CIS192 Python Programming
Probability, Simulations & Course Wrap-Up

Raymond Yin
University of Pennsylvania

December 8, 2016
Outline

1 Probability and Simulations
   • The Two Child Problem
   • Basic Definitions
   • The Law of Large Numbers
   • Monty Hall in Python

2 Looking Forward
   • CIS 192 in Review
   • Next Steps
The Two Child Problem

Suppose I have two children.

I announce that I have a son.

What is the probability that both children are boys?
<table>
<thead>
<tr>
<th>First child</th>
<th>Second child</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boy</td>
<td>Boy</td>
</tr>
<tr>
<td>Boy</td>
<td>Girl</td>
</tr>
<tr>
<td>Girl</td>
<td>Boy</td>
</tr>
<tr>
<td>Girl</td>
<td>Girl</td>
</tr>
</tbody>
</table>
I announce that

“I have a son who was born on a Tuesday.”

Now what is the probability that both children are boys?
The *Tuesday Child Problem*, visually demonstrated.
Outline

1 Probability and Simulations
   • The Two Child Problem
   • Basic Definitions
   • The Law of Large Numbers
   • Monty Hall in Python

2 Looking Forward
   • CIS 192 in Review
   • Next Steps
Definition: A *Random Variable* is a function from a set of possible outcomes to numeric values.

- For a coin toss we’ve been mapping [lands heads] to 1 and [lands tails] to 0.
- The values associated with a die roll are the numbers 1 through 6.

We will model random variables with "random" Python functions.
Each random variable has an associated *Probability Distribution* that maps its values to *probabilities*.

- Every probability must be in the interval \([0, 1]\).
- The sum of the probabilities must equal 1.

We say \(Pr(X = x) = p\) to denote the probability of an event.

- For example, if \(D\) represents the outcome of a fair die, \(Pr(D = 3) = \frac{1}{6}\).
Expectation

\[ E(X) = \sum_x x \times Pr(x) \]

- Where \( C \) represents a coin toss as above, 
  \[ E(C) = 0.5 \times 1 + 0.5 \times 0 = 0.5 \]
- For a die \( D \), 
  \[ E(D) = \frac{1}{6} \times 1 + \cdots + \frac{1}{6} \times 6 = 3.5 \]
Outline

1 Probability and Simulations
   - The Two Child Problem
   - Basic Definitions
   - The Law of Large Numbers
   - Monty Hall in Python

2 Looking Forward
   - CIS 192 in Review
   - Next Steps
The Law of Large Numbers

The *Weak Law of Large Numbers* states that as the number of trials approaches infinity, the frequency of a given outcome approaches its true probability.

- i.e. If we toss a lot of coins, we will get heads close to half the time.
Simulations

Simulating a situation’s possibilities is easy in Python!

Use the `random` library

- `random()` returns a float between 0 and 1 uniformly at random
- `randint(a, b)` returns a random integer between `a` and `b`, inclusive
- `choice(li)` samples a random element from `li`
- `sample(li)` returns a randomly selected subset of `li`
- `shuffle(li)` randomly permutes the list `li`
- `uniform(a, b)` returns a float uniformly distributed between `a` and `b`
1 Probability and Simulations
   - The Two Child Problem
   - Basic Definitions
   - The Law of Large Numbers
   - Monty Hall in Python

2 Looking Forward
   - CIS 192 in Review
   - Next Steps
You’re on a gameshow with Monty Hall!

There are three doors in front of you, two have goats behind them and one has a car. After you pick a door, Monty opens another door, revealing a goat. He then gives you the option of switching.

Should you switch?
def monty(switch = False):
    car = randint(1, 3)
    pick = randint(1, 3)
    opened = choice(list({1, 2, 3} - {car, pick}))
    if switch:
        pick = choice(list({1, 2, 3} - {pick, opened}))
    if pick == car:
        return "Car"
    else:
        return "Goat"
Outline

1. Probability and Simulations
   - The Two Child Problem
   - Basic Definitions
   - The Law of Large Numbers
   - Monty Hall in Python

2. Looking Forward
   - CIS 192 in Review
   - Next Steps
Topics We’ve Covered

- Python Basics & Fundamentals
- Functional Programming
- Object-Oriented Programming
- Iterators, Generators, Exceptions & IO
- Regular Expressions & Other Modules
- HTTP Requests / HTML Parsing
- Data Analysis
- Machine Learning
- Natural Language Processing
- Web Apps
- Artificial Intelligence
- Probability & Simulations
1. Probability and Simulations
   - The Two Child Problem
   - Basic Definitions
   - The Law of Large Numbers
   - Monty Hall in Python

2. Looking Forward
   - CIS 192 in Review
   - Next Steps
Scratching the Surface

- Each special topic has MUCH more depth than what we’ve covered this semester.
- Many topics we haven’t mentioned
  - Parallel & Distributed Computing
  - Concurrency
  - Graphical User Interfaces
  - Testing Frameworks
  - etc.
I hope you find the skills you’ve acquired from CIS 192 useful!
Build your own side-projects, big or small!
Learn more about Python!
  - PyCon conference recordings
  - Obey the Testing Goat (TDD in Python + Web Dev)
  - New Coder: practical tutorials in Python

Join the Python community!
  - /r/Python subreddit
  - Trending GitHub Python repositories
  - StackOverflow Python questions