



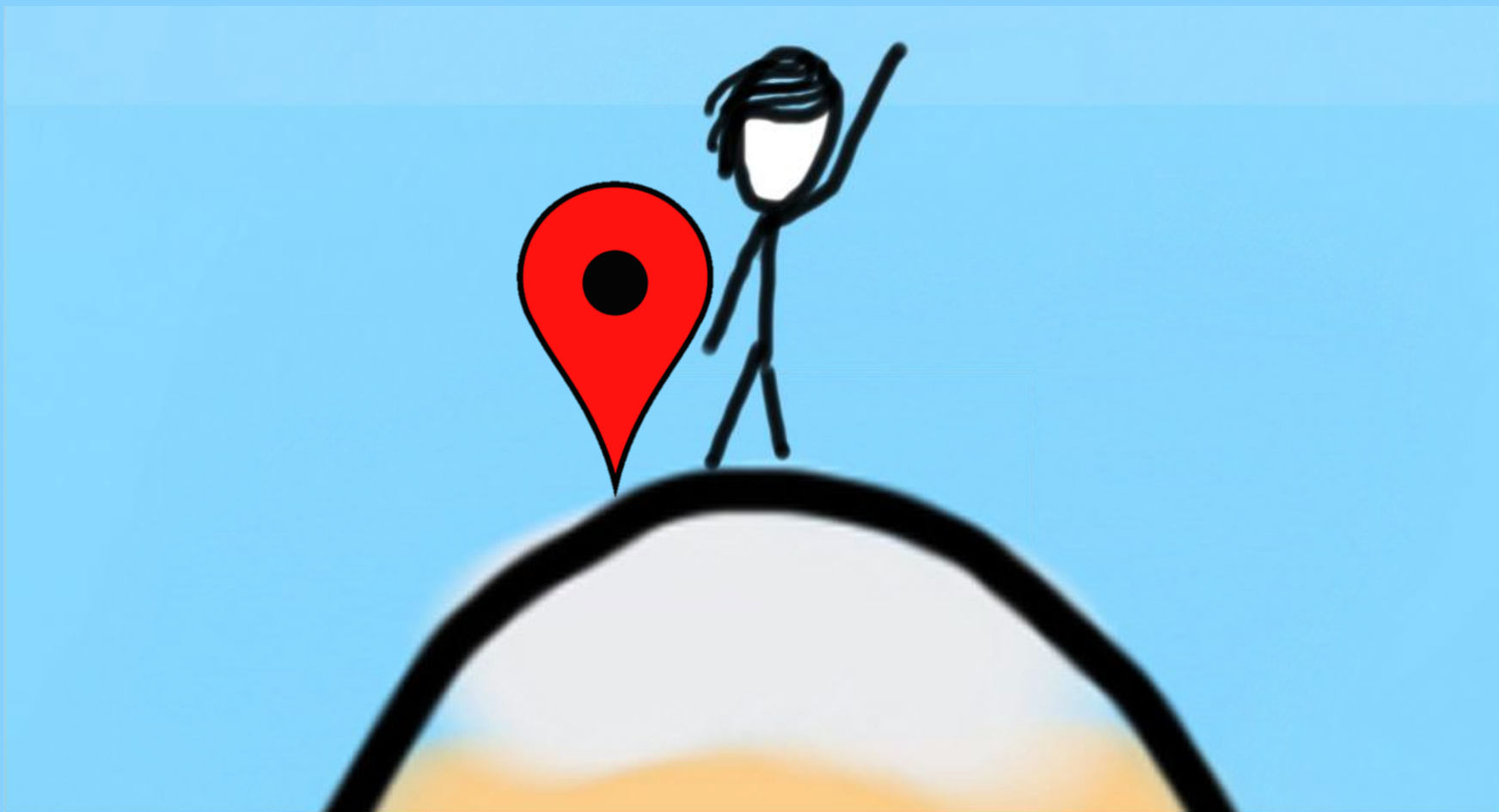
Algorithmic Analysis

Chris Piech

CS109, Stanford University

Learning Goals

1. Be able to compute conditional Expectation
2. Solve problems using the Law of Total Expectation
3. Use Conditional Expectation for Algorithmic Analysis



Pset 5 - Uncertainty Theory P x +

cs109psets.netlify.app/fall25/pset5/code_ana

PS5 Algorithmic Analysis: Dice Rolls

Consider the following function, which simulates repeated rolls of a die (where each integer value from 1 to 6 is equally likely) until a value ≥ 3 is "rolled".

```
def roll():
    total = 0
    while True:
        # equally likely to return 1,...,6
        roll = random_integer(1, 6)
        total += roll
        # exit condition
        if roll >= 3: break
    return total
```

Let X be the value returned by the function `roll()`. What is the expected return value of `roll()`?

10

Previous Question

Pset 5 - Uncertainty Theory P x +

cs109psets.netlify.app/fall25/pset5/code_analysis_flu

PS5 Llama Flu

Our ability to fight contagious diseases depends on our ability to model their spread. One person is exposed to llama-flu (a made up disease). The method below returns the number of individuals who will get infected.

```
def num_infected():
    """
    Returns the number of people infected by one individual
    """

    # most people are immune to llama-flu
    immune = bernoulli(p = 0.99)
    if immune: return 0

    # people who are not immune spread the disease far
    spread = 0

    # they make contact with k people (up to 100)
    k = binomial(n = 100, p = 0.25)
    for i in range(k):
        spread += num_infected()

    # total infections should include this individual
    return spread + 1
```

What is the expected return value of `num_infected()`?

11

Previous Question

https://cs109psets.netlify.app/fall25/pset5/code_analysis_flu

Pset 5 - Uncertainty Theory P x +

cs109psets.netlify.app/fall25/pset5/approx_counting

PS5 Approximate Counting

What if you needed a counter that could count up to the number of atoms in the universe, but you wanted to store the counter in 8 bits? You could use the algorithm below:

```
def stochastic_counter(true_count):
    n = -1
    for i in range(true_count):
        n += count(n)
    return 2 ** n # 2^n, aka 2 to the power of n

def count(n):
    # To return 1 you need n heads. Always returns 1 if n is <= 0
    for i in range(n):
        if not coin_flip():
            return 0
    return 1

def coin_flip():
    # returns true 50% of the time
    return random.random() < 0.5
```

Show that the expected return value of `stochastic_counter(4)`, where `count` is called 4 times, is in fact equal to 4.

12

Previous Question

Next Question

https://cs109psets.netlify.app/fall25/pset5/approx_counting

Def: Conditional Expectation

$$E[X|Y = y] = \sum_x xP(X = x|Y = y)$$

Def: Law of Total Expectation

$$E[X] = \sum_y E[X|Y = y]P(Y = y)$$

<review>

Expectation

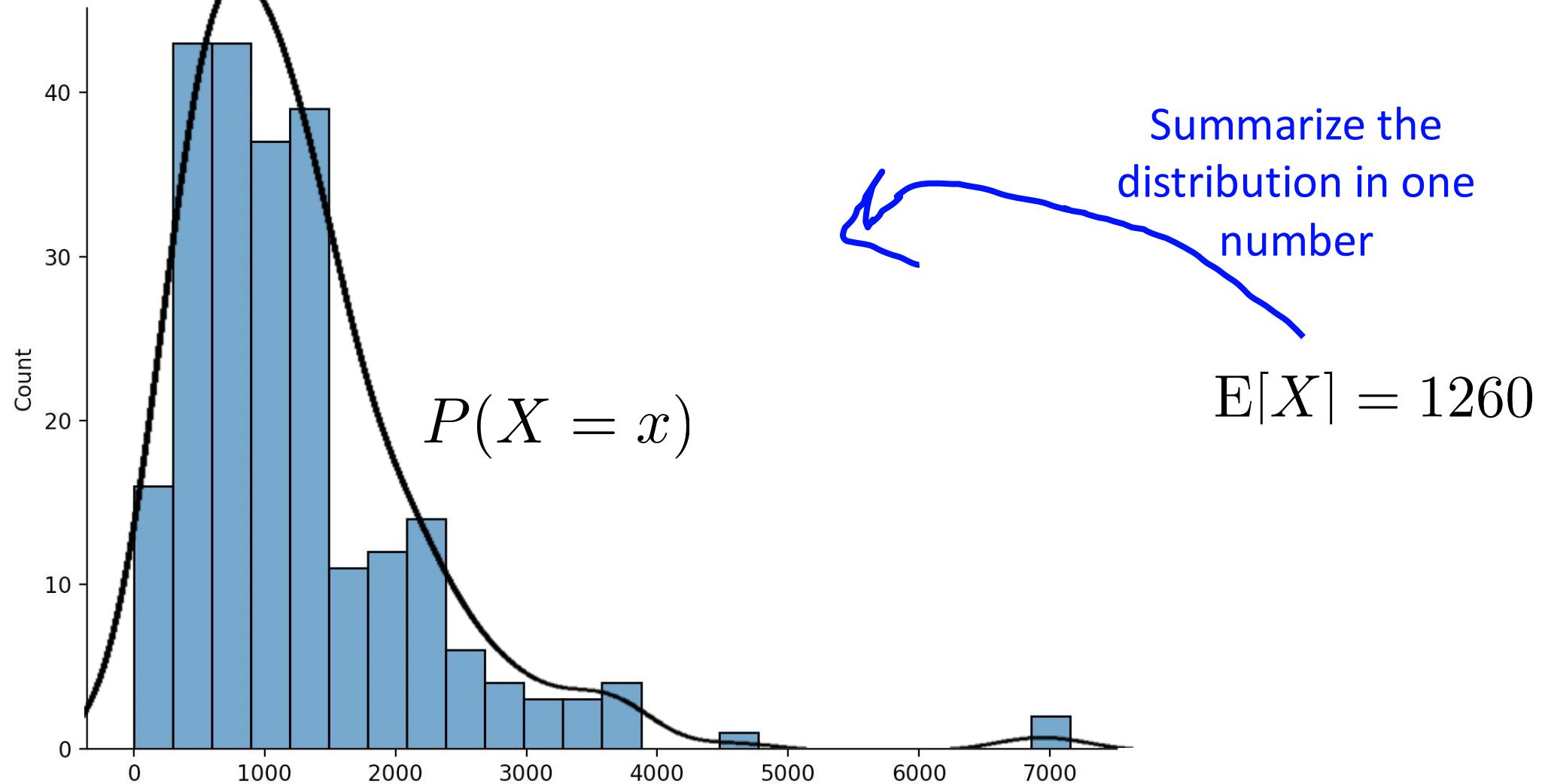
$$E[X] = \sum_x x \cdot P(X = x)$$

The probability that X takes on that value

All the values that X can take on

Limitation of Expectation

X = time to complete the medical diagnosis problem (in seconds)



Expectation of a Sum

$$E[X + Y] = E[X] + E[Y]$$

Generalized:

$$E\left[\sum_{i=1}^n X_i\right] = \sum_{i=1}^n E[X_i]$$

Holds regardless of dependency between X_i 's

Expectation of a Function

Law of unconscious statistician

$$\mathbb{E}[g(X)] = \sum_x g(x) \cdot P(X = x)$$

So for example...

$$\mathbb{E}[X^2] = \sum_x x^2 \cdot P(X = x)$$

End Review

Four practice examples to warm us up



Boole was Cool

Let E_1, E_2, \dots, E_n be events with indicator RVs X_i

- If event E_i occurs, then $X_i = 1$, else $X_i = 0$
- Recall $E[X_i] = P(E_i)$

- Why?

$$E[X_i] = 0 \cdot (1 - P(E_i)) + 1 \cdot P(E_i)$$

Bernoulli aka Indicator Random Variables were studied extensively by George Boole

Boole died of being too cool



Differential Privacy

Aims to provide means to **maximize the accuracy** of probabilistic queries while minimizing the **probability** of identifying its records.



Cynthia Dwork's celebrity lookalike is Cynthia Dwork.

Differential Privacy

100 independent values $X_1 \dots X_{100}$ where $X_i \sim \text{Bern}(p)$

Y_i
What is
returned

Maximize accuracy, while preserving privacy.

def calculateYi(Xi):

obfuscate = random()

if obfuscate:

return indicator(random())

else:

return Xi

random() returns True or
False with equal likelihood

Differential Privacy

100 independent values $X_1 \dots X_{100}$ where $X_i \sim \text{Bern}(p)$

Y_i
What is
returned

```
# Maximize accuracy, while preserving privacy.
```

```
def calculateYi(Xi):
```

```
    obfuscate = random()
```

```
    if obfuscate:
```

```
        return indicator(random())
```

```
    else:
```

```
        return Xi
```

random() returns True or
False with equal likelihood

What is $E[Y_i]$?

$$E[Y_i] = P(Y_i = 1) = \frac{p}{2} + \frac{1}{4}$$

Differential Privacy

100 independent values $X_1 \dots X_{100}$ where $X_i \sim \text{Bern}(p)$

Y_i
What is
returned

Maximize accuracy, while preserving privacy.

def calculateYi(Xi):

obfuscate = random()

if obfuscate:

return indicator(random())

else:

return Xi

random() returns True or
False with equal likelihood

Let $Z = \sum_{i=1}^{100} Y_i$

What is $E[Z]$?

$$E[Z] = E\left[\sum_{i=1}^{100} Y_i\right] = \sum_{i=1}^{100} E[Y_i] = \sum_{i=1}^{100} \left(\frac{p}{2} + \frac{1}{4}\right) = 50p + 25$$

Differential Privacy

100 independent values $X_1 \dots X_{100}$ where $X_i \sim \text{Bern}(p)$

Y_i
What is
returned

```
# Maximize accuracy, while preserving privacy.
```

```
def calculateYi(Xi):
```

```
    obfuscate = random()
```

```
    if obfuscate:
```

```
        return indicator(random())
```

```
    else:
```

```
        return Xi
```

random() returns True or
False with equal likelihood

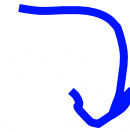
Let $Z = \sum_{i=1}^{100} Y_i$ $E[Z] = 50p + 25$ How do you estimate p ?

$$p \approx \frac{Z - 25}{50}$$

Challenge: What is the probability that our estimate is good?

Differential Privacy

Story which continues to unfold...



Generalization in Adaptive Data Analysis and Holdout Reuse*

Cynthia Dwork
Microsoft Research

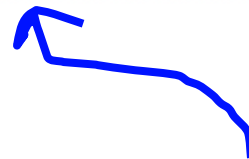
Vitaly Feldman
IBM Almaden Research Center[†]

Moritz Hardt
Google Research

Toniann Pitassi
University of Toronto

Omer Reingold
Samsung Research America

Aaron Roth
University of Pennsylvania



Professor at Stanford

What is Amazon?

amazon

The Amazon logo consists of the word "amazon" in a bold, lowercase, sans-serif font. Below the word is a thick, orange curved arrow that starts under the 'a' and ends under the 'n', pointing to the right.

Cape Town, 2006



What is Amazon?



* ~~52%~~ 74% of Amazons Profits

**More profitable than Amazon's North America commerce operations

<https://www.fool.com/investing/2022/07/07/aws-chief-says-amazons-most-profitable-segment-is/>

Computer Cluster Utilization

Computer cluster with k servers

- Requests independently go to server i with probability p_i
- Let event A_i = server i receives no requests
- X = # of events A_1, A_2, \dots, A_k that occur
- $E[X]$ after first n requests?

Since X is an expectation,
can you express it as a
sum?

-
- Let Bernoulli B_i be an **indicator** for A_i $X = \sum_{i=1}^k B_i$
 - Since requests independent: $P(A_i) = (1 - p_i)^n$

$$E[X] = E\left[\sum_{i=1}^k B_i\right] = \sum_{i=1}^k E[B_i] = \sum_{i=1}^k P(A_i) = \sum_{i=1}^k (1 - p_i)^n$$



When stuck, brainstorm
about random variables



Toy Collection

You are trying to collect n distinct toys

- Each purchase, each toy is equally likely
- Let $X = \#$ purchases until you have ≥ 1 of each toy. What is $E[X]$?

Let $X_i = \#$ of **trials to get success after i -th success** where “success” is getting an unseen toy.

$$X = X_0 + X_1 + \dots + X_{n-1} \Rightarrow E[X] = E[X_0] + E[X_1] + \dots + E[X_{n-1}]$$

After i successes, the probability of the next success is $p = (n - i) / n$

$$X_i \sim \text{Geo}(p = (n - i) / n)$$

$$E[X_i] = 1 / p = n / (n - i)$$

$$E[X] = \frac{n}{n} + \frac{n}{n-1} + \frac{n}{n-2} + \dots + \frac{n}{1} = n \left[\frac{1}{n} + \frac{1}{n-1} + \dots + 1 \right] = O(n \log n)$$



Conditional Expectation

Conditional Expectation

X and Y are discrete random variables:

$$E[X|Y = y] = \sum_x xP(X = x|Y = y)$$



Analogously, continuous random variables:

$$E[X|Y = y] = \int_x xP(X = x|Y = y)$$

Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

Roll two 6-sided dice D_1 and D_2

- $X = \text{value of } D_1 + D_2$ $Y = \text{value of } D_2$
- What is $E[X | Y = 6]$?

$$\begin{aligned} E[X | Y = 6] &= \sum_x x P(X = x | Y = 6) \\ &= \left(\frac{1}{6}\right)(7 + 8 + 9 + 10 + 11 + 12) = \frac{57}{6} = 9.5 \end{aligned}$$

- Intuitively makes sense: $6 + E[\text{value of } D_1] = 6 + 3.5$

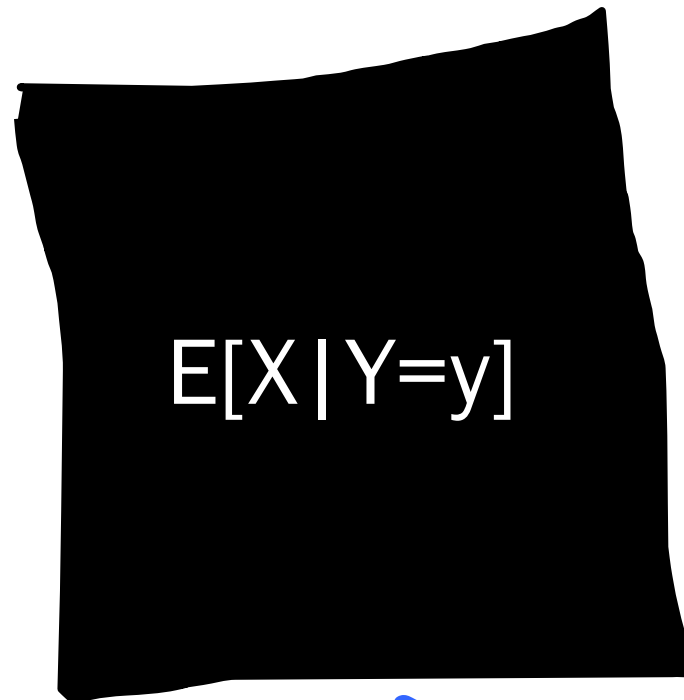
Conditional Expectation as a Function

Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

Define $g(Y) = E[X | Y]$

This is a function of Y



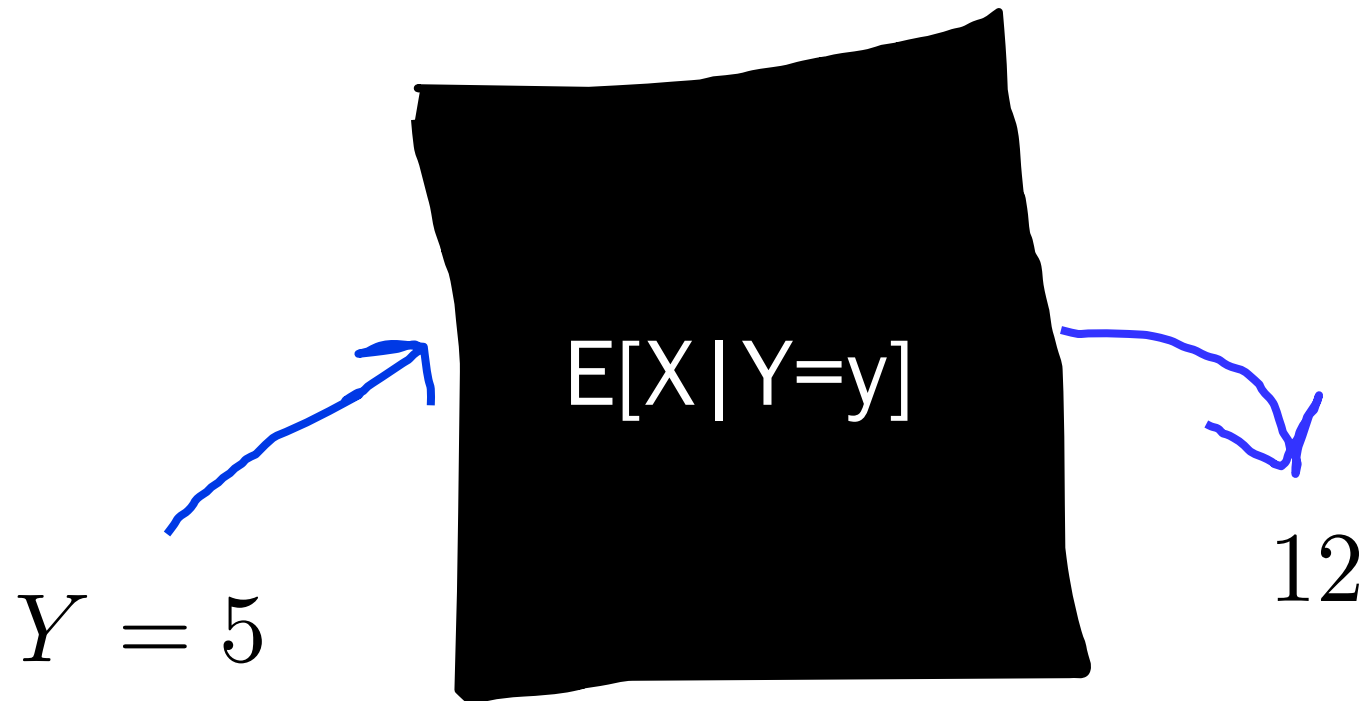
This is a function with Y as input

Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

Define $g(Y) = E[X | Y]$

This is a function of Y

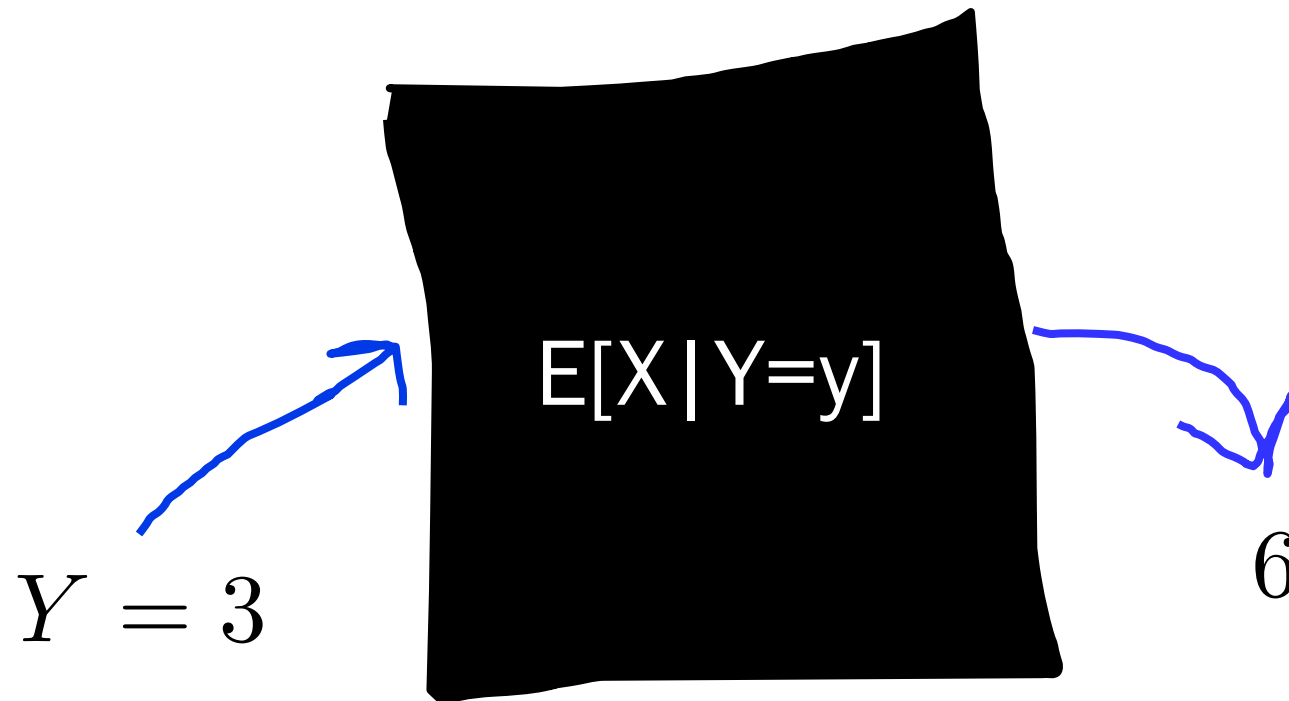


Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

Define $g(Y) = E[X | Y]$

This is a function of Y



Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

This is a number:

$$E[X]$$



This is a function of y :

$$E[X|Y = y]$$

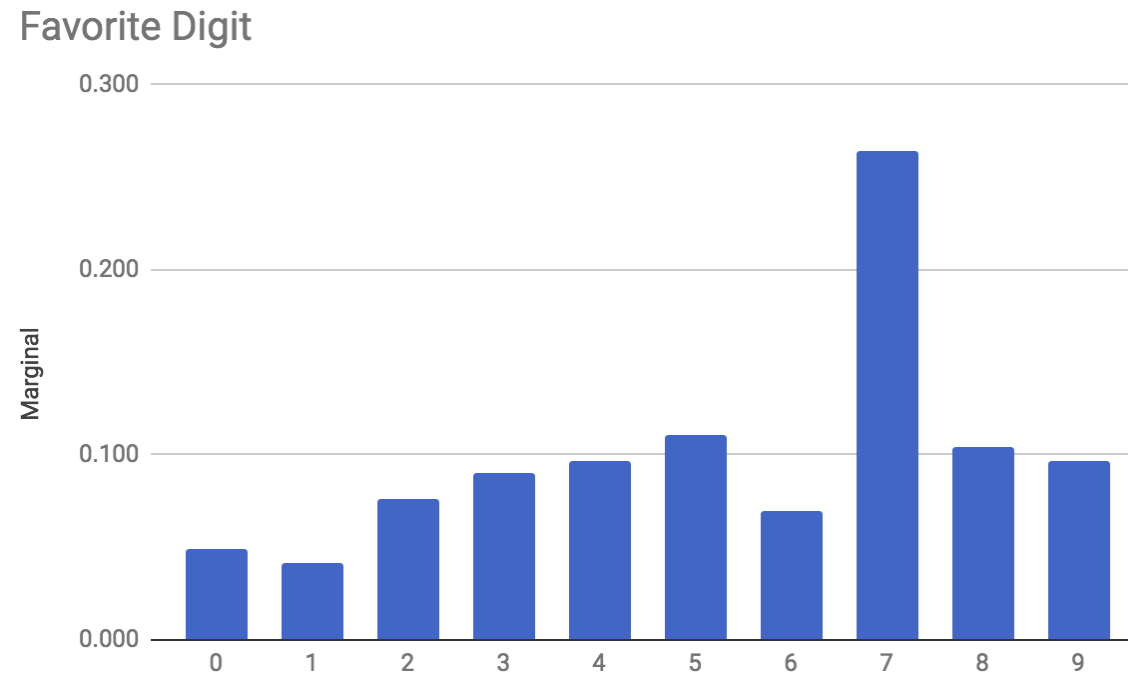
$$E[X = 5]$$

Doesn't make sense. Take expectation of random variables, not events

Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

X = favorite number
Y = year in school



$$E[X] = 0 * 0.05 + \dots + 9 * 0.10 = 5.38$$

Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

X = favorite number

Y = year in school

$E[X | Y] ?$

Year in school, $Y = y$	$E[X Y = y]$
2	5.5
3	5.8
4	6.0
5	4.7

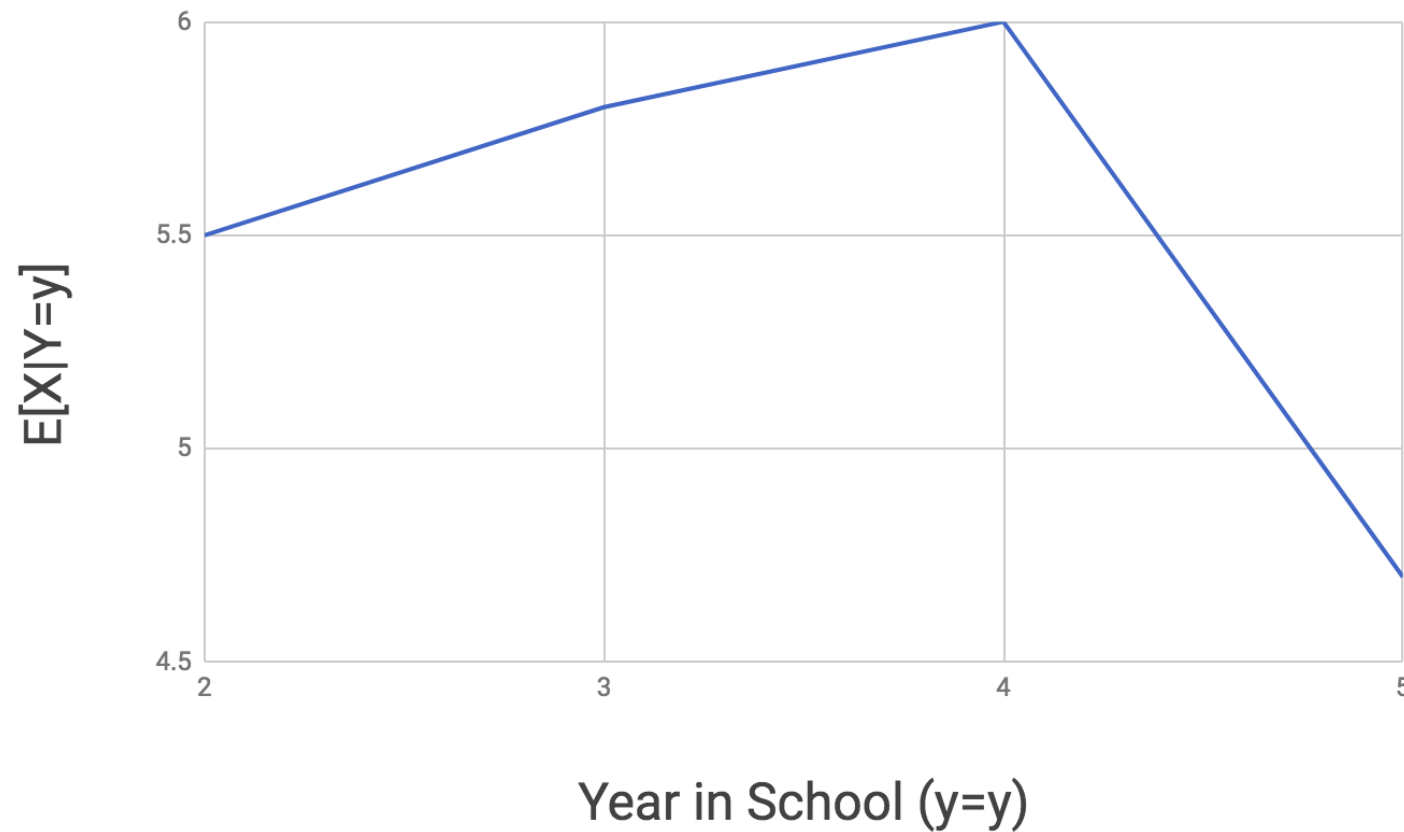
Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

X = favorite number

Y = year in school

$E[X | Y] ?$

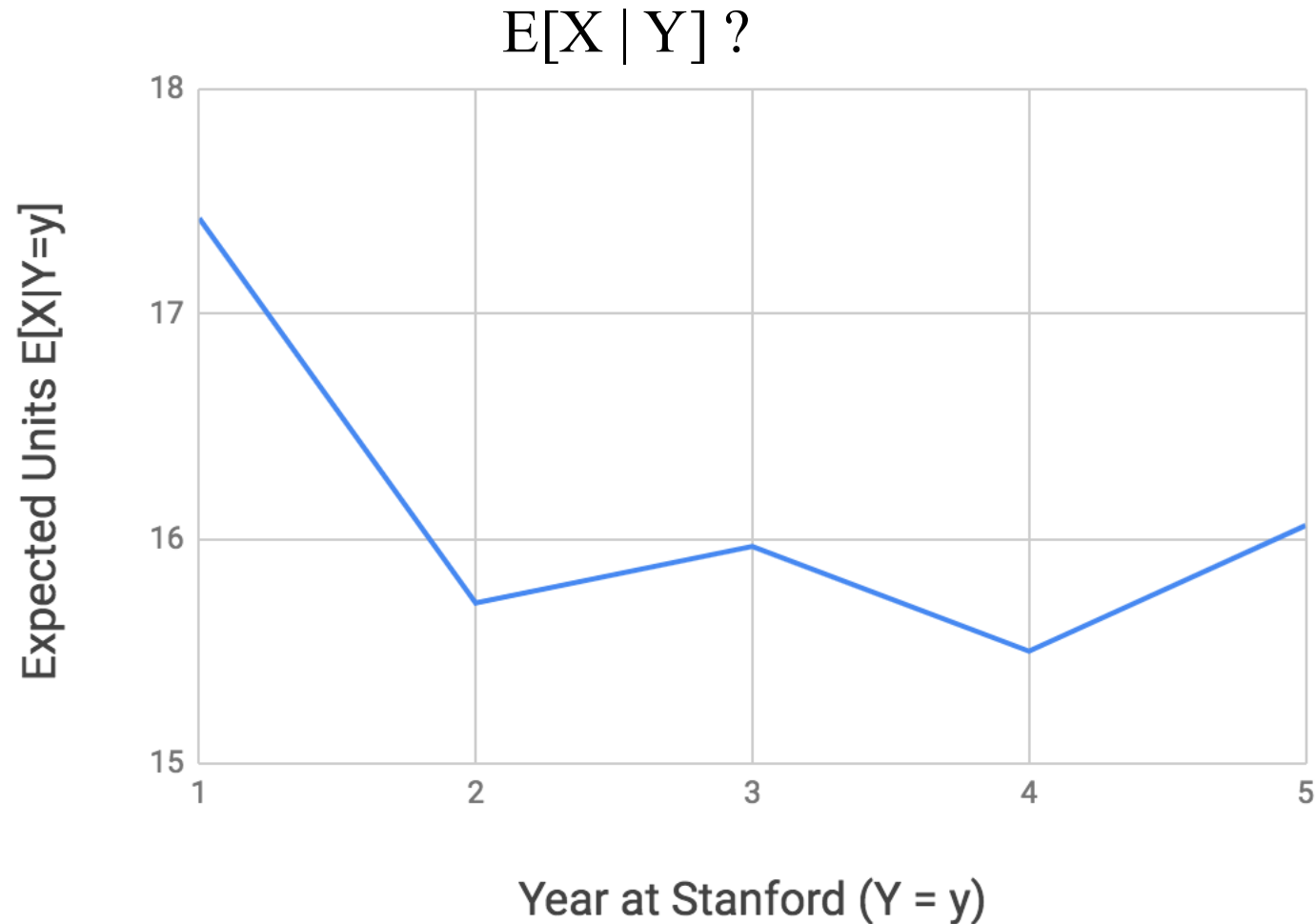


Conditional Expectation

$$E[X|Y = y] = \sum_x x \cdot P(X = x|Y = y)$$

X = units in fall quarter

Y = year in school



Want to see something cool?

What the heck does this give you?

$$\sum_y E[X|Y = y]P(Y = y)$$

Law of Total Expectation

$$E[X] = \sum_y E[X|Y = y]P(Y = y)?$$

$$\sum_y E[X|Y = y]P(Y = y) = \sum_y \sum_x xP(X = x|Y = y)P(Y = y)$$

Def of $E[X|Y]$

Law of Total Expectation

$$E[X] = \sum_y E[X|Y = y]P(Y = y)?$$

$$\begin{aligned} \sum_y E[X|Y = y]P(Y = y) &= \sum_y \sum_x xP(X = x|Y = y)P(Y = y) && \text{Def of } E[X|Y] \\ &= \sum_y \sum_x xP(X = x, Y = y) && \text{Chain rule!} \end{aligned}$$

Law of Total Expectation

$$E[X] = \sum_y E[X|Y = y]P(Y = y)?$$

$$\sum_y E[X|Y = y]P(Y = y) = \sum_y \sum_x xP(X = x|Y = y)P(Y = y) \quad \text{Def of } E[X|Y]$$

$$= \sum_y \sum_x xP(X = x, Y = y) \quad \text{Chain rule!}$$

$$= \sum_x \sum_y xP(X = x, Y = y) \quad \text{I switch the order of the sums}$$

Law of Total Expectation

$$E[X] = \sum_y E[X|Y = y]P(Y = y)?$$

$$\sum_y E[X|Y = y]P(Y = y) = \sum_y \sum_x xP(X = x|Y = y)P(Y = y) \quad \text{Def of } E[X|Y]$$

$$= \sum_y \sum_x xP(X = x, Y = y) \quad \text{Chain rule!}$$

$$= \sum_x \sum_y xP(X = x, Y = y) \quad \text{I switch the order of the sums}$$

$$= \sum_x x \sum_y P(X = x, Y = y) \quad \text{Move that } x \text{ outside the } y \text{ sum}$$

Law of Total Expectation

$$E[X] = \sum_y E[X|Y = y]P(Y = y)?$$

$$\sum_y E[X|Y = y]P(Y = y) = \sum_y \sum_x xP(X = x|Y = y)P(Y = y) \quad \text{Def of } E[X|Y]$$

$$= \sum_y \sum_x xP(X = x, Y = y) \quad \text{Chain rule!}$$

$$= \sum_x \sum_y xP(X = x, Y = y) \quad \text{I switch the order of the sums}$$

$$= \sum_x x \sum_y P(X = x, Y = y) \quad \text{Move that } x \text{ outside the } y \text{ sum}$$

$$= \sum_x xP(X = x) \quad \text{Marginalization}$$

Law of Total Expectation

$$E[X] = \sum_y E[X|Y = y]P(Y = y)?$$

$$\sum_y E[X|Y = y]P(Y = y) = \sum_y \sum_x xP(X = x|Y = y)P(Y = y) \quad \text{Def of } E[X|Y]$$

$$= \sum_y \sum_x xP(X = x, Y = y) \quad \text{Chain rule!}$$

$$= \sum_x \sum_y xP(X = x, Y = y) \quad \text{I switch the order of the sums}$$

$$= \sum_x x \sum_y P(X = x, Y = y) \quad \text{Move that } x \text{ outside the } y \text{ sum}$$

$$= \sum_x xP(X = x) \quad \text{Marginalization}$$

$$= E[X] \quad \text{Def of } E[X]$$

Law of Total Expectation

Law of Total Expectation



For any discrete random variable X
and any discrete random variable Y

$$E[X] = \sum_y E[X|Y = y]P(Y = y)$$

Recall the Law of Total Probability

$$P(X = x) = \sum_y P(X = x|Y = y)P(Y = y)$$

NETFLIX

(The Streaming Part)

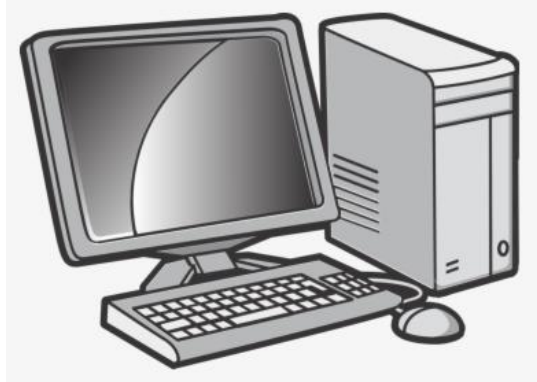
How long does this code take to run?

Netflix streams millions of hours of videos per day. They REALLY care about the speed of the following code:

```
database.get_movie(movie_name)
```

How long does this line of code take? Say 512 MB movie.

How long does this code take to run?

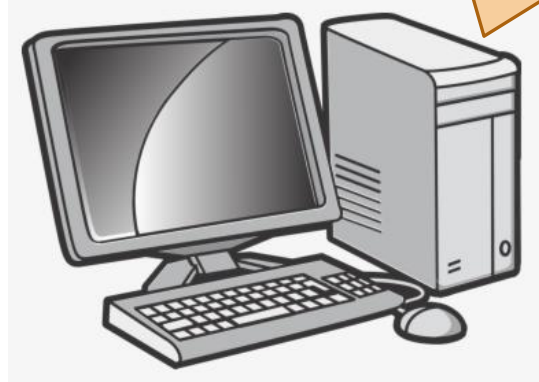


```
database.get_movie(movie_name)
```

1. 0.3s
2. 1.6 mins
3. 5 mins
4. 2 hours

Millisecond Latency

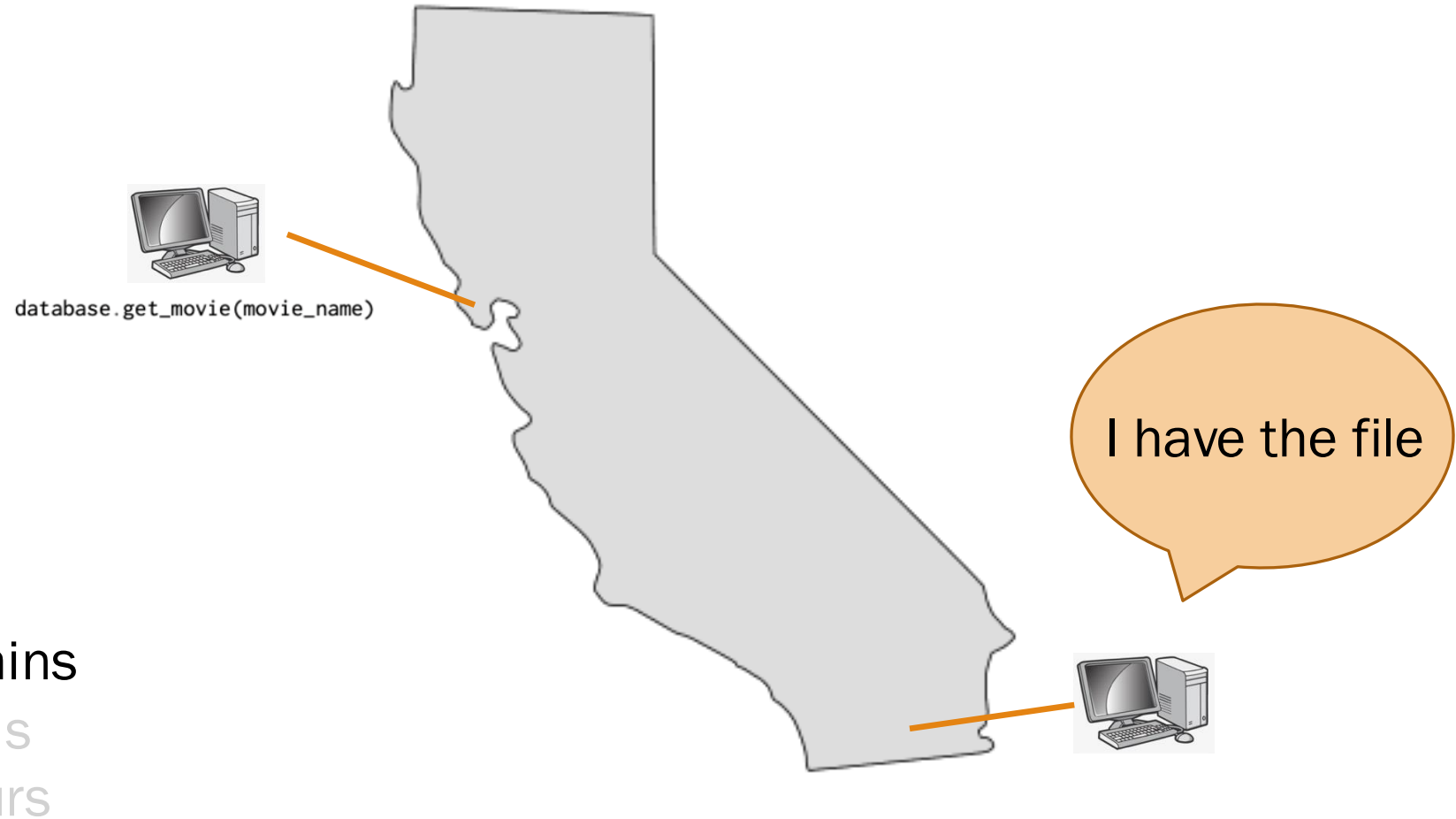
Its in Palo Alto



```
database.get_movie(movie_name)
```

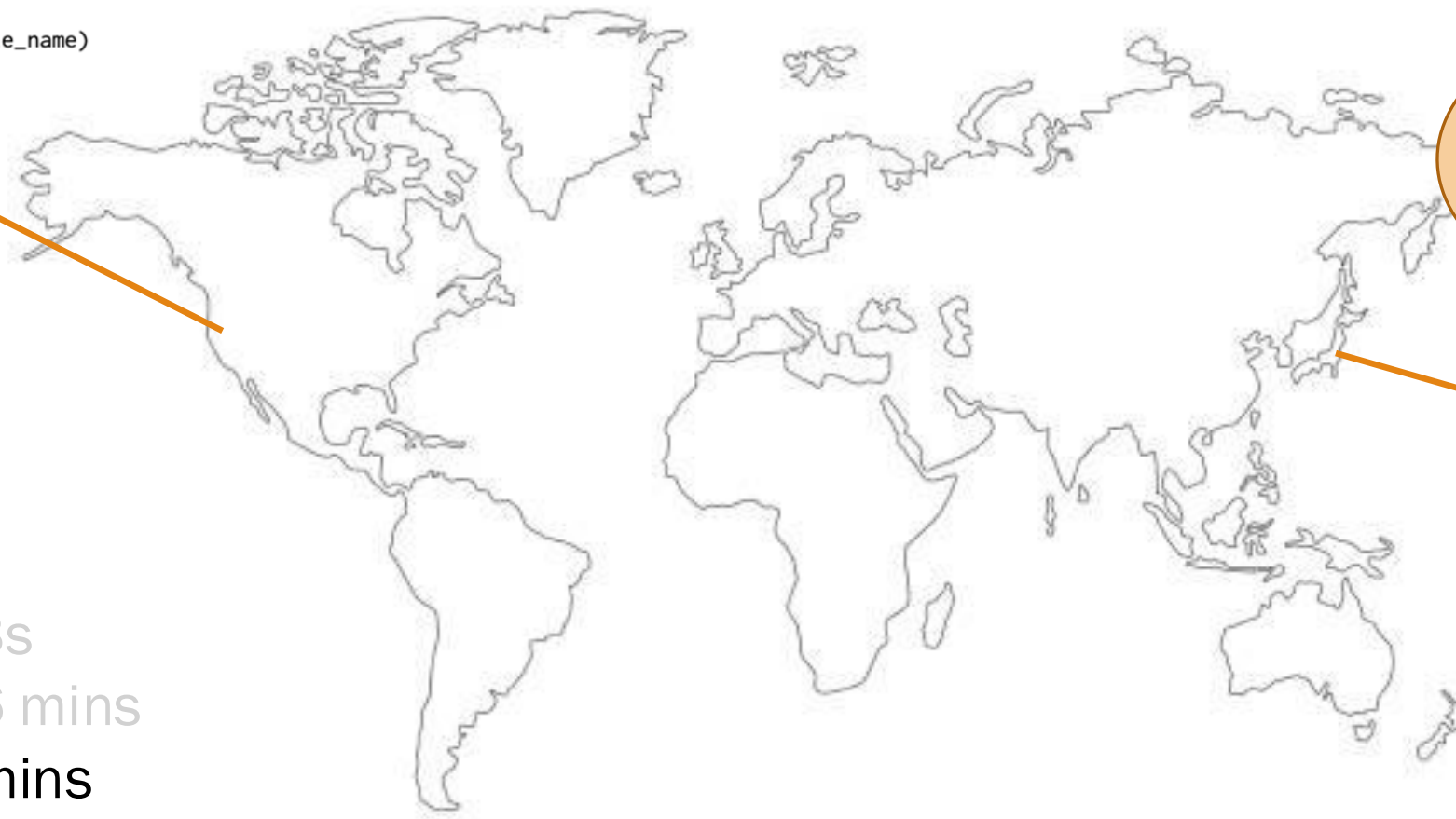
1. 0.3s
2. 1.6 mins
3. 5 mins
4. 2 hours

Minute Latency



Many Minutes Latency

database.get_movie(movie_name)



私はファイルを持っています



1. 0.3s
2. 1.6 mins
3. 5 mins
4. 2 hours

Are we done?

```
database.get_movie(movie_name)
```



5mins across the world!!!!!!

2 hours



```
database.get_movie(movie_name)
```



Expected Run Time

Expected runtime

Location, l	$E[\text{Runtime} \mid \text{Location} = l]$	$P(\text{Location} = l)$
Palo Alto	0.3s	0.10
SoCal	1.6s	0.50
Japan	300.0s	0.37
Space	7200.0s	0.03

Let Location (L for short) be the location where the movie file is stored

What is the expected runtime of `database.get_movie(movie_name)`

Using the Law of Total Expectation

$$\begin{aligned} E[\text{Runtime}] &= E[\text{Runtime} \mid L = \text{Palo Alto}] \cdot P(L = \text{Palo Alto}) \\ &\quad + E[\text{Runtime} \mid L = \text{SoCal}] \cdot P(L = \text{SoCal}) \\ &\quad + E[\text{Runtime} \mid L = \text{Japan}] \cdot P(L = \text{Japan}) \\ &\quad + E[\text{Runtime} \mid L = \text{Space}] \cdot P(L = \text{Space}) = 327s \end{aligned}$$

Analyze Recursive Code

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by **Recurse ()**. **What is $E[Y]$?**

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by **Recurse ()**. **What is $E[Y]$?**

$$E[Y] = E[Y | X = 1]P(X = 1) + E[Y | X = 2]P(X = 2) + E[Y | X = 3]P(X = 3)$$

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by **Recurse ()**. **What is $E[Y]$?**

$$E[Y] = E[Y | X = 1]P(X = 1) + E[Y | X = 2]P(X = 2) + E[Y | X = 3]P(X = 3)$$



$\frac{1}{3}$



$\frac{1}{3}$



$\frac{1}{3}$

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by `Recurse()`. **What is $E[Y]$?**

$$E[Y] = E[Y | X = 1]P(X = 1) + E[Y | X = 2]P(X = 2) + E[Y | X = 3]P(X = 3)$$

3



$\frac{1}{3}$



$\frac{1}{3}$



$\frac{1}{3}$

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by **Recurse ()**. **What is $E[Y]$?**

$$E[Y] = E[Y | X = 1]P(X = 1) + E[Y | X = 2]P(X = 2) + E[Y | X = 3]P(X = 3)$$

3

$\frac{1}{3}$

$(5 + E[Y])$

$\frac{1}{3}$

$\frac{1}{3}$

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by **Recurse ()**. **What is $E[Y]$?**

$$E[Y] = E[Y | X = 1]P(X = 1) + E[Y | X = 2]P(X = 2) + E[Y | X = 3]P(X = 3)$$

3

$\frac{1}{3}$

$(5 + E[Y])$

$\frac{1}{3}$

$(7 + E[Y])$

$\frac{1}{3}$

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by **Recurse ()**. **What is $E[Y]$?**

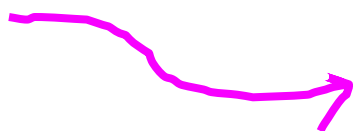
$$\begin{aligned} E[Y] &= E[Y \mid X = 1]P(X = 1) + E[Y \mid X = 2]P(X = 2) + E[Y \mid X = 3]P(X = 3) \\ &= 3 \cdot \frac{1}{3} + (5 + E[Y]) \cdot \frac{1}{3} + (7 + E[Y]) \cdot \frac{1}{3} \end{aligned}$$

Analyzing Recursive Code

```
int Recurse() {  
    int x = randomInt(1, 3); // Equally likely values  
  
    if (x == 1) return 3;  
    else if (x == 2) return (5 + Recurse());  
    else return (7 + Recurse());  
}
```

Let Y = value returned by **Recurse()**. **What is $E[Y]$?**

$$\begin{aligned} E[Y] &= E[Y \mid X = 1]P(X = 1) + E[Y \mid X = 2]P(X = 2) + E[Y \mid X = 3]P(X = 3) \\ &= 3 \cdot \frac{1}{3} + (5 + E[Y]) \cdot \frac{1}{3} + (7 + E[Y]) \cdot \frac{1}{3} \\ &= \frac{15}{3} + \frac{2}{3}E[Y] \end{aligned}$$

 $E[Y] = 15$

If we have time...

Hiring an Engineer

Your company has **one job opening** for a software engineer.

You have n candidates. But you have to say yes/no **immediately** after each interview!

Proposed algorithm: reject the first k and accept the next one who is better than all of them.

What's the best value of k ?



Aka The Secretary Problem
Aka The Optimal Stopping Problem
Aka The Marriage Problem

Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule

What is the $P(B|X = i)$?

Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule

What is the $P(B|X = i)$?



Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule

What is the $P(B|X = i)$?



Hint: where is the
best among the first
 $i - 1$ candidates?

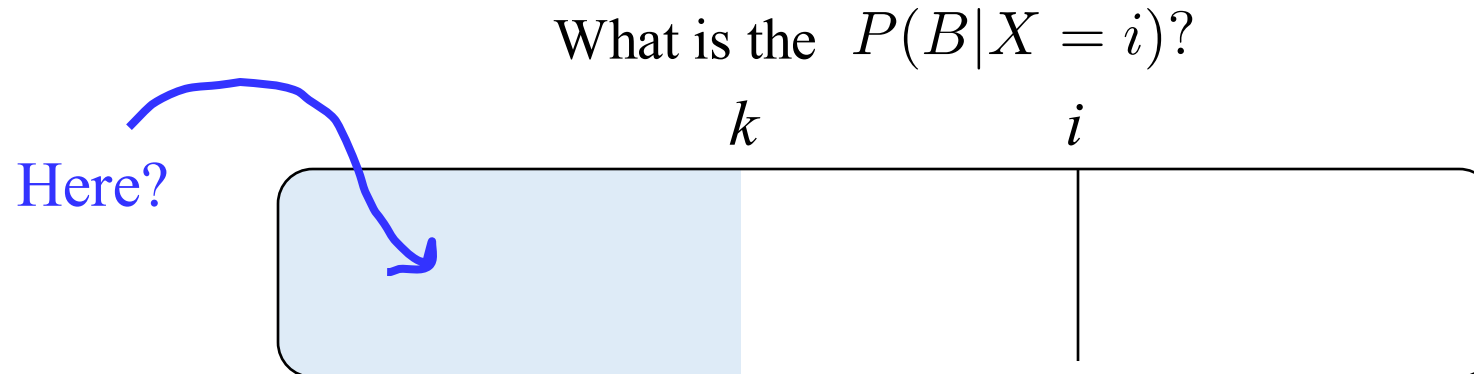


Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule



Hint: where is the best among the first $i - 1$ candidates?

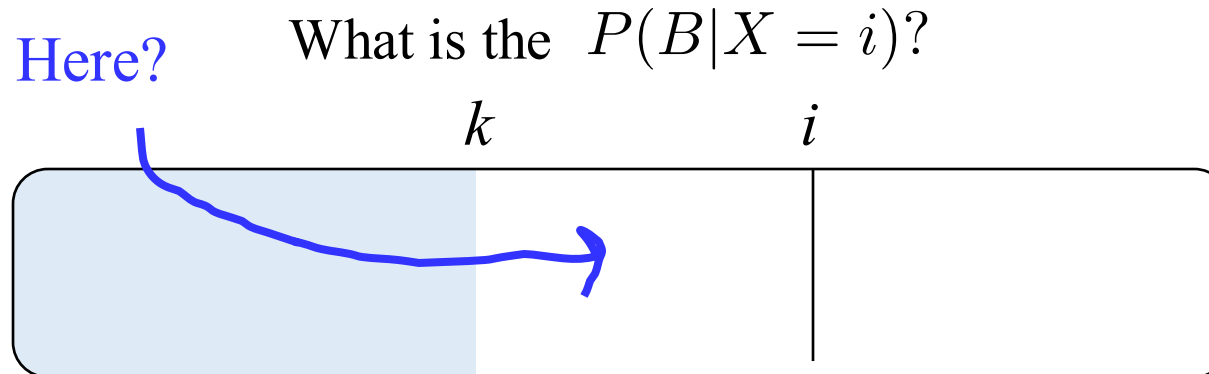


Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule



Hint: where is the
best among the first
 $i - 1$ candidates?



Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule

$$P(B|X = i) = \frac{k}{i-1} \text{ if } i > k$$



Hint: where is the best among the first $i-1$ candidates?



Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule

$$P_k(B) = \sum_{i=1}^n P_k(B|X = i)P(X = i)$$

By the law of total expectation

$$= \frac{1}{n} \sum_{i=1}^n P_k(B|X = i)$$

$$= \frac{1}{n} \sum_{i=k+1}^n \frac{k}{i-1}$$

since we know $P_k(\text{Best}|X = i)$

$$\approx \frac{1}{n} \int_{i=k+1}^n \frac{k}{i-1} di$$

By Riemann Sum approximation

$$= \frac{k}{n} \ln(i-1) \Big|_{k+1}^n = \frac{k}{n} \ln \frac{n-1}{k} \approx \frac{k}{n} \ln \frac{n}{k}$$

Hiring and Engineer

n candidates, must say yes/no **immediately** after each interview.
Reject the first k , accept the next who is better than all of them.
What's the best value of k ?

B: event that you hire the best engineer

X: position of the best engineer on the interview schedule

$$P_k(B) = \sum_{i=1}^n P_k(B|X = i)P(X = i)$$

By the law of total expectation

$$\approx \frac{k}{n} \ln \frac{n}{k}$$

Fun fact. Optimized when: $k = \frac{n}{e}$

Uncertainty Theory

Beta
Distributions

Thompson
Sampling

Adding
Random Vars

Central Limit
Theorem

Sampling

Bootstrapping

Algorithmic
Analysis


Information
Theory

As requested by CS faculty



Where are we in CS109?


On Friday...


Counting
Theory


Core
Probability

x_2
Random
Variables


Probabilistic
Models


Uncertainty
Theory


Machine
Learning