Pset #5 is out
Llama Flu

Heads up: This problem uses material that we are going to cover on Feb 28th.

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

```python
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 `code(num_infected())`?
Better Peer Grading

Stanford's HCI class runs a massive online class that was taken by ten thousand students. The class used peer assessment to evaluate student's work. We are going to use their data to learn more about peer graders. In the class, each student has their work evaluated by 5 peers and every student is asked to evaluate 6 assignments: five peers and the control assignment (the graders were un-aware of which assignment was the control). All 10,000 students evaluated the same control assignment and the scores they gave are in the file peerGrades.csv in the pset5 data zip:

pset5.zip

Would you use the mean or the median of 5 peer grades to assign scores in the online version of Stanford's HCI class? Explain why. You may use simulations to solve any part of this question. Hint: it might help to visualize the scores in peerGrades.csv. In order to make your decision compute the statistics in part a) and b).
The Null Hypothesis. How can we use bootstrapping here?

There is no difference between the two groups, so everyone is drawn from the same distribution. Any difference you observe is due to sampling error.
Algorithmic Analysis
NETFLIX

(The Streaming Part)
Review Expected Values
Expectation

\[ E[X] = \sum x \cdot P(X = x) \]

All the values that \( X \) can take on

The probability that \( X \) takes on that value
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
Limitation of Expectation

\[ X = \text{time a program takes to finish running (in milli seconds)} \]

\[ \Pr(X = x) \]

\[ \mathbb{E}[X] = 12600 \]

Summarize the distribution in one number.
Law of unconscious statistician

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

So for example...

$$E[X^2] = \sum_x x^2 \cdot P(X = x)$$
End Review
Let $E_1, E_2, \ldots, E_n$ 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
Expectation of the Binomial

Let $Y \sim \text{Bin}(n, p)$
- $n$ independent trials
- Let $X_i = 1$ if $i$-th trial is “success”, 0 otherwise
- $X_i \sim \text{Ber}(p)$  \hspace{1cm} $E[X_i] = p$

\[
Y = X_1 + X_2 + \cdots + X_n = \sum_{i=1}^{n} X_i
\]
\[
E[Y] = E[\sum_{i=1}^{n} X_i]
\]
\[
= \sum_{i=1}^{n} E[X_i]
\]
\[
= E[X_1] + E[X_2] + \ldots + E[X_n]
\]
\[
= np
\]
Let \( Y \sim \text{NegBin}(r, p) \)
- Recall \( Y \) is number of trials until \( r \) “successes”
- Let \( X_i = \# \) of trials to get success after \((i - 1)\)st success
- \( X_i \sim \text{Geo}(p) \) (i.e., Geometric RV)

\[
E[Y] = E[X_1 + X_2 + \cdots + X_r] = \sum_{i=1}^{r} E[X_i] = \frac{r}{p}
\]

\[
E[Y] = E[\sum_{i=1}^{r} X_i] = \sum_{i=1}^{r} E[X_i] = E[X_1] + E[X_2] + \cdots + E[X_r] = \frac{r}{p}
\]
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
- Let Bernoulli $B_i$ be an indicator for $A_i$
- $X =$ # of events $A_1, A_2, \ldots A_k$ that occur
- $Y =$ # servers that receive ≥ 1 request $= k - X$
- $E[Y]$ after first $n$ requests?
  - Since requests independent: $P(A_i) = (1 - p_i)^n$

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

\[
E[Y] = k - E[X] = k - \sum_{i=1}^{k} (1 - p_i)^n
\]
Amazon Monetized This

amazon

Stanford University
* 52% of Amazons Profits
**More profitable than Amazon’s North America commerce operations

Harder Practice!
When stuck, brainstorm about random variables
Hash Tables

Consider a hash table with \( n \) buckets
- Each string equally likely to get hashed into any bucket
- Let \( X \) = number of strings to hash until each bucket \( \geq 1 \) string
- What is \( E[X] \)?
Consider a hash table with \( n \) buckets
- Each string equally likely to get hashed into any bucket
- Let \( X = \) # strings to hash until each bucket \( \geq 1 \) string
- What is \( E[X] \)?

“Our 109 students are the best!”

Hash Tables

Round 1!
Consider a hash table with \( n \) buckets

- Each string equally likely to get hashed into any bucket
- Let \( X = \# \) strings to hash until each bucket \( \geq 1 \) string
- What is \( E[X] \)?

“Take care of yourselves!”

Round 2!
Hash Tables

Consider a hash table with \( n \) buckets
- Each string equally likely to get hashed into any bucket
- Let \( X = \# \) strings to hash until each bucket \( \geq 1 \) string
- What is \( E[X] \)?

"Don’t stress about exams!"

Round 3!
Hash Tables

Consider a hash table with $n$ buckets
- Each string equally likely to get hashed into any bucket
- Let $X = \#$ strings to hash until each bucket $\geq 1$ string
- What is $E[X]$?

“Know that we care!
Eat fruits!”

Round 4!
Consider a hash table with $n$ buckets
- Each string equally likely to get hashed into any bucket
- Let $X = \#$ strings to hash until each bucket $\geq 1$ string
- What is $E[X]$?

“We believe in you!”

Round 5!
Consider a hash table with \( n \) buckets

- Each string equally likely to get hashed into any bucket
- Let \( X = \# \) strings to hash until each bucket \( \geq 1 \) string
- What is \( E[X] \)?

“Everything will make sense!”

Round 6!
Consider a hash table with \( n \) buckets
- Each string equally likely to get hashed into any bucket
- Let \( X = \# \) strings to hash until each bucket \( \geq 1 \) string
- What is \( E[X] \)?

“You can do it!!!”

Round 7! – Done! Yay
Hash Tables

Consider a hash table with $n$ buckets

- Each string equally likely to get hashed into any bucket
- Let $X = \#$ strings to hash until each bucket $\geq 1$ string
- What is $E[X]$?

Hash Function

Expected Number of Rounds?
Consider a hash table with $n$ buckets

- Each string equally likely to get hashed into any bucket
- Let $X =$ # strings to hash until each bucket ≥ 1 string
- What is $E[X]$?
- Let $X_i =$ # of trials to get success after $i$-th success
  - where “success” is hashing string to previously empty bucket
  - After $i$ buckets have ≥ 1 string, probability of hashing a string to an empty bucket is $p = (n - i) / n$
Consider a hash table with $n$ buckets

- Each string equally likely to get hashed into any bucket
- Let $X = \#$ strings to hash until each bucket $\geq 1$ string
- What is $E[X]$?
- Let $X_i = \#$ of trials to get success after $i$-th success
  - where “success” is hashing string to previously empty bucket
  - After $i$ buckets have $\geq 1$ string, probability of hashing a string to an empty bucket is $p = (n - i) / n$

“Drink water!”

Round $i!$ (here, $i = 3$)
Consider a hash table with \( n \) buckets

- Each string equally likely to get hashed into any bucket
- Let \( X = \# \) strings to hash until each bucket \( \geq 1 \) string
- What is \( E[X] \)?
- Let \( X_i = \# \) of trials to get success after \( i \)-th success
  - where “success” is hashing string to previously empty bucket
  - After \( i \) buckets have \( \geq 1 \) string, probability of hashing a string to an empty bucket is \( p = (n - i) / n \)
  - equivalently: \( X_i \sim \text{Geo}((n - i) / n) \)

“Drink water!”

Hash Function

Round \( i! \) (here, \( i = 3 \))
Consider a hash table with $n$ buckets

- Each string equally likely to get hashed into any bucket
- Let $X =$ # strings to hash until each bucket $\geq 1$ string
- What is $E[X]$?

Let $X_i =$ # of trials to get success after $i$-th success

- where “success” is hashing string to previously empty bucket
- After $i$ buckets have $\geq 1$ string, probability of hashing a string to an empty bucket is $p = (n - i) / n$

- $P(X_i = k) = \frac{n-i}{n} \left(\frac{i}{n}\right)^{k-1}$ equivalently: $X_i \sim \text{Geo}((n - i) / n)$
- $E[X_i] = 1 / p = n / (n - i)$

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

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

This is your final answer
Conditional Expectation
X and Y are jointly discrete random variables

- Recall conditional PMF of X given Y = y:

\[ p_{X|Y}(x \mid y) = P(X = x \mid Y = y) = \frac{p_{X,Y}(x, y)}{p_Y(y)} \]

Define conditional expectation of X given Y = y:

\[ E[X \mid Y = y] = \sum_x xP(X = x \mid Y = y) = \sum_x xp_{X|Y}(x \mid y) \]

Analogously, jointly continuous random variables:

\[ f_{X|Y}(x \mid y) = \frac{f_{X,Y}(x, y)}{f_Y(y)} \quad E[X \mid Y = y] = \int_{-\infty}^{\infty} x f_{X|Y}(x \mid y) \, dx \]
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 $E[X \mid Y = 6]$?

\[
E[X \mid Y = 6] = \sum_x x P(X = x \mid Y = 6)
\]

\[
= \left(\frac{1}{6}\right)(7 + 8 + 9 + 10 + 11 + 12) = \frac{57}{6} = 9.5
\]

- Intuitively makes sense: $6 + E[\text{value of } D_1] = 6 + 3.5$
Define $g(Y) = E[X | Y]$

This is just a function of $Y$

Conditional Expectation

$$E[X|Y = y] = \sum_{x} x \cdot P(X = x | Y = y)$$
Conditional Expectation

- Define $g(Y) = \mathbb{E}[X | Y]$
- This is just function of $Y$

$\mathbb{E}[X | Y = y] = \sum_x x \cdot P(X = x | Y = y)$
Conditional Expectation

- Define $g(Y) = E[X | Y]$
- This is just function of $Y$

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

Stanford University
**Expectation**

\[ E[X|Y = y] = \sum_{x} x \cdot P(X = x|Y = y) \]

\( X = \text{favorite number} \)
\( Y = \text{year in school} \)

\[ E[X] = 0 \times 0.05 + \ldots + 9 \times 0.10 = 5.38 \]
Conditional Expectation

\[ E[X|Y = y] = \sum_{x} x \cdot P(X = x|Y = y) \]

**X** = favorite number  
**Y** = year in school

<table>
<thead>
<tr>
<th>Year in school, ( Y = y )</th>
<th>( E[X \mid Y = y] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>5.5</td>
</tr>
<tr>
<td>3</td>
<td>5.8</td>
</tr>
<tr>
<td>4</td>
<td>6.0</td>
</tr>
<tr>
<td>5</td>
<td>4.7</td>
</tr>
</tbody>
</table>
**Conditional Expectation**

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

- \(X\) = favorite number
- \(Y\) = year in school

**Graph:**

- \(E[X|Y] \) vs. Year in School (\(y = y\))
Conditional Expectation

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

\( X = \) units in fall quarter
\( Y = \) year in school

---

**Expected Units** \( E[X|Y=y] \)

**Year at Stanford** \( (Y = y) \)
Break
What is this???

$$E[E[X|Y]]$$

Function of Y
Law of Total Expectation

\[ E[E[X|Y]] = E[X] \]

\[
E[E[X|Y]] = \sum_y E[X|Y=y] P(Y=y) \quad \text{g(Y) = E[X|Y]}
\]

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

\[
= \sum_y \sum_x x P(X=x, Y=y) \quad \text{Chain rule!}
\]

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

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

\[
= \sum_x x P(X=x) \quad \text{Marginalization}
\]

\[
= E[X] \quad \text{Def of E[X]}
\]
Law of Total Expectation

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

$$E[X] = \sum_{y} E[X|Y = y]P(Y = y)$$
Speed of Code?
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:

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

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

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

All are correct! It is a RV!
How long does this code take to run?

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

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

I have the file
Minute Latency

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

I have the file
Many Minutes Latency

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

私はファイルを持っています
Are we done?

database.get_movie(movie_name)

5mins across the world!!!!!!
database.get_movie(movie_name)
Anyways

1. 0.3s
2. 1.6 mins
3. 5 mins
4. 2 hours
Expected Run Time
I have the file.
Expected runtime (single file from database)

Assume the file location is distributed with the PMF:
- \( P(\text{file on computer}) = 0.10 \)
- \( P(\text{file in SoCal}) = 0.50 \)
- \( P(\text{file in Japan}) = 0.37 \)
- \( P(\text{file in Space}) = 0.03 \)

What is the expected runtime of \( \text{database.get_movie(movie_name)} \)?

\[
E[\text{get_movie_database_time}] = E[\text{get_movie_database_time}|\text{Home}] \cdot P(\text{Home}) \\
+ E[\text{get_movie_database_time}|\text{SoCal}] \cdot P(\text{SoCal}) \\
+ E[\text{get_movie_database_time}|\text{Japan}] \cdot P(\text{Japan}) \\
+ E[\text{get_movie_database_time}|\text{Space}] \cdot P(\text{Space}) \\
= 0.1 \cdot 0.3s + 0.5 \cdot 1.6 \text{min} + 0.37 \cdot 5 \text{min} + 0.03 \cdot 2 \text{hours} \\
\approx 6.25 \text{mins}
\]
We can store a local copy of movies:

```python
if movie_name in movie_cache:
    return movie_cache[movie_name]
else:
    return database.get_movie(movie_name)
```

- Assume movie is in the cache is a Bernoulli with parameter 0.8
- Getting movie from the cache takes 0.3s time (same as having a local copy).

What is the expected runtime?

\[
E[\text{get\_movie\_time}] = P(\text{Movie not in Cache}) \cdot E[\text{get\_movie\_time}|\text{Movie not in Cache}]
+ P(\text{Movie in Cache}) \cdot E[\text{movie\_from\_cache}|\text{Movie in Cache}]
\]

\[
= P(\text{Movie not in Cache}) \cdot E[\text{get\_movie\_database}]
+ P(\text{Movie in Cache}) \cdot E[\text{movie\_from\_cache}|\text{Movie in Cache}]
\]

\[
= 0.2 \cdot 6.25\text{mins} + 0.8 \cdot 0.3s
\approx 1.2\text{mins}
\]
Expected runtime (storing many files)

We can preload many movies:

```python
for movie_name in movie_name_list:
    if movie_name not in old_movie_cache:
        movie_cache[movie_name] = database.get_movie(movie_name)
    else:
        movie_cache[movie_name] = old_movie_cache[movie_name]
```

- Assume we are interested in \( n \) movies.
- Assume retrieval time of a movie from the old cache takes 0.3s time.

What is the expected runtime? What is the approximate distribution of the runtime?

\[
E[get_{n\_movies}] = \sum_{i=1}^{n} E[get_{\_movie}]
= n \cdot E[get_{\_movie}]
\approx n \cdot 1.2mins
\]

\( get_{n\_movies} \sim \mathcal{N}(n \cdot 1.2, n \cdot 19.8) \)
Netflix Video Caching?

```python
class MovieRetriever(object):
    def __init__(self, database, movie_cache):
        self.database = database # Database of movies.
        self.cache = movie_cache # Fixed size dict.

    def get_movie(self, movie_name):
        if movie_name in self.movie_cache:
            return self.movie_cache[movie_name]
        else:
            if self.movie_cache.is_full():
                self.movie_cache.evict() # Remove a movie uniformly
                movie_data = self.database.get_movie(movie_name)
                self.movie_cache[movie_name] = movie_data
                cache_size += 1
            return movie_data

    def batch_load(self, movie_name_list):
        for movie_name in movie_name_list:
            if movie_name not in self.movie_cache:
                self.movie_cache[movie_name] = self.database.get_movie(movie_name)
```

- Can distribute a copy per geo location.
- Users can get a movie in 1.2 mins on average.
- Users can load popular shows while watching current videos.

These help to form metrics for the video quality team to better stream videos.
Bye Friend!
Theory Problems
Analyzing Recursive Code

```cpp
int Recurse() {
    int x = randint(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 \( \text{Recurse}() \). What is \( E[Y] \)?

\[
\]

\[
E[Y | X = 1] = 3
\]

\[
\]

\[
E[Y | X = 3] = E[7 + Y] = 7 + E[Y]
\]

\[
E[Y] = 3(1/3) + (5 + E[Y])(1/3) + (7 + E[Y])(1/3) = (1/3)(15 + 2E[Y])
\]

\[
E[Y] = 15
\]
Protip: do this in CS161
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 $X_1 \ldots X_{100}$ where $X_i \sim \text{Bern}(p)$

```python
# 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 $X_1 \ldots X_{100}$ where $X_i \sim \text{Bern}(p)$

```python
# Maximize accuracy, while preserving privacy.
def calculateYi(Xi):
    obfuscate = random()
    if obfuscate:
        return indicator(random())
    else:
        return Xi
```

What is $E[Y_i]$?

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

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

```python
# Maximize accuracy, while preserving privacy.
def calculateYi(Xi):
    obfuscate = random()
    if obfuscate:
        return indicator(random())
    else:
        return Xi
```

Let $Z = \sum_{i=1}^{100} Y_i$, What is the $E[Z]$?

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

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

```python
# Maximize accuracy, while preserving privacy.
def calculateYi(Xi):
    obfuscate = random()
    if obfuscate:
        return indicator(random())
    else:
        return Xi
```

Let $Z = \sum_{i=1}^{100} Y_i$ \quad $E[Z] = 50p + 25$  \quad 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

Now at Stanford
Uncertainty Theory

- Beta Distributions
- Thompson Sampling
- Adding Random Vars
- Central Limit Theorem
- Sampling
- Bootstrapping
- Algorithmic Analysis
Where are we in CS109?

On Monday...

- Counting Theory
- Core Probability
- $X^2$ Random Variables
- Probabilistic Models
- Uncertainty Theory
- Machine Learning