = \frac{1}{3} N m v_{\mathrm{rms}}^2 . \quad f(v) = 4\pi\left(\frac{m}{2\pi kT}\right)^{\frac{3}{2}} v^2 e^{-\frac{mv^2}{2kT}} \quad v_{\mathrm{rms}}^2 = \int_0^\infty v^2 f(v)\, dv = 4\pi\left(\frac{m}{2\pi kT}\right)^{\frac{3}{2}} \int_0^\infty v^4 e^{-\frac{mv^2}{2kT}}\, dv$$

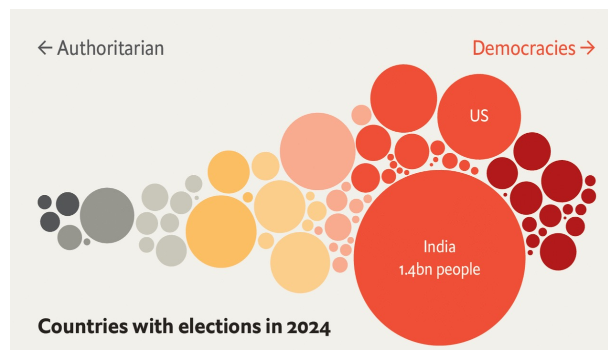
What makes this quarter important

We are seeing a huge surge in **statistics, predictions, and probabilistic models** shared through global news, governing bodies, and social media.

The **technological and social innovation** we develop during this time will strongly influence how we solve interesting problems impacting the **lives of countless people across the globe**.

National Weather Service Alerts
https://www.weather.gov/





World Politics
https://abcnews.go.com/538
https://www.nytimes.com/
https://www.economist.com/

Course Mechanics

Prerequisites

CS106B

Programming
Recursion
Hash tables
Binary trees

MATH 51

Multivariate differentiation
Multivariate integration
Working knowledge of linear algebra (e.g., vectors)

CS103

Proofs (induction)
Set theory
Mathematical maturity

Companion class: CS109ACE

- CS109ACE is an extra 1-unit "ACE" section that provides additional support, practice, and instruction for undergraduate students concerned about their preparation and mathematical background.
- Meets for an additional weekly section and has additional review sessions, office hours, and practice problems
- Admission is via application. You can ignore the published deadline in the form, as our CS109ACE application is due this Friday, April 5th at 5:00pm.
- CS109ACE meets on Mondays from 5:30 – 7:20pm, (location TBD) and starts on April 8th.
- Feel free to email Michelle Qin at **`mdqin@stanford.edu`** with any questions.

Michelle Qin

Course components

42%	6 Problem Sets
22%	Two Midterms
21%	Final Exam
5%	Section Participation
10%	Concept Checks

Course components

42%	**6 Problem Sets**
22%	Two Midterms
21%	Final Exam
5%	Section Participation
10%	Concept Checks

LaTeX

Written portion
- LaTeX for powerful typesetting
- Tutorial on CS109 website

python

Coding portion in Python
- Review session on Thursday 04/04 at noon in Huang 018

Late policy
- Need a short extension? No need to ask! Take an extra class period.
- Need a longer extension? Just ask us and we'll probably be okay with it.
- Extensions can be at most two extra class periods.

Course components

42%	6 Problem Sets
22%	**Two Midterms**
21%	Final Exam
5%	Section Participation
10%	Concept Checks

- In person! But held outside of class so we can let you work *sans* time pressure.
- Closed-book, mostly-closed-notes, closed-computer, no calculators.
- You can bring **two** 8.5" x 11" pages of notes—using both sides—and refer to them during the exams.
- Held on Wednesdays.
  - Week 4: Wed, 04/24, 7:00 – 9:00pm
  - Week 7: Wed, 05/15, 7:00 – 9:00pm
- Irreconcilable Conflict? Let Jerry know and we'll work something out.

Course components

42% 6 Problem Sets

22% Two Midterms

21% Final Exam

- Scheduled for Saturday, June 8th from 8:30 until 11:30am (our official time).
- Closed-book, mostly-closed-notes, closed computer, no calculators.
- You can prepare **four** 8.5” x 11” pages of notes—using both sides—and refer to them and a provided reference sheet during the exam.
- Conflict with another final exam? I’ll offer the final on Friday, June 7th from 12:15pm to 3:15pm for those with a documented conflict with another final exam.

5% Section Participation

10% Concept Checks

Course components

42%	6 Problem Sets
22%	Two Midterms
21%	Final Exam
5%	**Section Participation**
10%	Concept Checks

- Sections meet on Thursdays and Fridays. Times are already posted right here.
- Sections start Week 2
- Your section grade is 100%, but each absence (beyond one freebie) reduces the weight of section participation and increases the weight of the final exam
- Go to section!

Course components

42%	6 Problem Sets
22%	Two Midterms
21%	Final Exam
5%	Section Participation
10%	**Concept Checks**

- Short set of questions released after each lecture.
- Questions are straightforward and there to ensure you've absorbed the key points and formulas from class.
- All of Week n's concept checks are due the Tuesday of Week n + 1 at noon.
- No late submissions accepted unless truly extenuating circumstances make it truly impossible to meet deadline.

CS109 Contest

- Announced mid-quarter, genuinely optional
- Boost final course grades after letter grade buckets have been determined

Your baseline is CS109, and the sky is the limit.

Some of last quarter's winners:

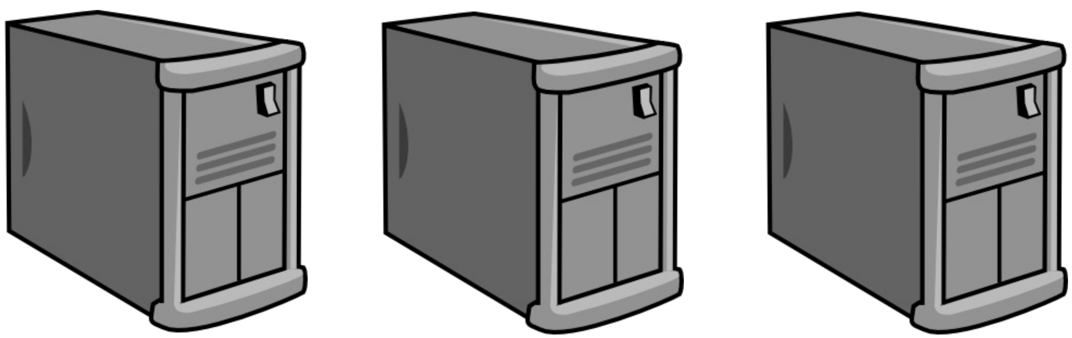
- The Probability of Curing Cancer: Will My Clinical Trial Succeed?
- Modeling Indexical Fields as Bayesian Networks
- StatTuring: Distinguishing between LLM and Human text
- Parka: A Mobile App for Early Parkinson's Disease Detection

Why you should take CS109

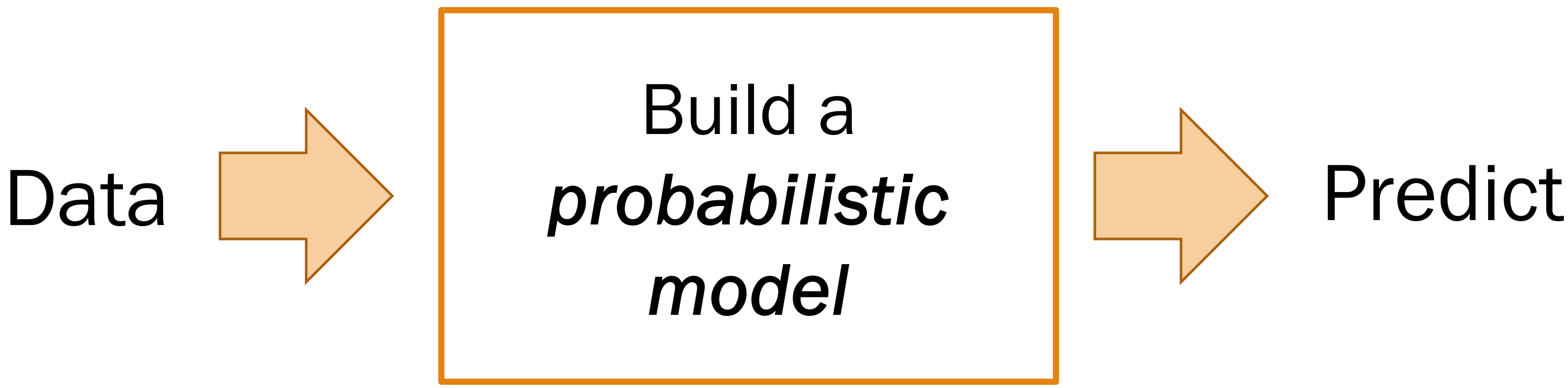
Traditional View of Probability

CS view of probability

```
http://www.site.com
```



Moonshot: Machine Learning

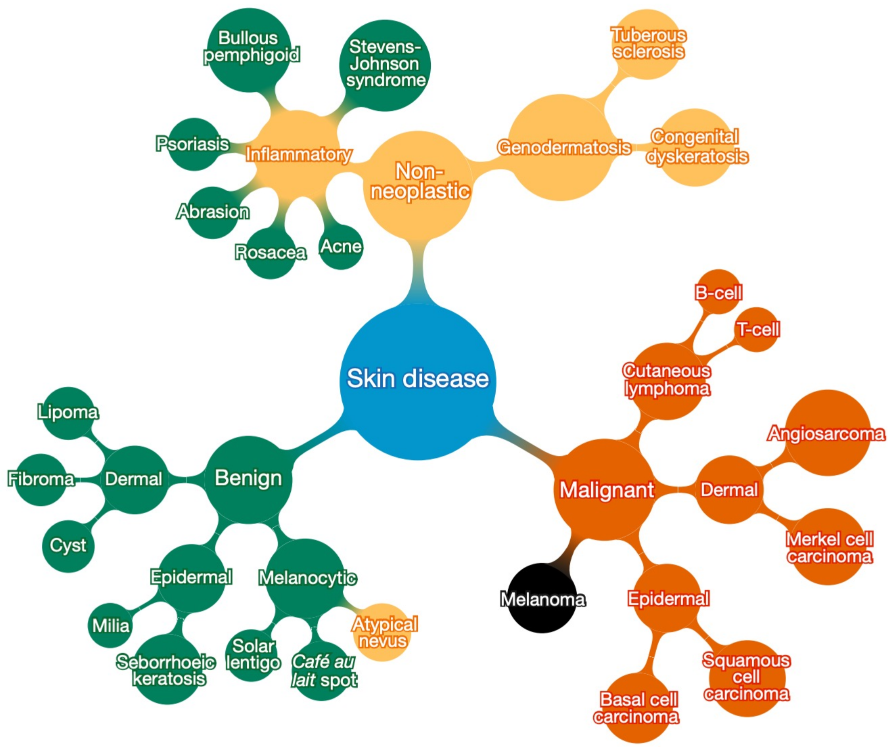


Binary Classification Silliness

chihuahua or muffin?

poodle or fried chicken?

Classification: Where is this useful?



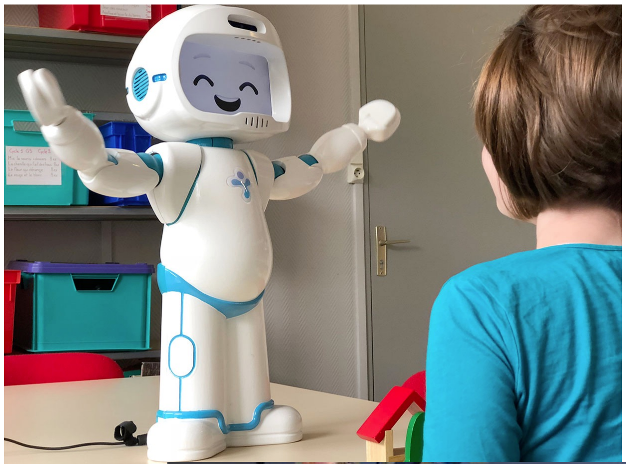
A machine learning algorithm performs **better than** the best dermatologists.

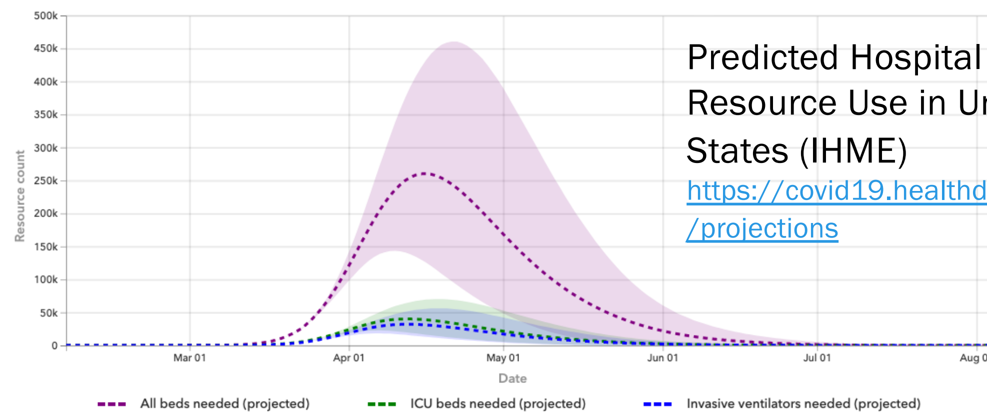
Developed in 2017 at Stanford.

Esteva, Andre, et al. "Dermatologist-level classification of skin cancer with deep neural networks." *Nature* 542.7639 (2017): 115-118.

Probability is *more* than just machine learning.

Probability and medicine



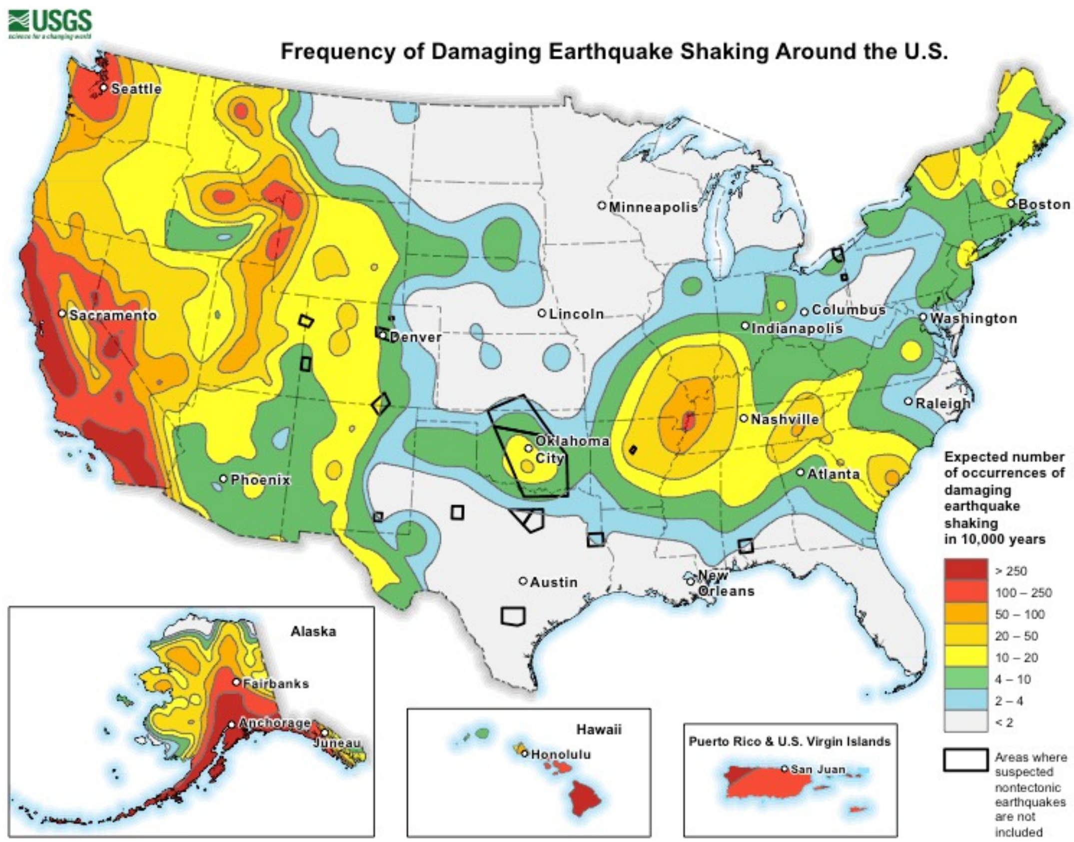


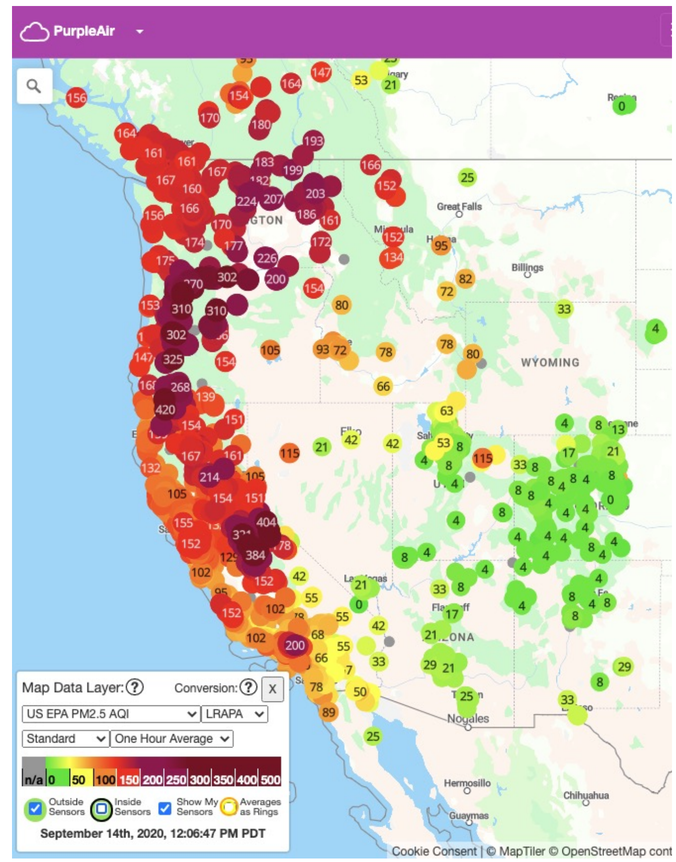
Predicted Hospital Resource Use in United States (IHME)
https://covid19.healthdata.org/projections

How do COVID-19, RSV, and monkeypox testing rates in a region correlate with the actual spread of the disease?

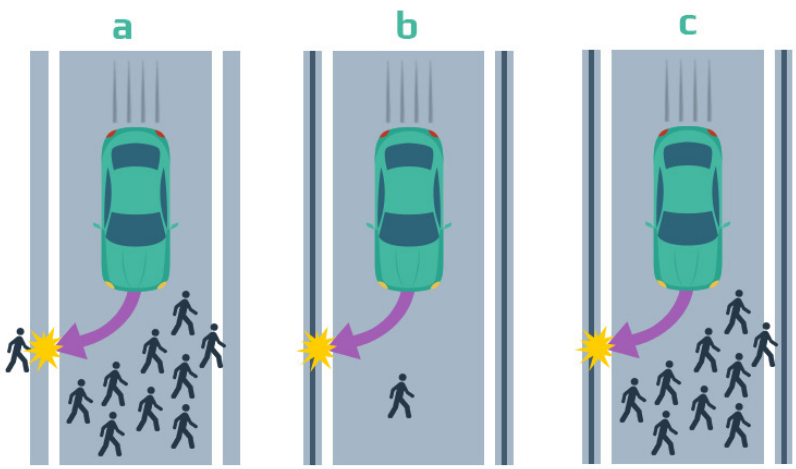
Probability and art

Probability, Meteorology, and Earthquake Prediction





Probability and ethics



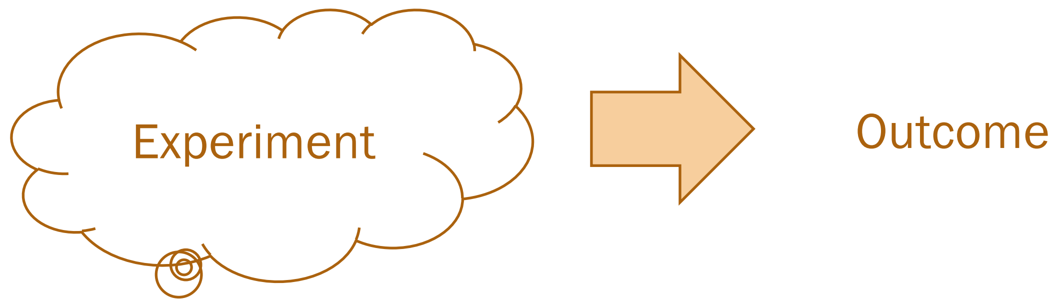
The golden rule for autonomous car ethics doesn't exist

So far, there are no unified ethical standards … for autonomous cars. The big [Moral Machine study](#) conducted by MIT showed that it's hard to identify universal ethical values. The moral choices that people made in the MIT survey were different and varied even at a local level. That's why it's hard to create a universal ethics of self-driving cars that won't be controversial. [[source](#)]

Counting

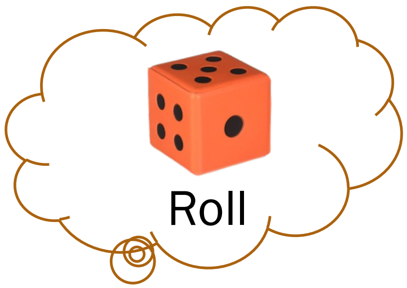
What is Counting?

An experiment
in probability:



Counting: How many possible outcomes can occur by performing this experiment?

What is ~~Counting~~ Combinatorial Analysis?

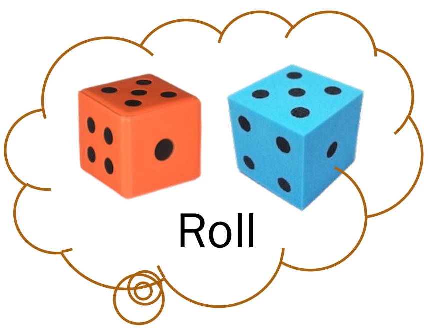


6

{1, 2, 3,
4, 5, 6}

3

{2, 4, 6}



36

{(1, 1) , (1, 2), (1, 3), (1, 4), (1, 5), (1, 6),
(2, 1) , (2, 2), (2, 3), (2, 4), (2, 5), (2, 6),
(3, 1) , (3, 2), (3, 3), (3, 4), (3, 5), (3, 6),
(4, 1) , (4, 2), (4, 3), (4, 4), (4, 5), (4, 6),
(5, 1) , (5, 2), (5, 3), (5, 4), (5, 5), (5, 6),
(6, 1) , (6, 2), (6, 3), (6, 4), (6, 5), (6, 6)}

for the moment, assume that all outcomes are equally likely – i.e. that all dice in this slide are fair. It's a simplifying assumption, but even if outcomes aren't equally likely, it doesn't influence how we count them!

Sum Rule of Counting, Inclusion-Exclusion Principle

If the outcome of an experiment can be either from

Set A, where $|A| = m$,

or Set B, where $|B| = n$,

where A and B may overlap, then

The total number of outcomes of the experiment is

$$|A \cup B| = |A| + |B| - |A \cap B|.$$

example: $A = \{2, 4, 6, 8, 10, 12, 14\}$

$B = \{3, 6, 9, 12, 15\}$

$A \cap B = \{6, 12\}$

here, $m = 7$, $n = 5$

outcomes in $A \cup B = 7 + 5 - 2 = 10$

Product Rule of Counting

If an experiment has two parts, where

the first part's outcomes are drawn from A, where $|A| = m$,
<u>and</u> the second part's outcomes are drawn from B, where $|B| = n$,

Then the number of outcomes of the experiment is

$$|A||B| = mn.$$

example:

$\{H, T\} \times \{H, T\} \rightarrow$ HH, HT, TH, TT

Two-step experiment

→ A → B

This generalizes to multistep experiments—i.e., three steps, five steps, fifty steps, and so forth.

example $|A||B||C||D||E| = m \cdot n \cdot p \cdot q \cdot r$

Baby's First Example: Transmitting bytes over a network

An 8-bit string is sent over a network.

- The receiver only accepts strings that either start with 01 or end with 00.

How many 8-bit strings will the receiver accept?

`01001100`

byte (8 bits)

Define

A : 8-bit strings starting with 01

B : 8-bit strings ending with 00

Baby's First Example: Transmitting bytes over a network

An 8-bit string is sent over a network.

- The receiver only accepts strings that either start with 01 (A) or end with 00 (B).

How many 8-bit strings will the receiver accept?

01001100

byte (8 bits)

Define

A : 8-bit strings starting with 01

B : 8-bit strings ending with 00

two options for each

A: all members structured as 0 1 ? ? ? ? ? ? $|A| = 2^6$

B: all members structured as : ? ? ? ? ? ? 0 0 $|B| = 2^6$

A∩B: all members structured as : 0 1 ? ? ? ? 0 0 $|A \cap B| = 2^4$

$$\text{answer} = |A \cup B| = |A| + |B| - |A \cap B| = 2^6 + 2^6 - 2^4$$
$$= 2 \cdot 2^6 - 2^4$$
$$= 2^7 - 2^4 = 112$$

License plates

How many CA license plates are possible with each of the following formats?

(pre-1982)

(present day)

License plates

How many CA license plates are possible with each of the following formats?

(pre-1982)

$\underbrace{26}_{A-Z} \cdot \underbrace{26}_{A-Z} \cdot \underbrace{26}_{A-Z} \cdot \underbrace{10}_{0-9} \cdot \underbrace{10}_{0-9} \cdot \underbrace{10}_{0-9} = 26^3 \cdot 10^3 = 17,576,000$

(present day)

allows for leading 1, 2, 3, 4, 5, 6, 7, 8, 9

approach 1: $9 \cdot 26 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10$
$+ \; 26 \cdot 26 \cdot 26 \cdot 10 \cdot 10 \cdot 10 = 175,760,000$

original count, pre-1982

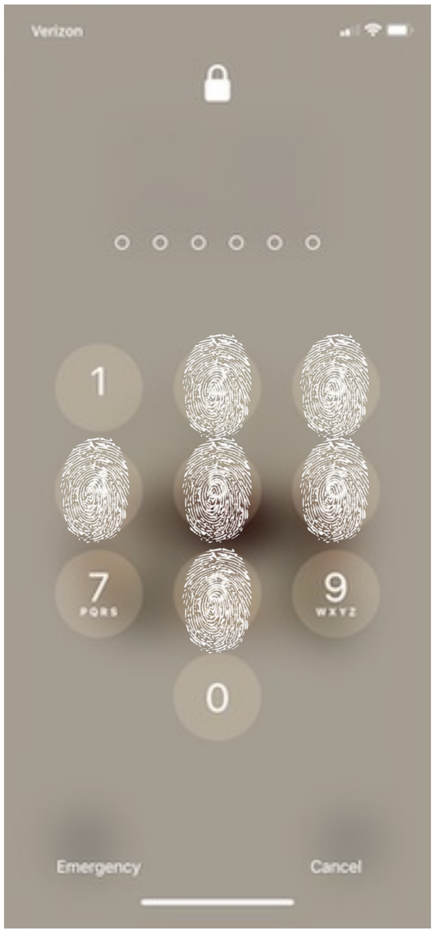
approach 2: $(9 + 1) \cdot 17,576,000 = 175,760,000$

leading 1-9 or no leading digit

pre-1982 count

Permutations I

Unique 6-digit passcodes with six smudges



How many unique 6-digit passcodes are possible if a phone password uses each of **six** distinct numbers?

Arrange n indistinct objects

Arrange n distinct objects

Michelle Jacob Groucho Isabel Kathleen

Arrange n distinct objects

1st 2nd 3rd 4th 5th

Steps:

1. Choose 1st can — 5 options
2. Choose 2nd can — 4 options

...

5. Choose 5th can — 1 option

$$\text{Total} = 5 \times 4 \times 3 \times 2 \times 1$$
$$= 120$$

Permutations

CS106A has you compute these iteratively
CS106B has you compute these recursively
CS109 requires you count using them

A permutation is an ordered arrangement of objects.

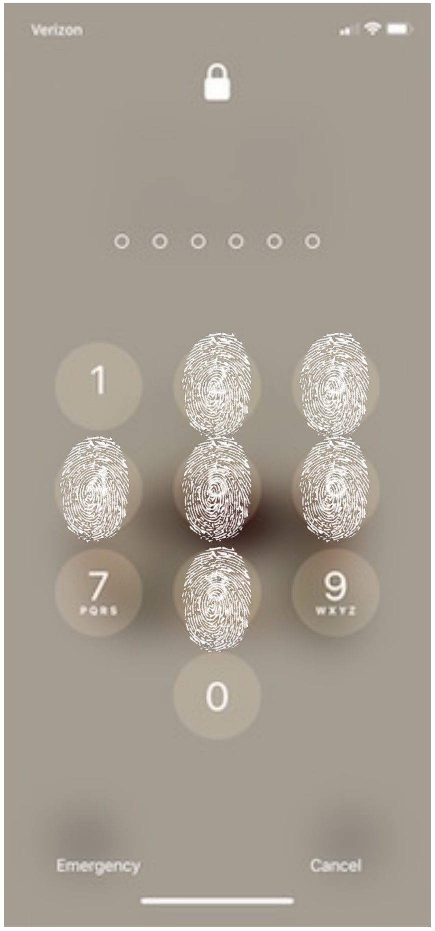
ordered means order is important

The number of unique orderings (permutations) of n distinct objects is

$$n! = n \times (n-1) \times (n-2) \times \cdots \times 2 \times 1$$

other notation for this: $n! = \prod_{k=1}^{n} k$

Unique 6-digit passcodes with six smudges



How many unique 6-digit passcodes are possible if a phone password uses each of **six** distinct numbers?

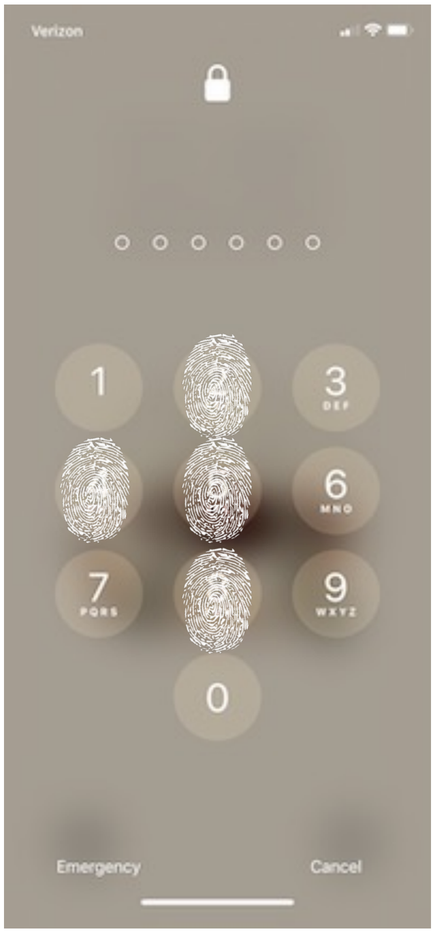
restated, how many ways can we permute 234568?

Total $= 6!$ *← this is just as good of an answer as 720*

$= 720$ passcodes

```
>>> import math
>>> math.factorial(6)
720
```

Unique 6-digit passcodes with four smudges



How many unique 6-digit passcodes are possible if a phone password uses each of four distinct numbers?

next time we'll break this counting problem into multiple categories and compute the full answer together.