

CMPU 100 · Programming with Data

# Simulation and Probability

Class 19





arte HD



[youtu.be/2NhFoSFNQQMQ](https://youtu.be/2NhFoSFNQQMQ)

arte HD



[youtu.be/2NhFoSFNQQMQ](https://youtu.be/2NhFoSFNQQMQ)

Notebook: *Warm-up: Generating sonnets*

Simple simulation: Rolling dice

Probability is a very mathematical topic, but it's one with a lot of importance to computer science and data science.

We'll just see some of the basics!

Notebook: *Rolling a die*

Suppose a friend proposes the following two scenarios for a game:

(a) *Roll two dice and sum their values*



(b) *Roll one die and multiply its value by two*



Which scenario will give you a higher score on average?

Notebook: *Roll once or roll twice?*

# Simulation using loops



*Roman coin featuring  
Pompey the Great and a  
ship*

If you flip a fair coin 100 times, how often can you expect to get between 40 and 60 heads?



*Roman coin featuring  
Pompey the Great and a  
ship*

If you flip a fair coin 100 times, *what is the chance* of getting between 40 and 60 heads?



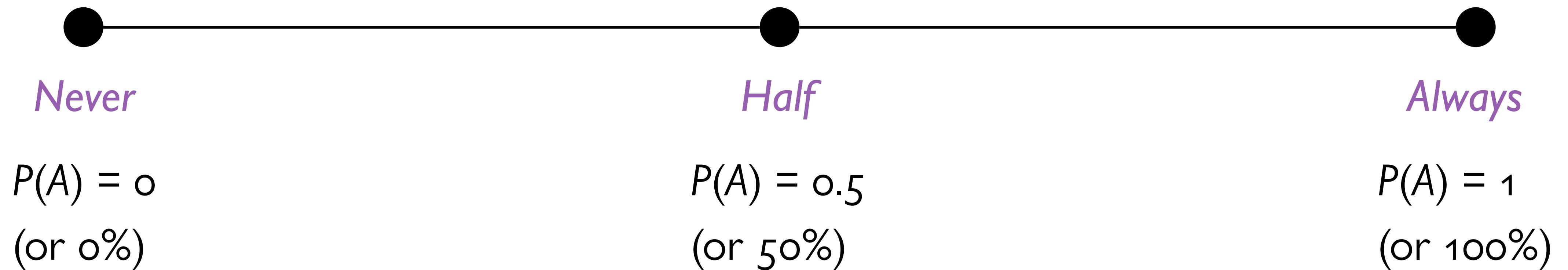
Roman coin featuring  
Pompey the Great and a  
ship

If you flip a fair coin 100 times, *what is the chance* of getting between 40 and 60 heads?

***The intuition of probability:*** If we repeat an event (flipping a coin) many times (say, 1 million) ... what fraction of the times would we see the outcome we care about (heads)?

# Probability

$P(A)$ : Chance that an event  $A$  will occur



$$P(\text{heads}) = 1/2$$

# How to calculate an event's probability?

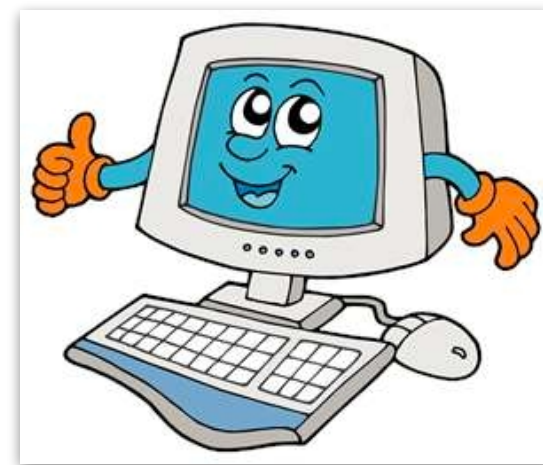
## Computers (simulation)

```
N = 1000000 #Roll the dice 1 million times
option_a = np.random.choice(dice, N) + np.random.c
option_b = 2 * np.random.choice(dice, N)

print("Option A Mean: ", np.mean(option_a))
print("Option B Mean: ", np.mean(option_b))

Option A Mean: 7.003198
Option B Mean: 6.99884

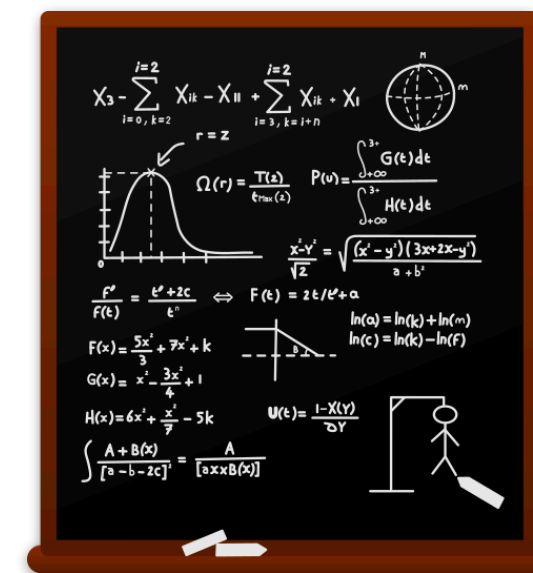
samples = Table().with_columns("Option A", option_
samples.hist("Option A", bins=np.arange(0,14))
samples.hist("Option B", bins=np.arange(0,14))
```



Rooted in *algorithms*

- ✗ Approximate solutions
- ✓ Often convincing
- ✓ Non-trivial problems potentially captured cleanly with code

## Math (analytical)



Rooted in *rules (axioms)*

- ✓ Exact solutions
- ✓ Straightforward for simple problems
- ✗ Non-trivial problems potentially difficult to analyze/verify

# Estimating probability with simulation algorithm

If you flip a fair coin 100 times, *what is the chance* of getting 40–60 heads?



Repeat many times:  
    Simulate one trial  
    Record the outcome  
Analyze outcomes for all trials

# Estimating probability with simulation algorithm

If you flip a fair coin 100 times, *what is the chance* of getting 40–60 heads?



Repeat many times:  
    Simulate one trial  
    Record the outcome  
Analyze outcomes for all trials

*What is one trial?*

# Estimating probability with simulation algorithm

If you flip a fair coin 100 times, *what is the chance* of getting 40–60 heads?



Repeat many times:

    Simulate one trial

    Record the outcome

Analyze outcomes for all trials

*What is one trial?*

*What is the outcome of a trial?*

# Estimating probability with simulation algorithm

If you flip a fair coin 100 times, *what is the chance* of getting 40–60 heads?



Repeat many times:

    Simulate one trial

    Record the outcome

Analyze outcomes for all trials

*What is one trial?*

*What is the outcome of a trial?*

*How to answer the question using the outcomes?*

Notebook: *Simulation using loops*



