What are the chances?

INTRODUCTION TO STATISTICS IN PYTHON

Maggie Matsui
Senior Content Developer, DataCamp

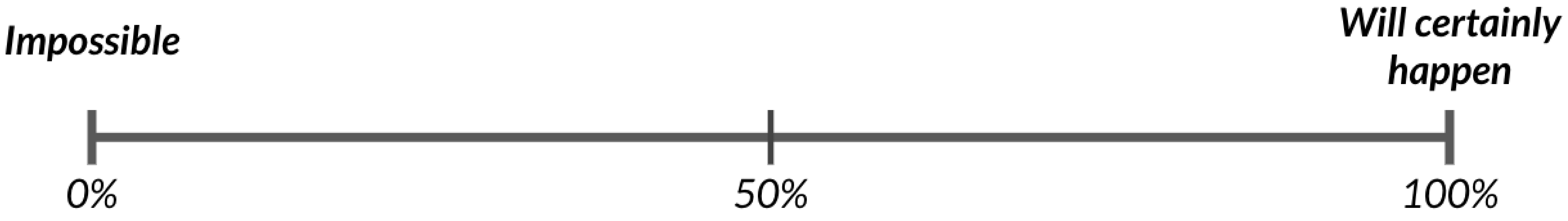
Measuring chance

What's the probability of an event?

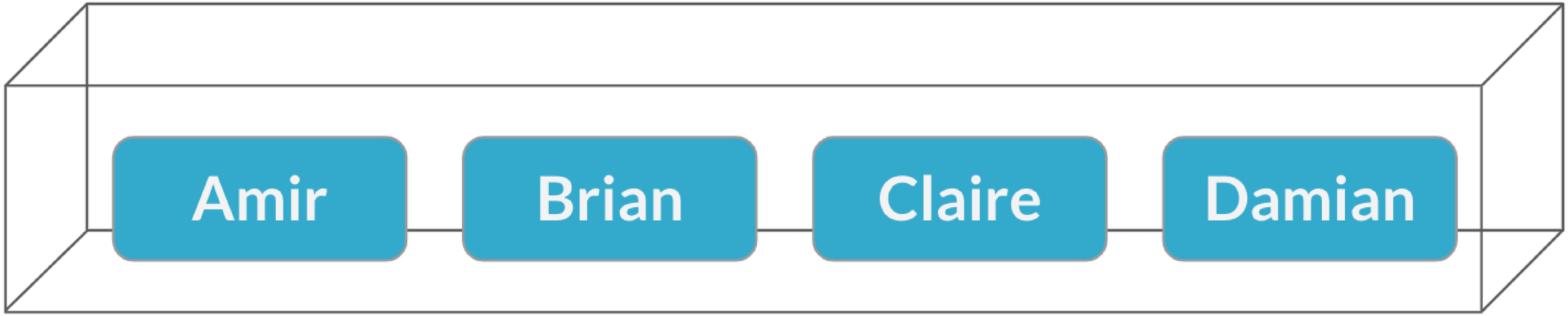
$$P(\text{event}) = \frac{\#\text{ ways event can happen}}{\text{total }\#\text{ of possible outcomes}}$$

Example: a coin flip

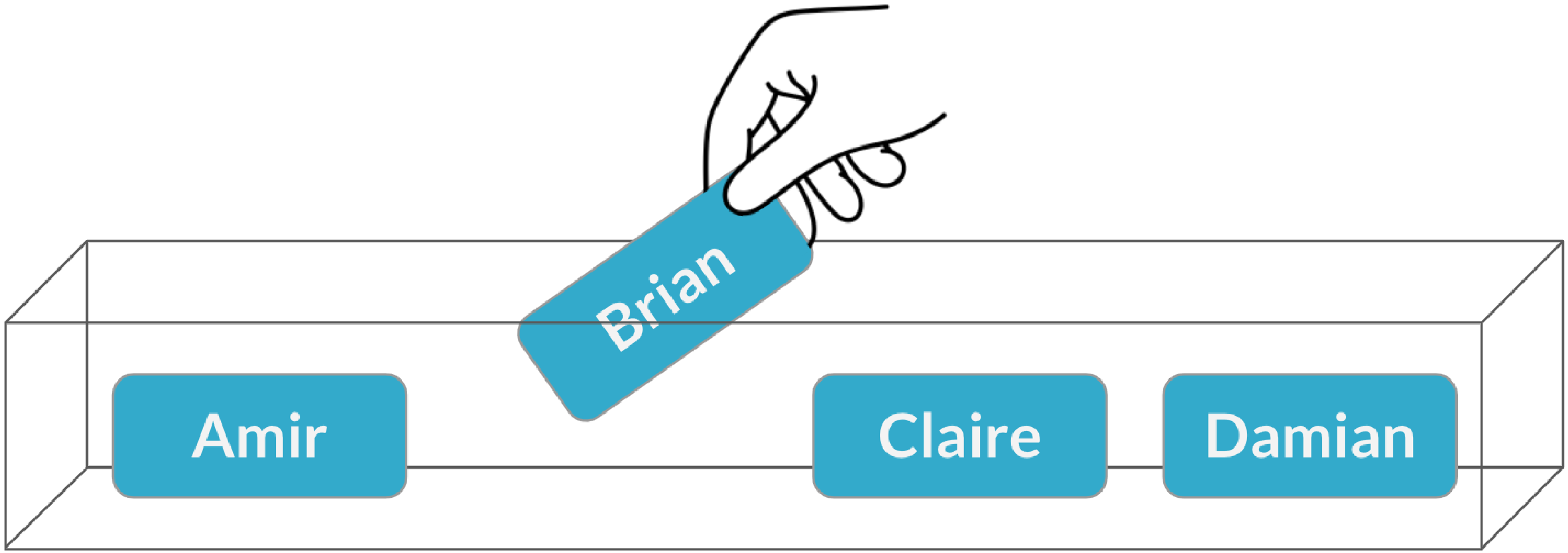
$$P(\text{heads}) = \frac{1\text{ way to get heads}}{2\text{ possible outcomes}} = \frac{1}{2} = 50\%$$



Assigning salespeople



Assigning salespeople



$$P(\text{Brian}) = \frac{1}{4} = 25\%$$

Sampling from a DataFrame

```
print(sales_counts)
```

```
     name  n_sales
0    Amir      178
1   Brian      128
2  Claire       75
3  Damian       69
```

```
sales_counts.sample()
```

```
    name  n_sales
1  Brian      128
```

```
sales_counts.sample()
```

```
     name  n_sales
2  Claire       75
```

Setting a random seed

```
np.random.seed(10)

sales_counts.sample()
```

```
    name  n_sales
1  Brian      128
```

```
np.random.seed(10)
sales_counts.sample()
```

```
    name  n_sales
1  Brian      128
```

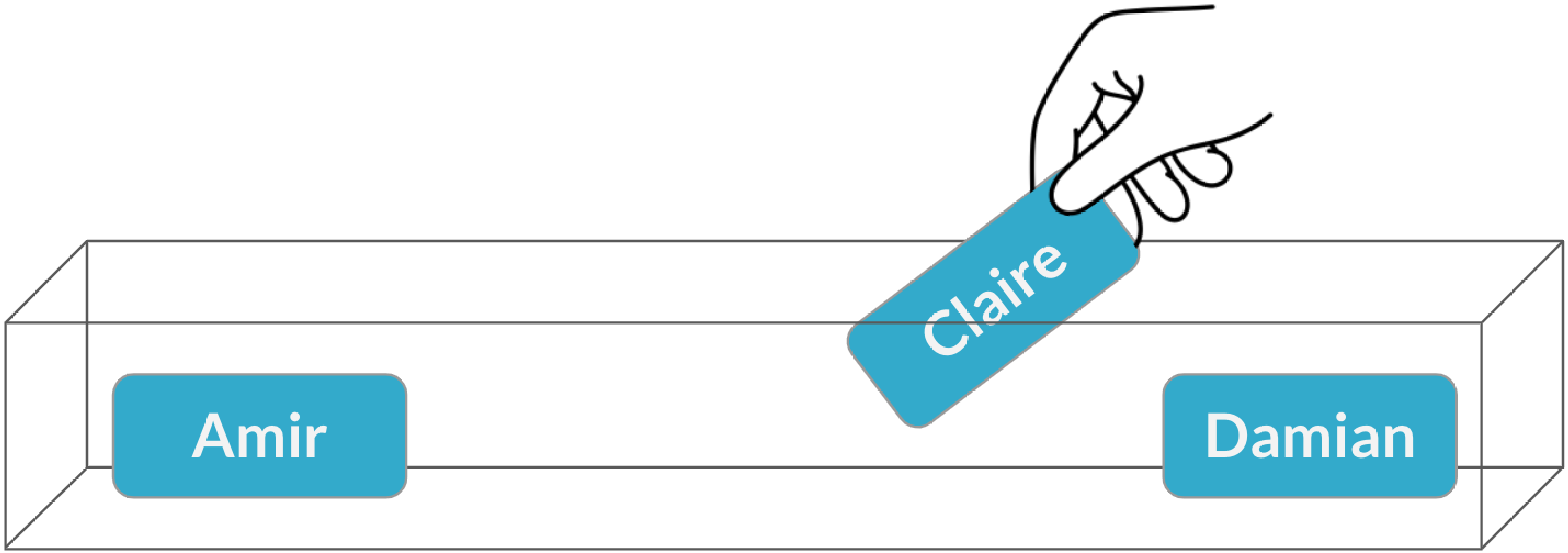
```
np.random.seed(10)
sales_counts.sample()
```

```
    name  n_sales
1  Brian      128
```

A second meeting

Sampling without replacement

A second meeting



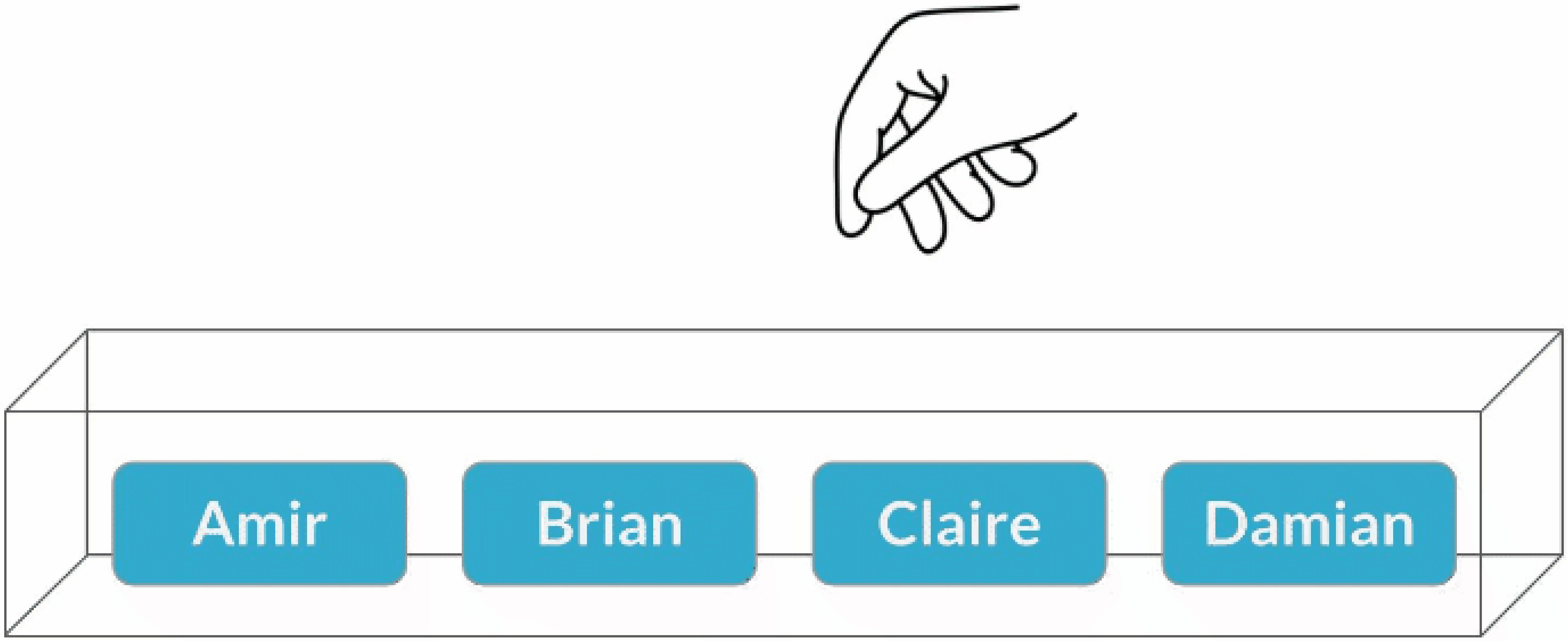
$$P(\text{Claire}) = \frac{1}{3} = 33\%$$

Sampling twice in Python

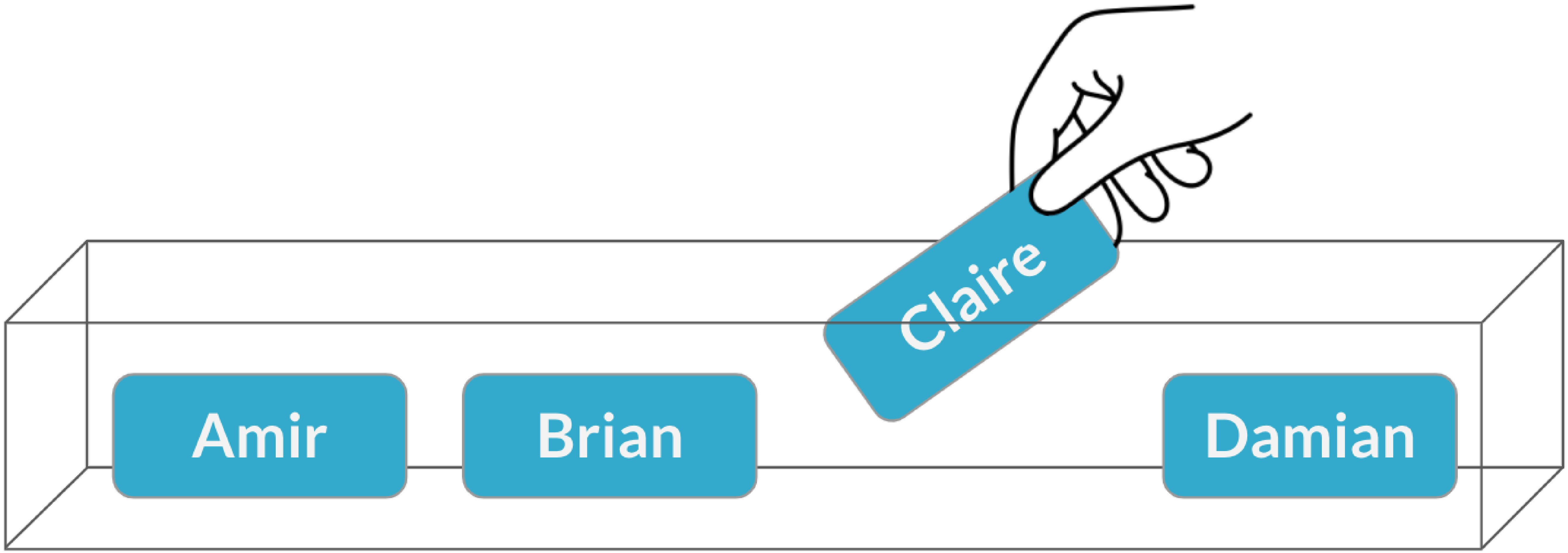
```
sales_counts.sample(2)
```

```
     name  n_sales
1   Brian      128
2  Claire       75
```

Sampling with replacement



Sampling with replacement



$$P(\text{Claire}) = \frac{1}{4} = 25\%$$

Sampling with/without replacement in Python

```
sales_counts.sample(5, replace = True)
```

```
     name  n_sales
1   Brian      128
2  Claire       75
1   Brian      128
3  Damian       69
0    Amir      178
```

Independent events

*Two events are **independent** if the probability of the second event **isn't** affected by the outcome of the first event.*

Sampling with Replacement

First pick

Second pick

Amir

Brian

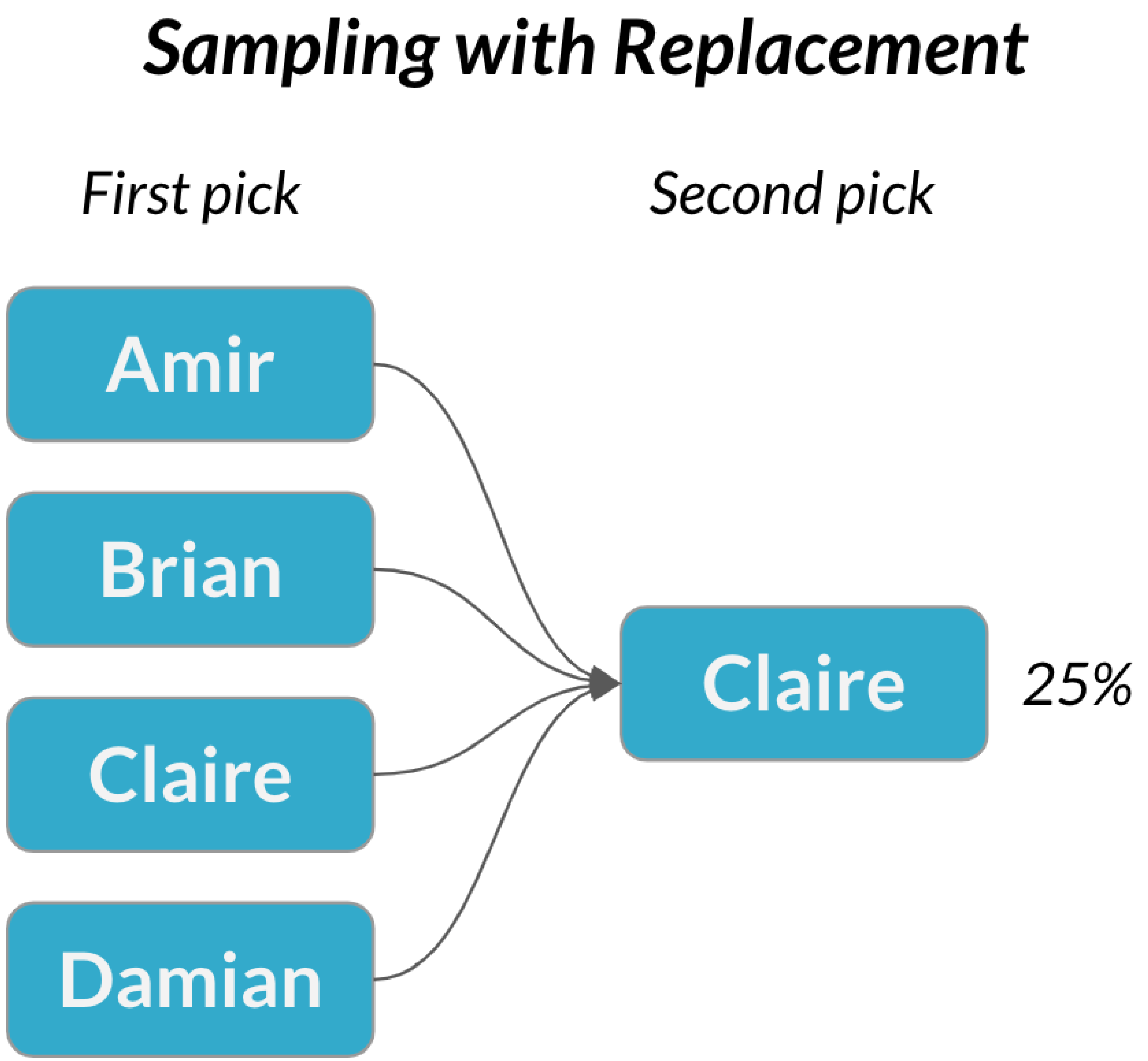
Claire

Damian

Independent events

*Two events are **independent** if the probability of the second event **isn't** affected by the outcome of the first event.*

Sampling with replacement = each pick is independent



Dependent events

*Two events are **dependent** if the probability of the second event **is** affected by the outcome of the first event.*

Sampling without Replacement

First pick | *Second pick*

Amir

Brian

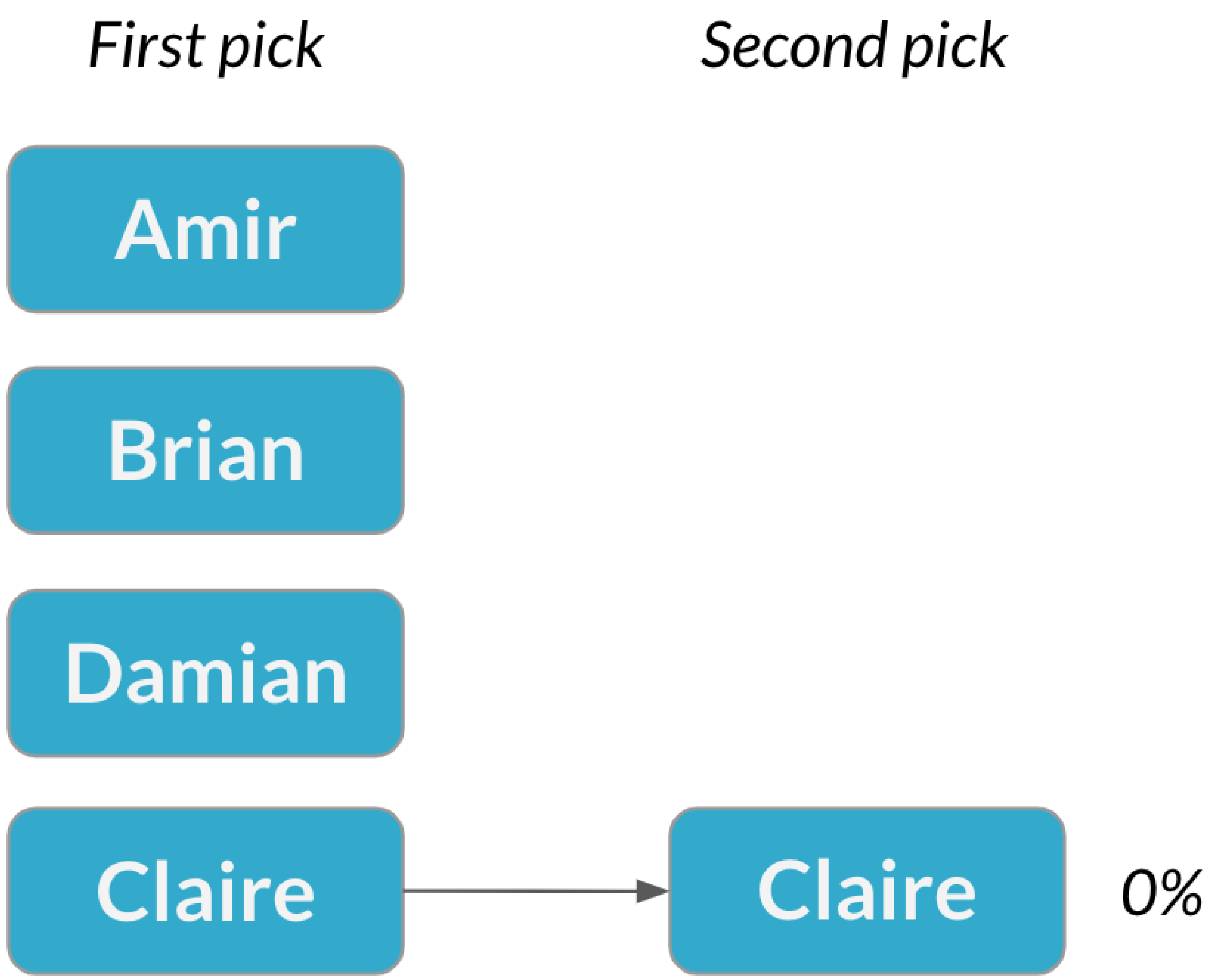
Damian

Claire

Dependent events

*Two events are **dependent** if the probability of the second event **is** affected by the outcome of the first event.*

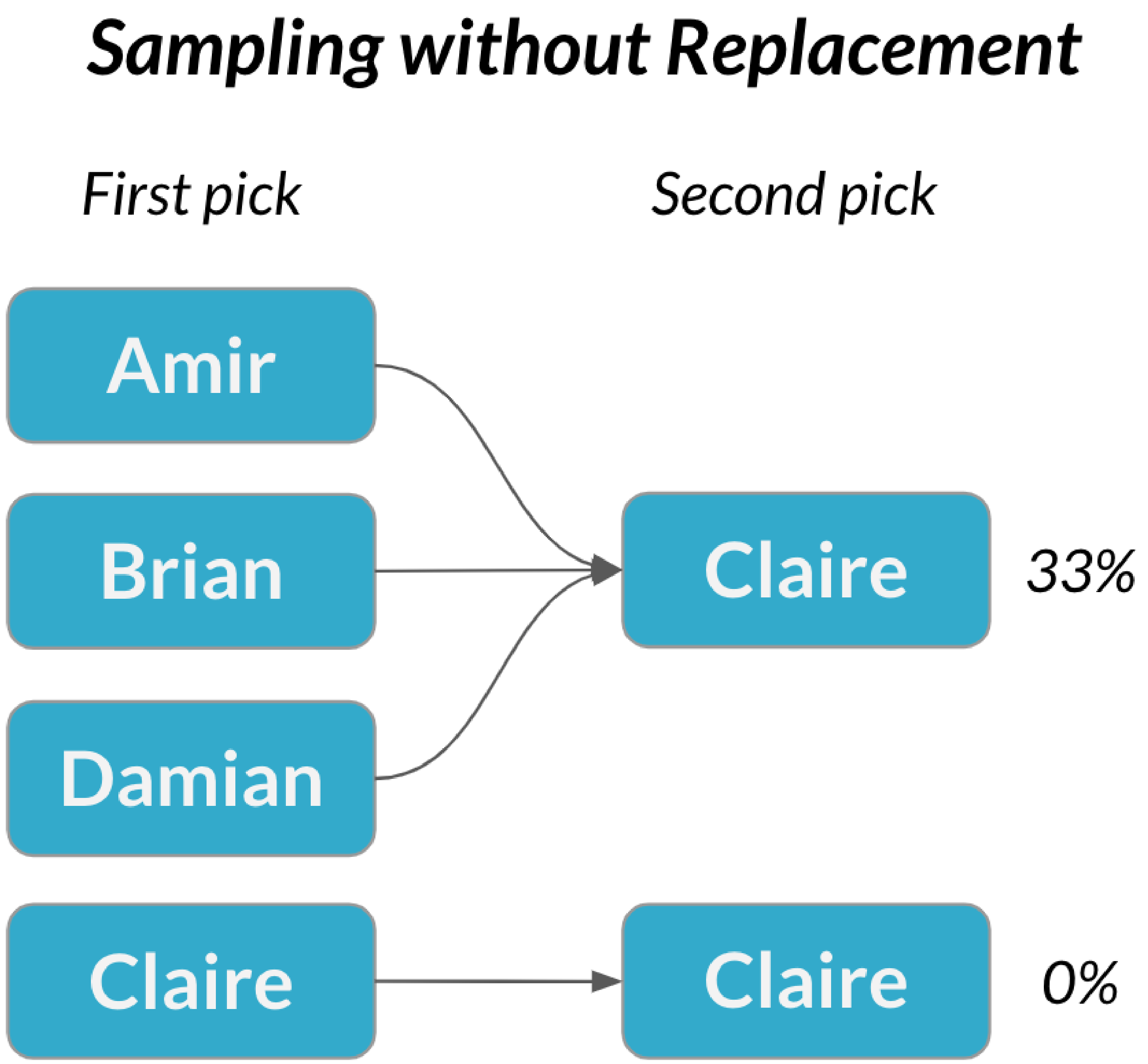
Sampling without Replacement



Dependent events

*Two events are **dependent** if the probability of the second event **is** affected by the outcome of the first event.*

Sampling without replacement = each pick is dependent



Let's practice!

INTRODUCTION TO STATISTICS IN PYTHON

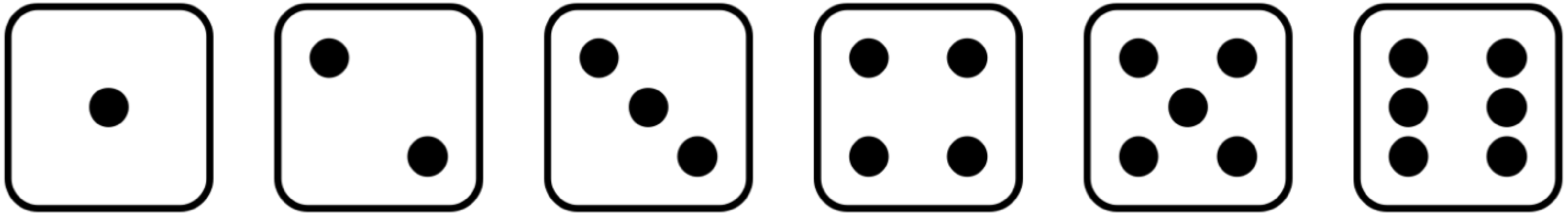
Discrete distributions

INTRODUCTION TO STATISTICS IN PYTHON

Maggie Matsui
Content Developer, DataCamp

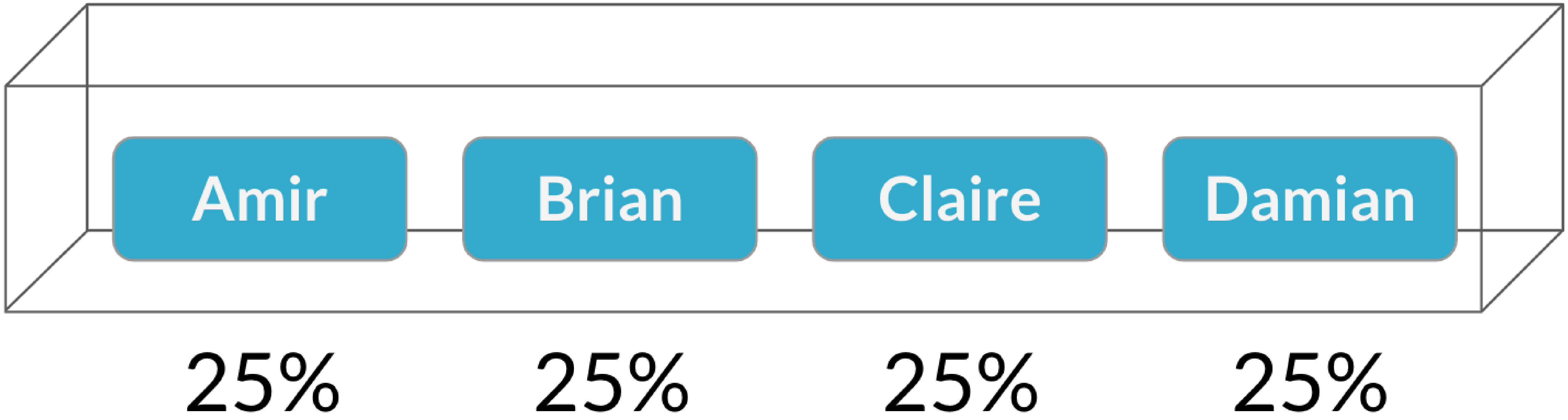
Rolling the dice

Rolling the dice



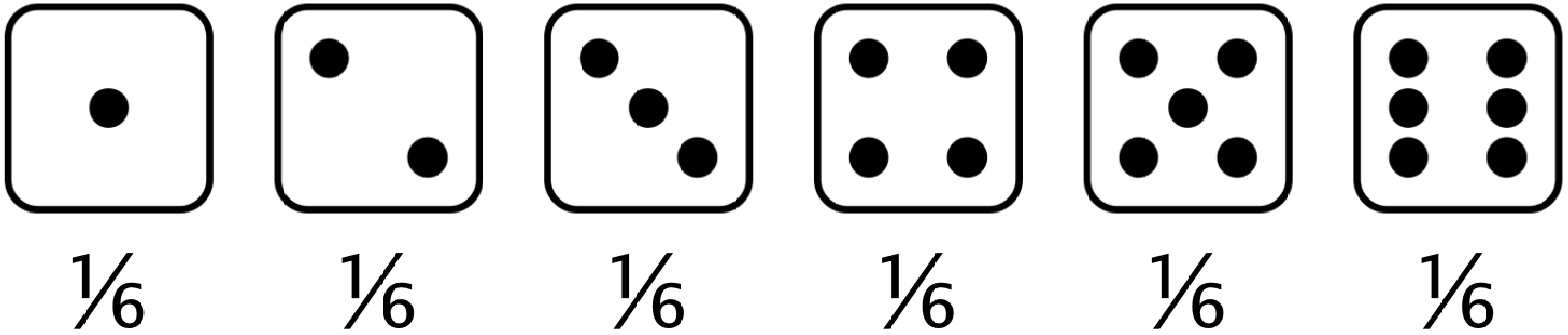
⅙ ⅙ ⅙ ⅙ ⅙ ⅙

Choosing salespeople



Probability distribution

Describes the probability of each possible outcome in a scenario

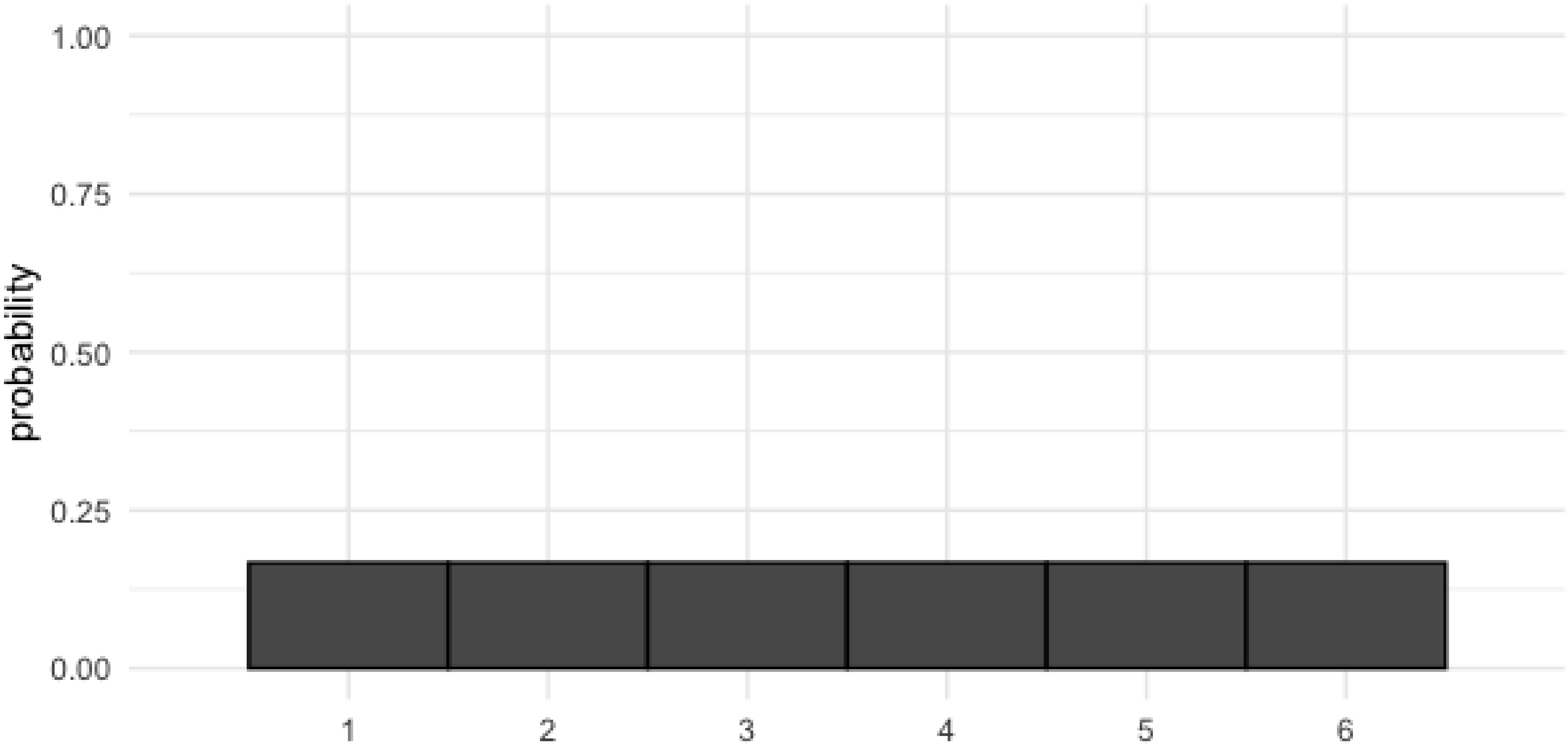


Expected value: mean of a probability distribution

Expected value of a fair die roll =

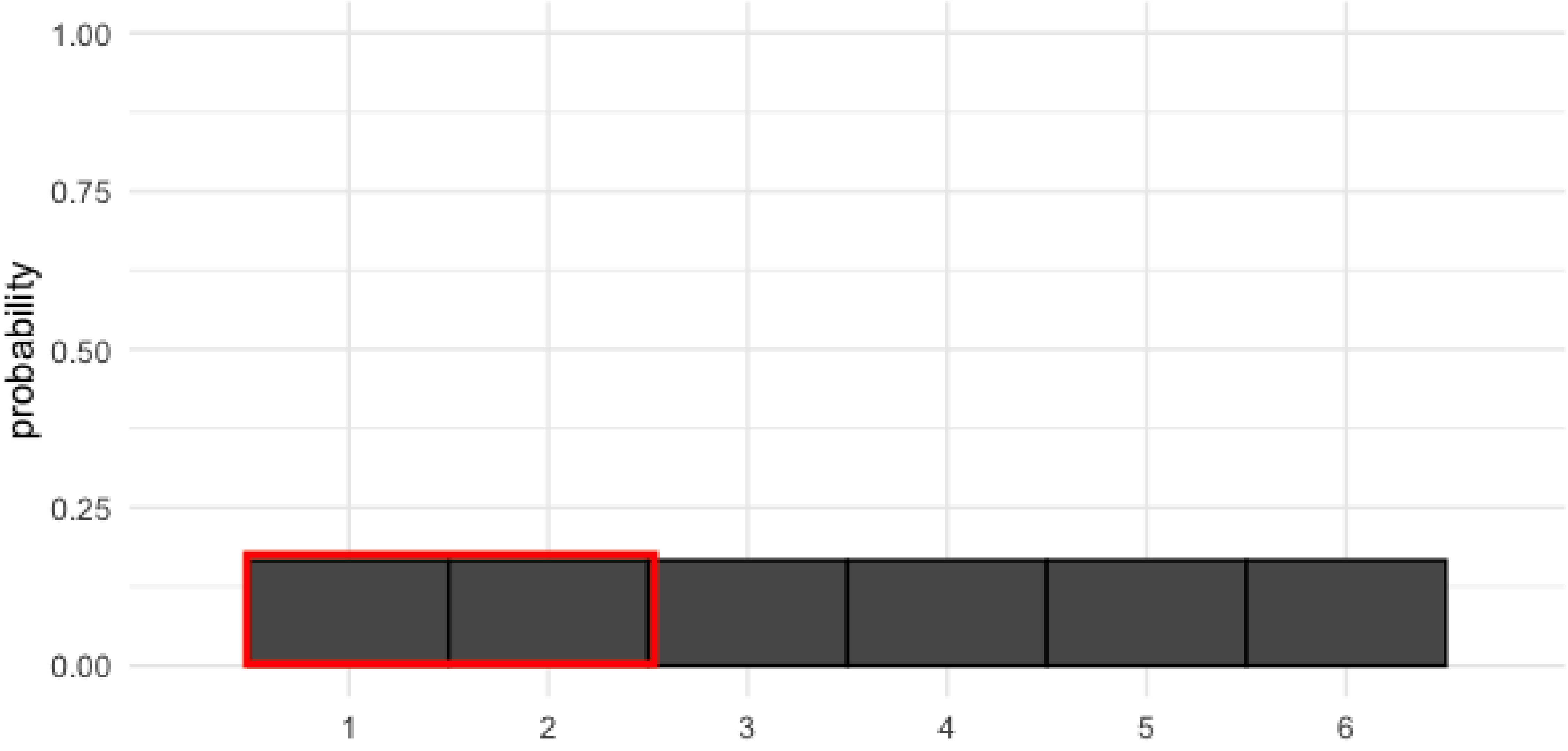
$$(1 \times \frac{1}{6}) + (2 \times \frac{1}{6}) + (3 \times \frac{1}{6}) + (4 \times \frac{1}{6}) + (5 \times \frac{1}{6}) + (6 \times \frac{1}{6}) = 3.5$$

Visualizing a probability distribution



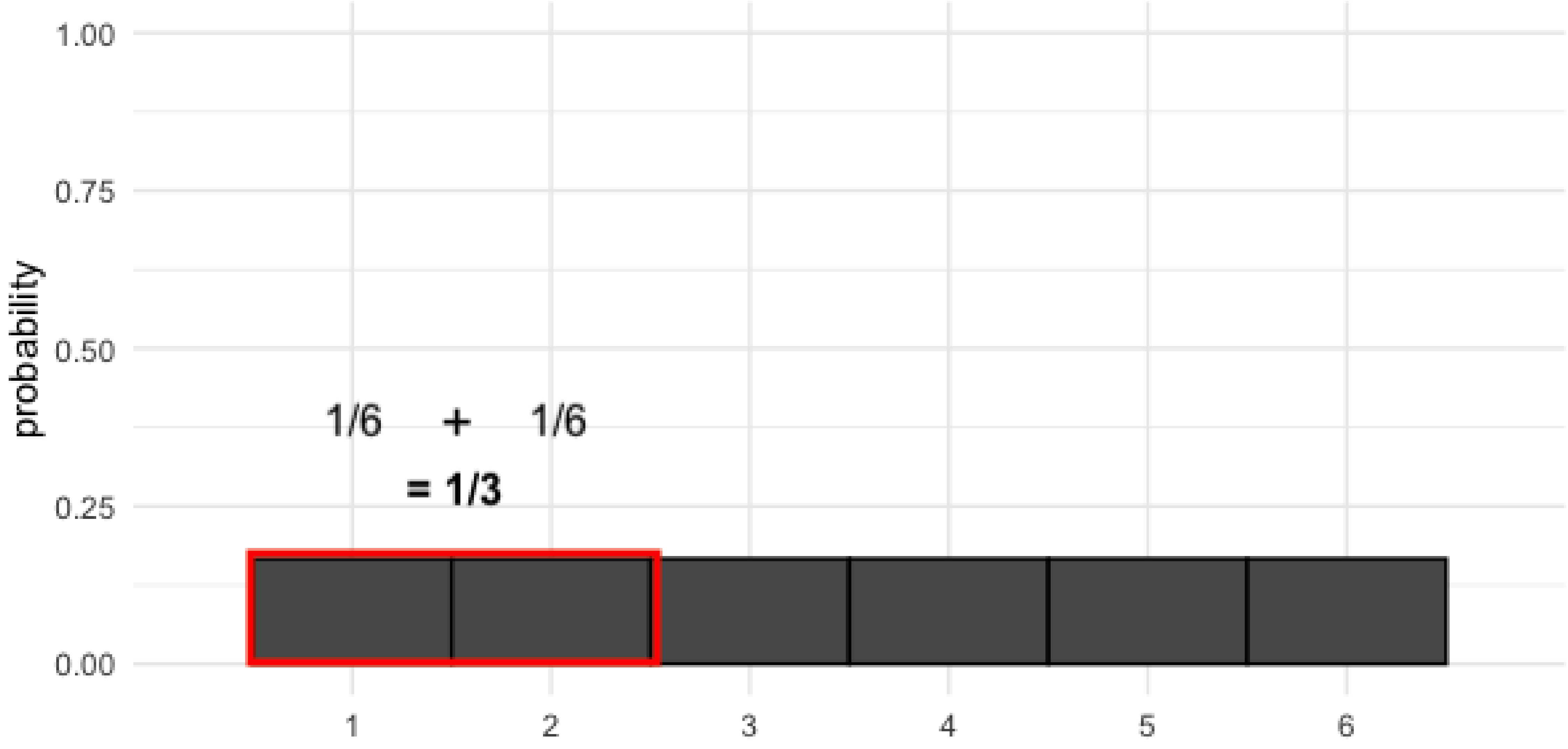
Probability = area

$$P(\text{die roll}) \leq 2 = ?$$

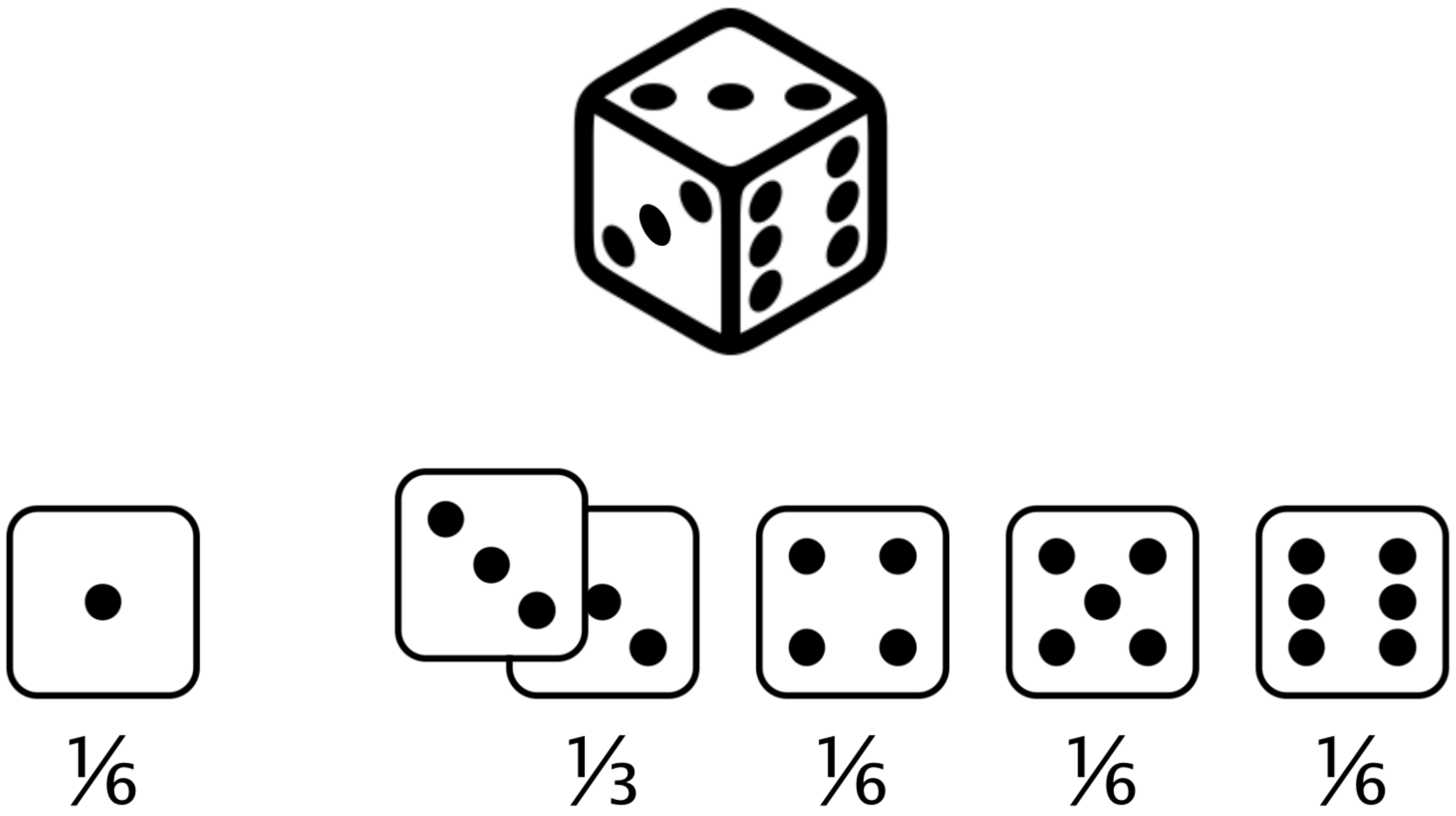


Probability = area

$$P(\text{die roll}) \leq 2 = 1/3$$



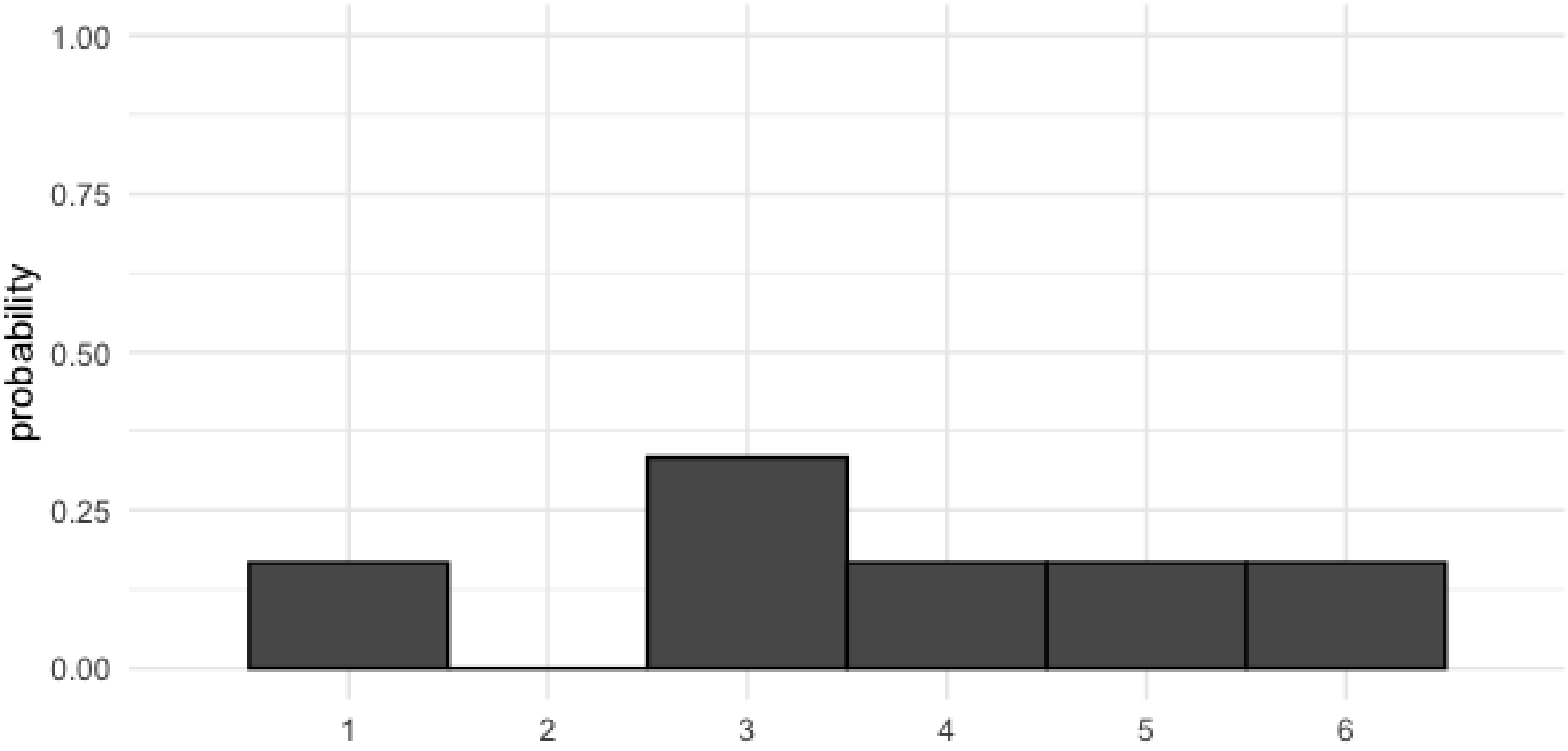
Uneven die



Expected value of uneven die roll =

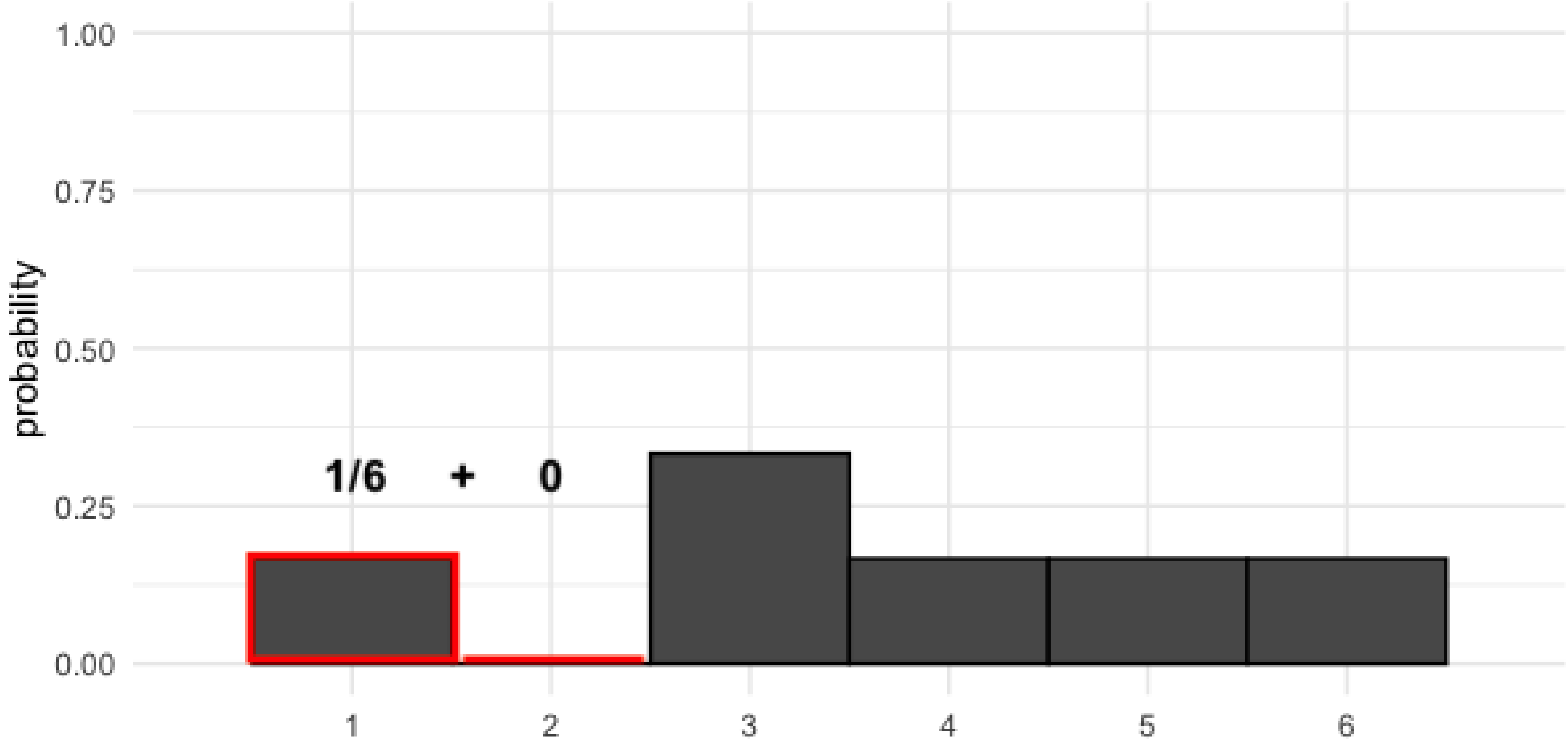
$$(1 \times \tfrac{1}{6}) + (2 \times 0) + (3 \times \tfrac{1}{3}) + (4 \times \tfrac{1}{6}) + (5 \times \tfrac{1}{6}) + (6 \times \tfrac{1}{6}) = 3.67$$

Visualizing uneven probabilities



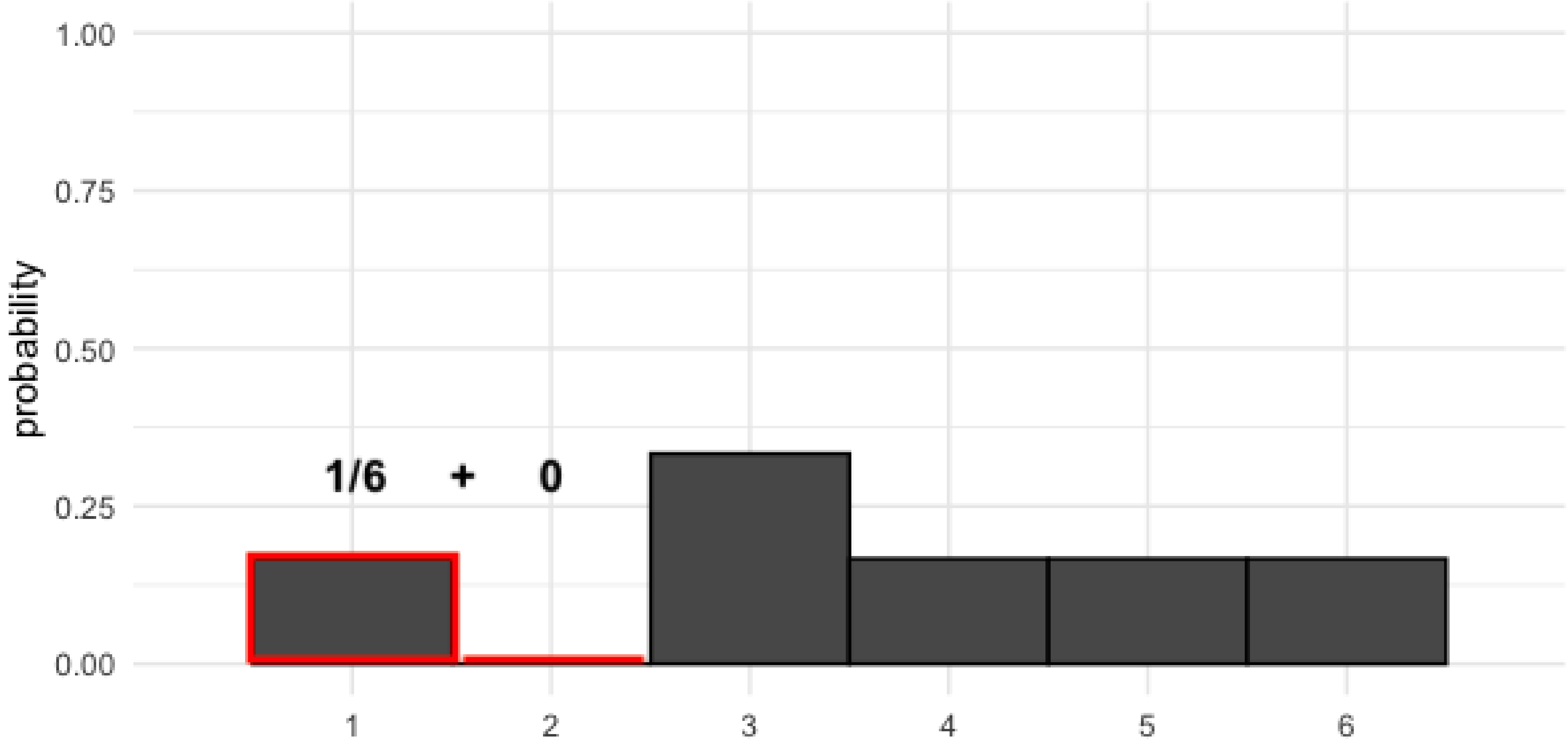
Adding areas

$$P(\text{uneven die roll}) \leq 2 = ?$$



Adding areas

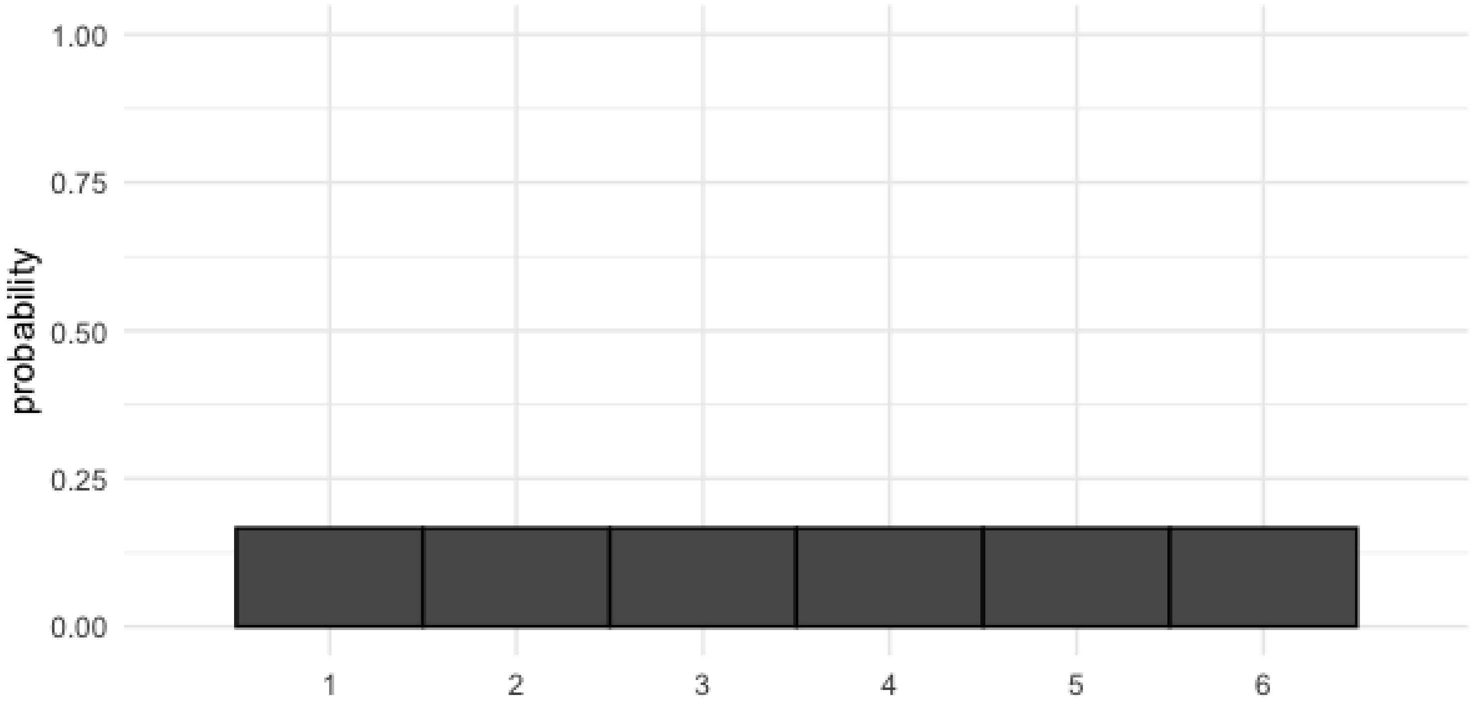
$$P(\text{uneven die roll}) \leq 2 = 1/6$$



Discrete probability distributions

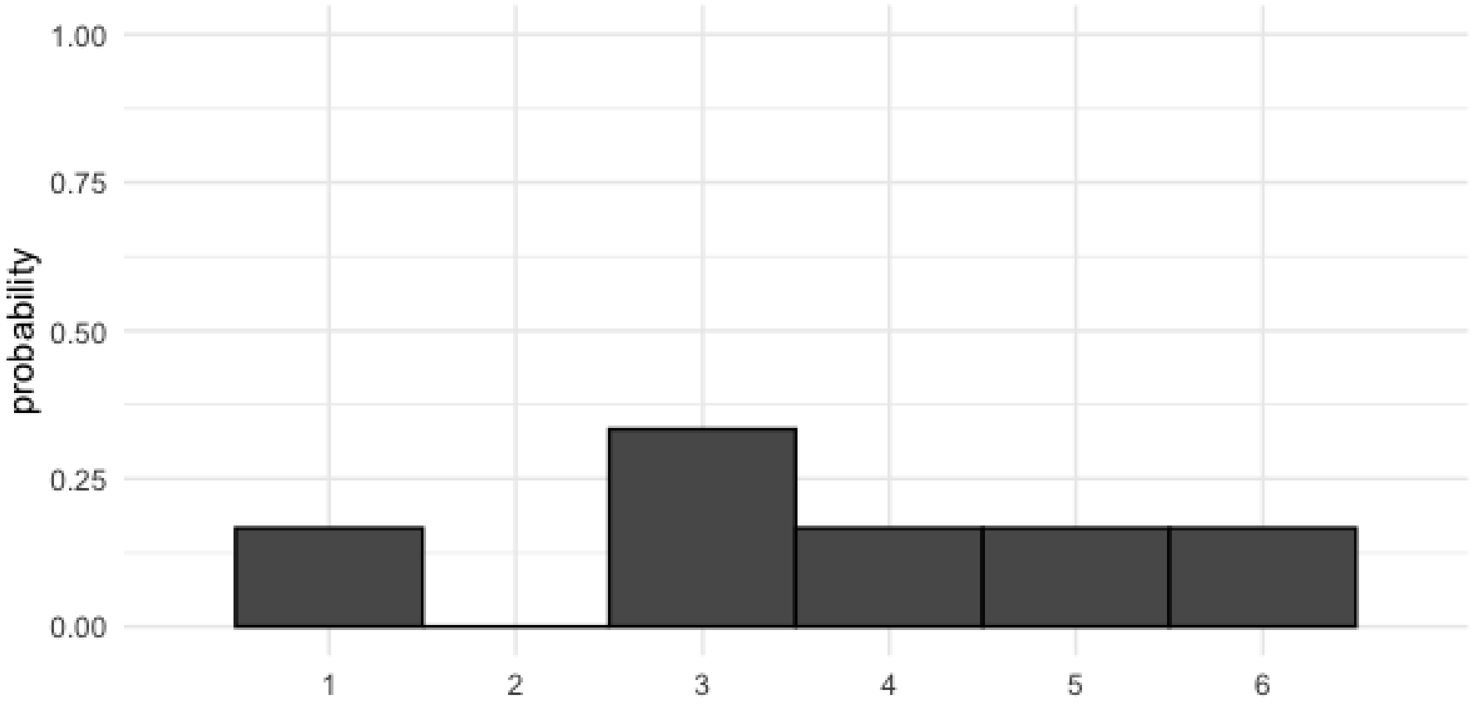
Describe probabilities for discrete outcomes

Fair die



Discrete uniform distribution

Uneven die



Sampling from discrete distributions

```
print(die)
```

```
   number      prob
0       1  0.166667
1       2  0.166667
2       3  0.166667
3       4  0.166667
4       5  0.166667
5       6  0.166667
```

```
np.mean(die['number'])
```

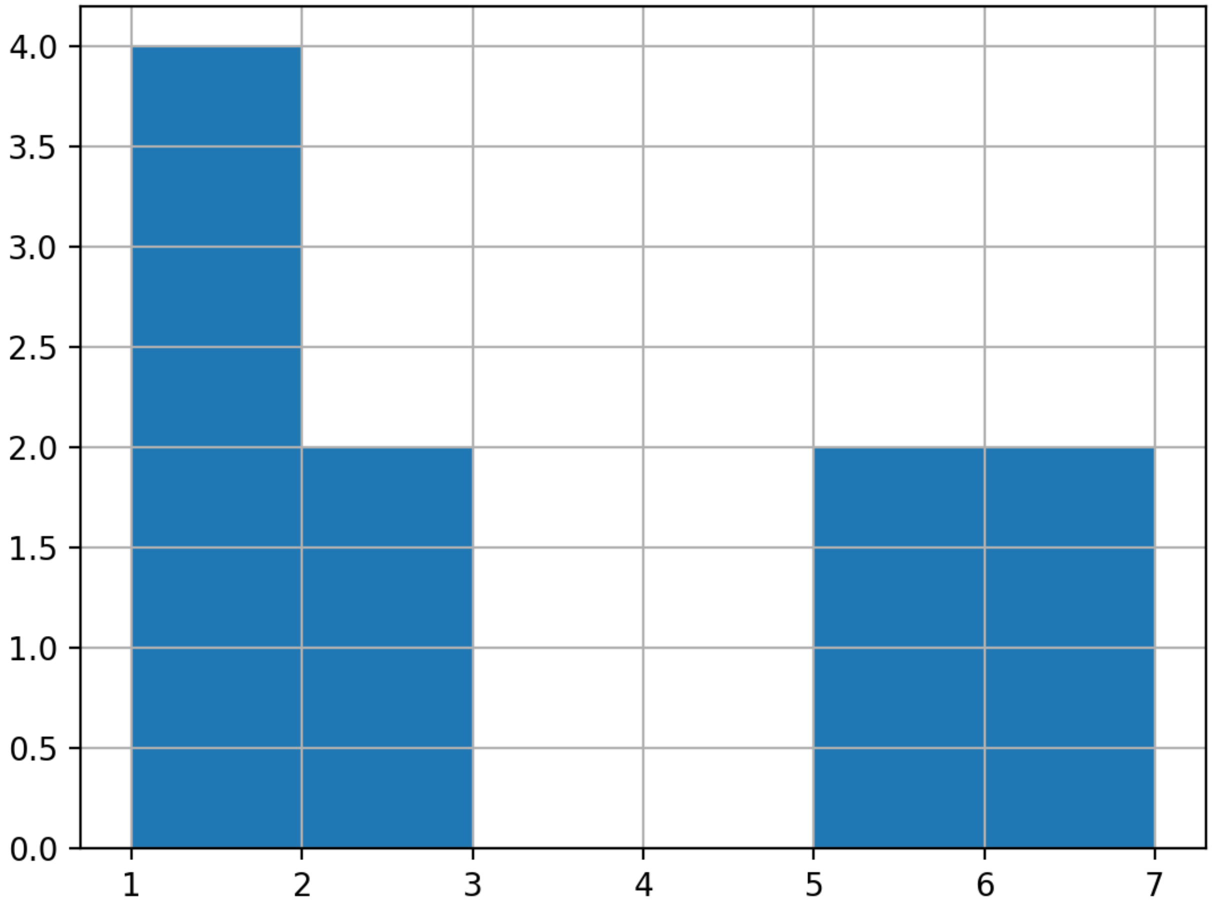
```
3.5
```

```
rolls_10 = die.sample(10, replace = True)
rolls_10
```

```
   number      prob
0       1  0.166667
0       1  0.166667
4       5  0.166667
1       2  0.166667
0       1  0.166667
0       1  0.166667
5       6  0.166667
5       6  0.166667
...
```

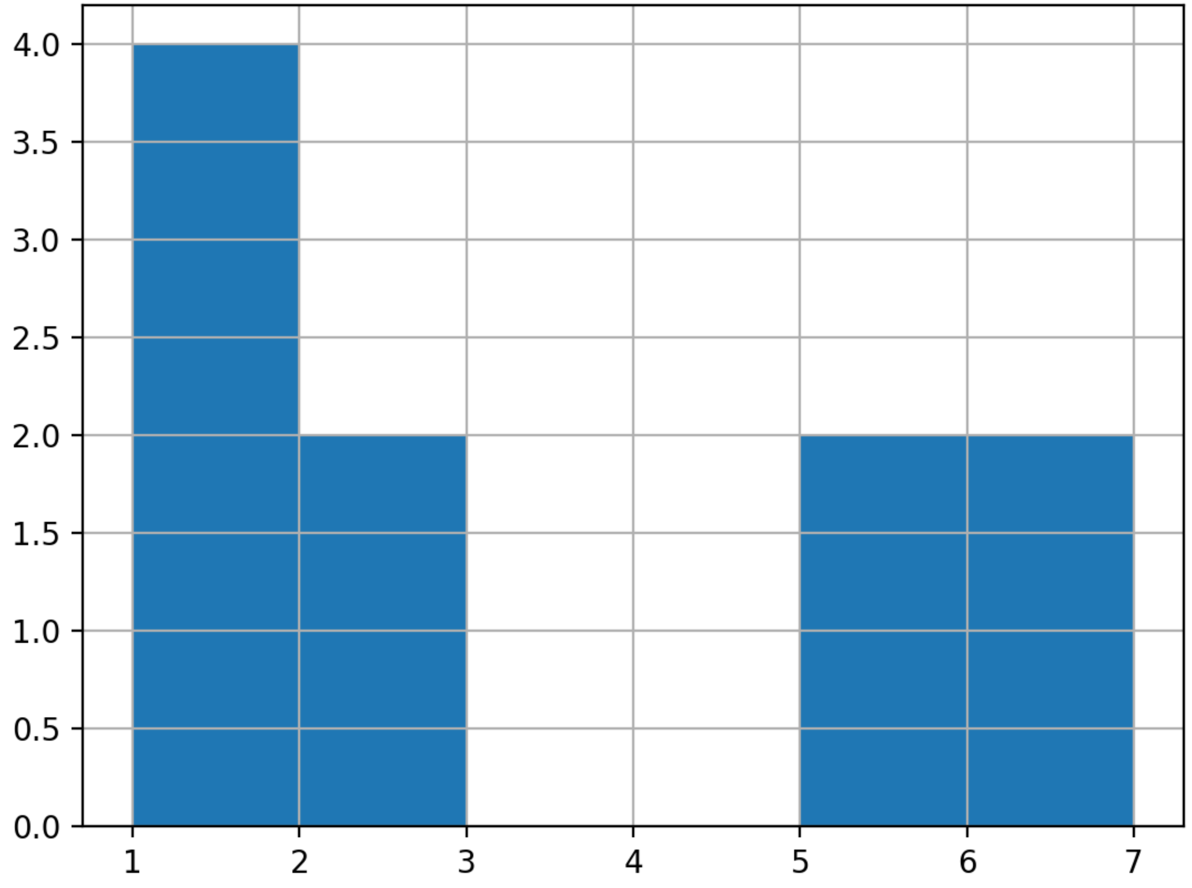
Visualizing a sample

```
rolls_10['number'].hist(bins=np.linspace(1,7,7))
plt.show()
```



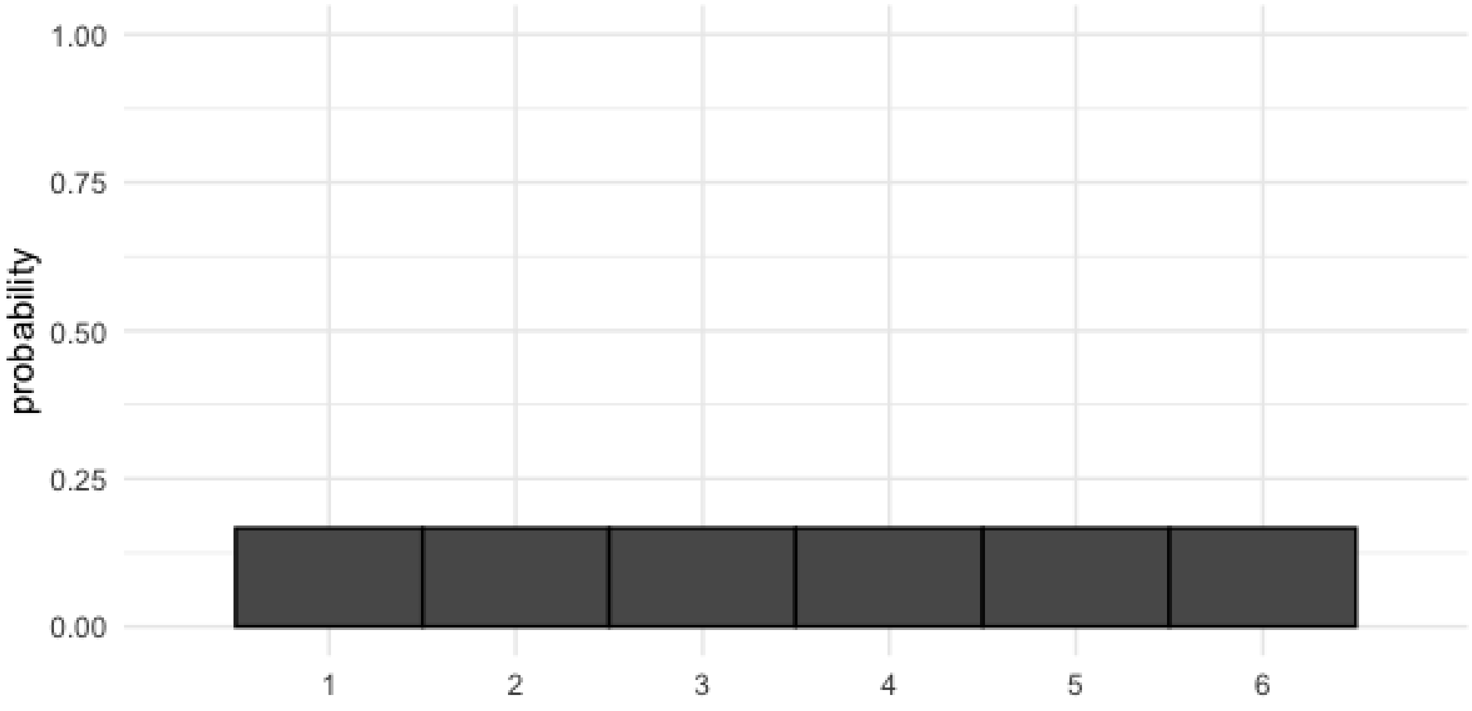
Sample distribution vs. theoretical distribution

Sample of 10 rolls



`np.mean(rolls_10['number'])` = `3.0`

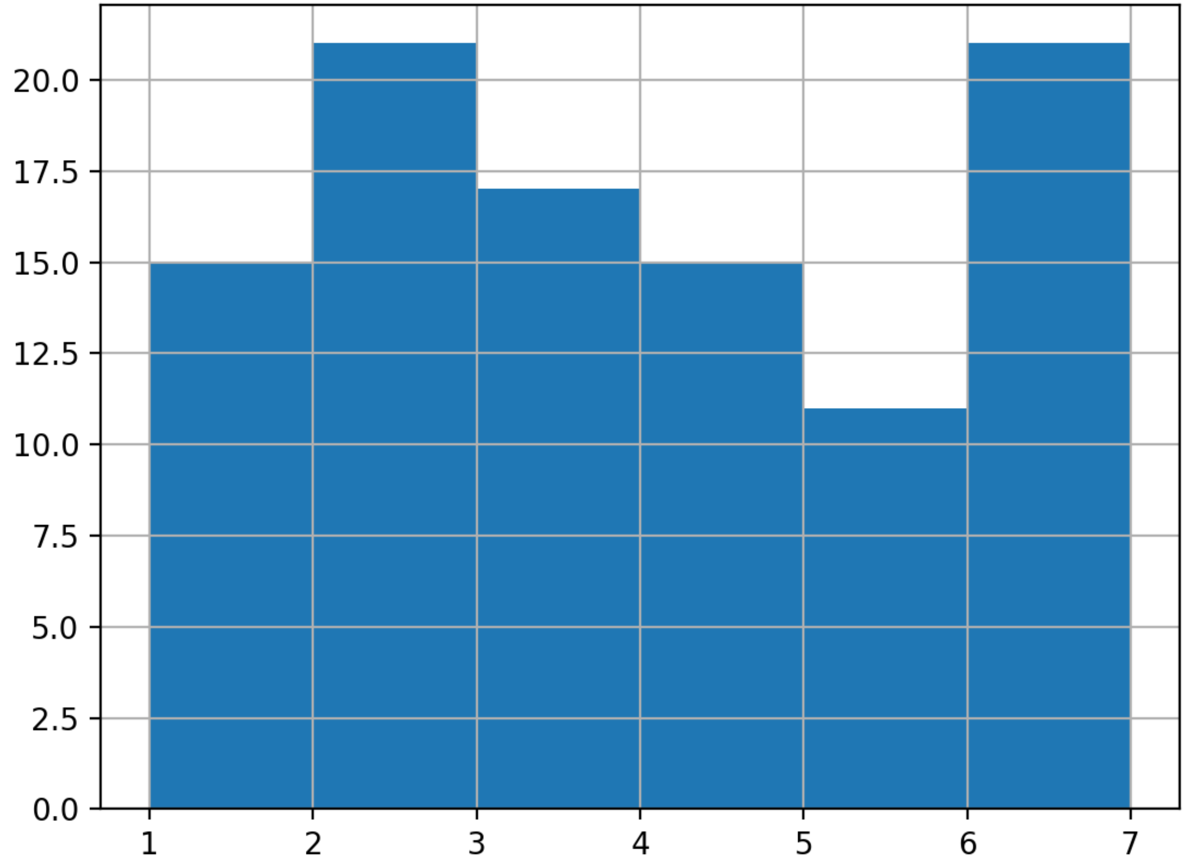
Theoretical probability distribution



`mean(die['number'])` = `3.5`

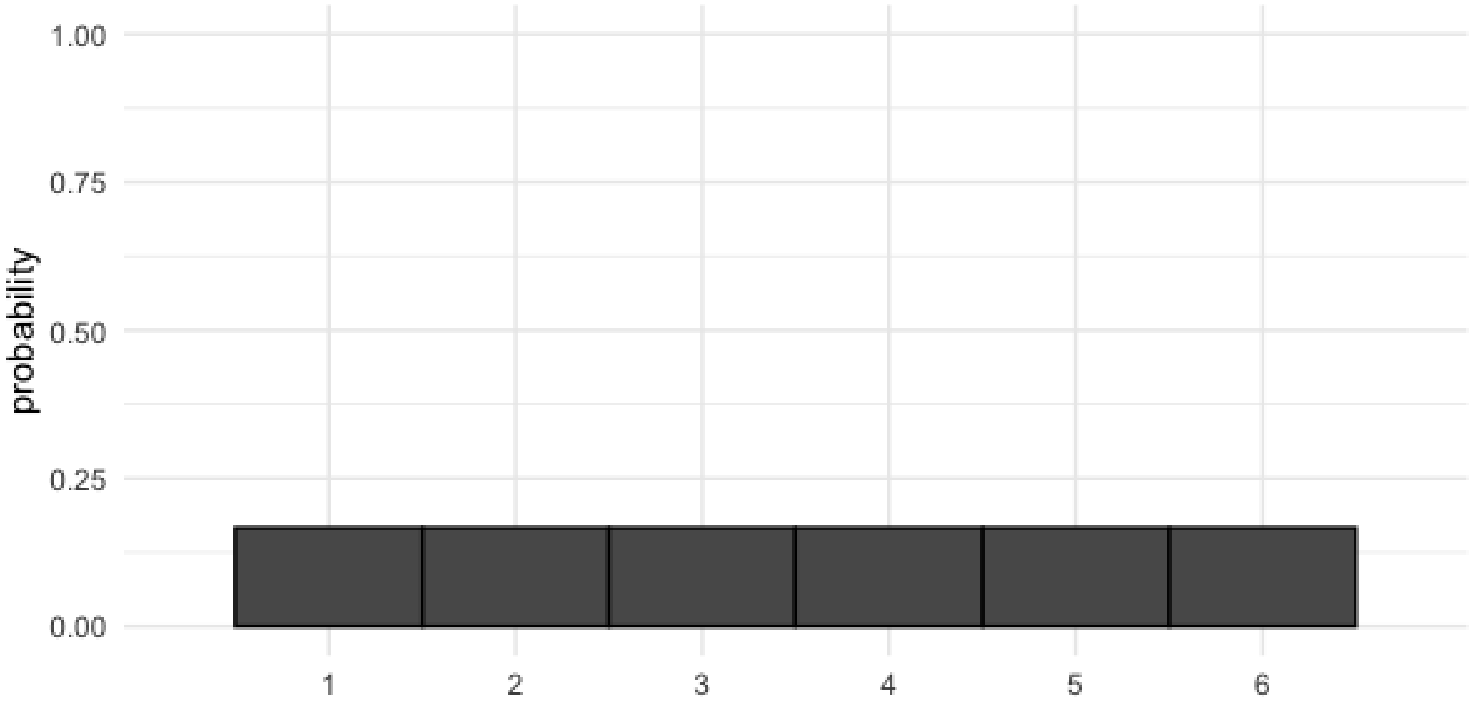
A bigger sample

Sample of 100 rolls



`np.mean(rolls_100['number'])` = `3.4`

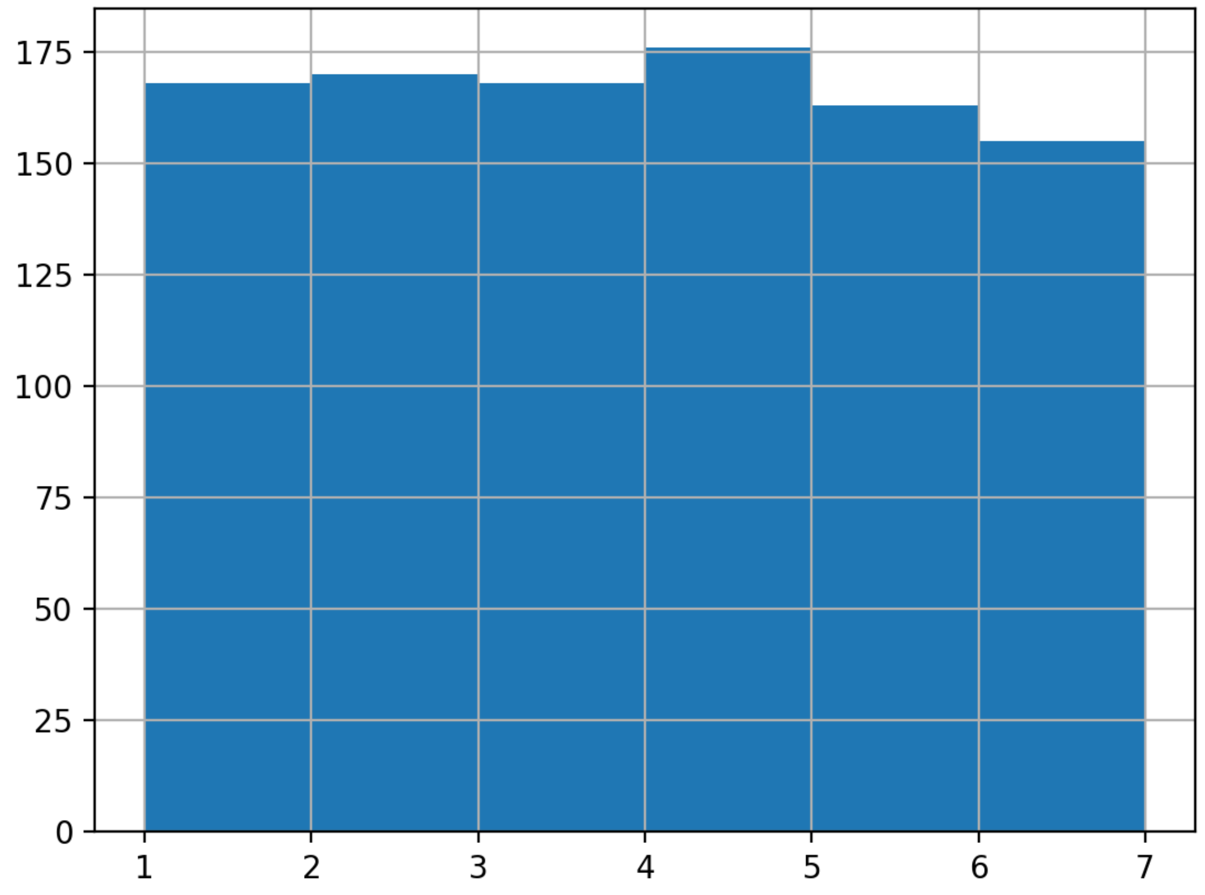
Theoretical probability distribution



`mean(die['number'])` = `3.5`

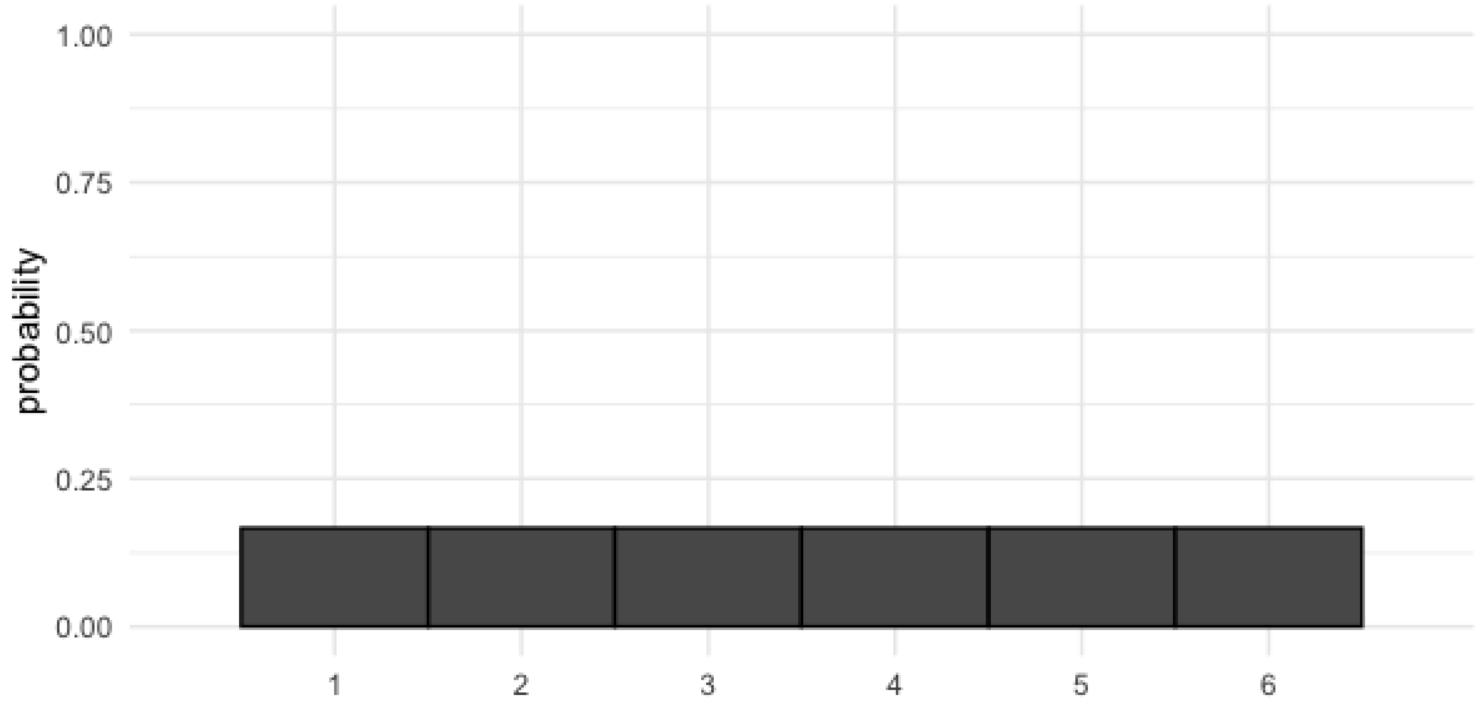
An even bigger sample

Sample of 1000 rolls



`np.mean(rolls_1000['number'])` = `3.48`

Theoretical probability distribution



`mean(die['number'])` = `3.5`

Law of large numbers

As the size of your sample increases, the sample mean will approach the expected value.

Sample size	Mean
10	3.00
100	3.40
1000	3.48

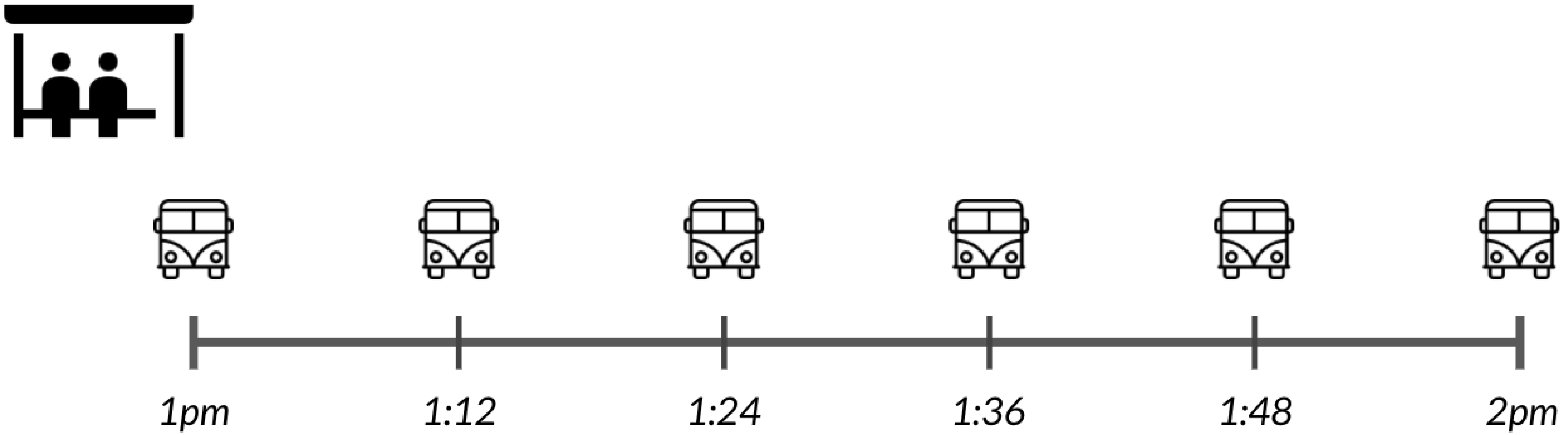
Let's practice!

Continuous distributions

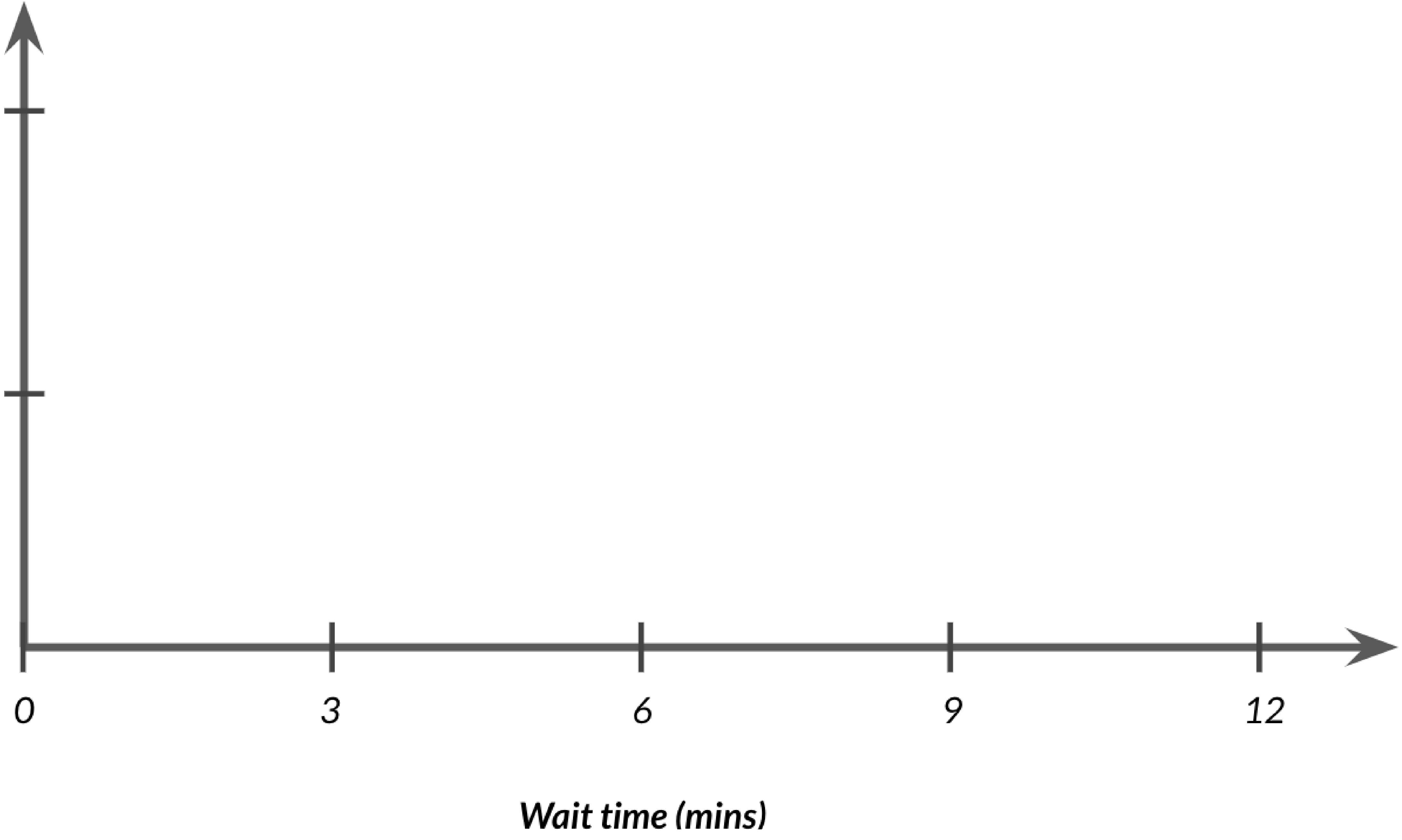
INTRODUCTION TO STATISTICS IN PYTHON

Maggie Matsui
Content Developer, DataCamp

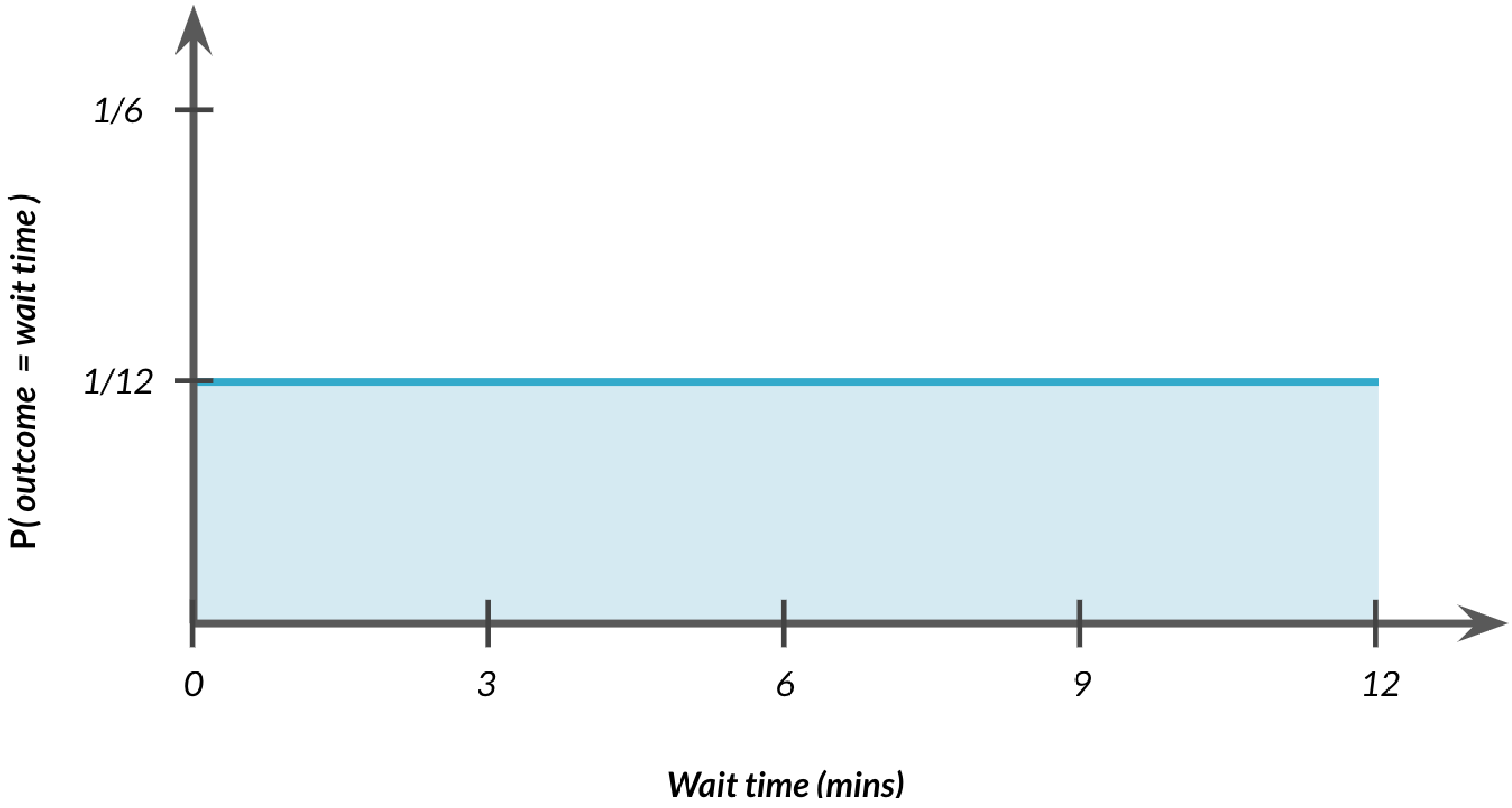
Waiting for the bus



Continuous uniform distribution

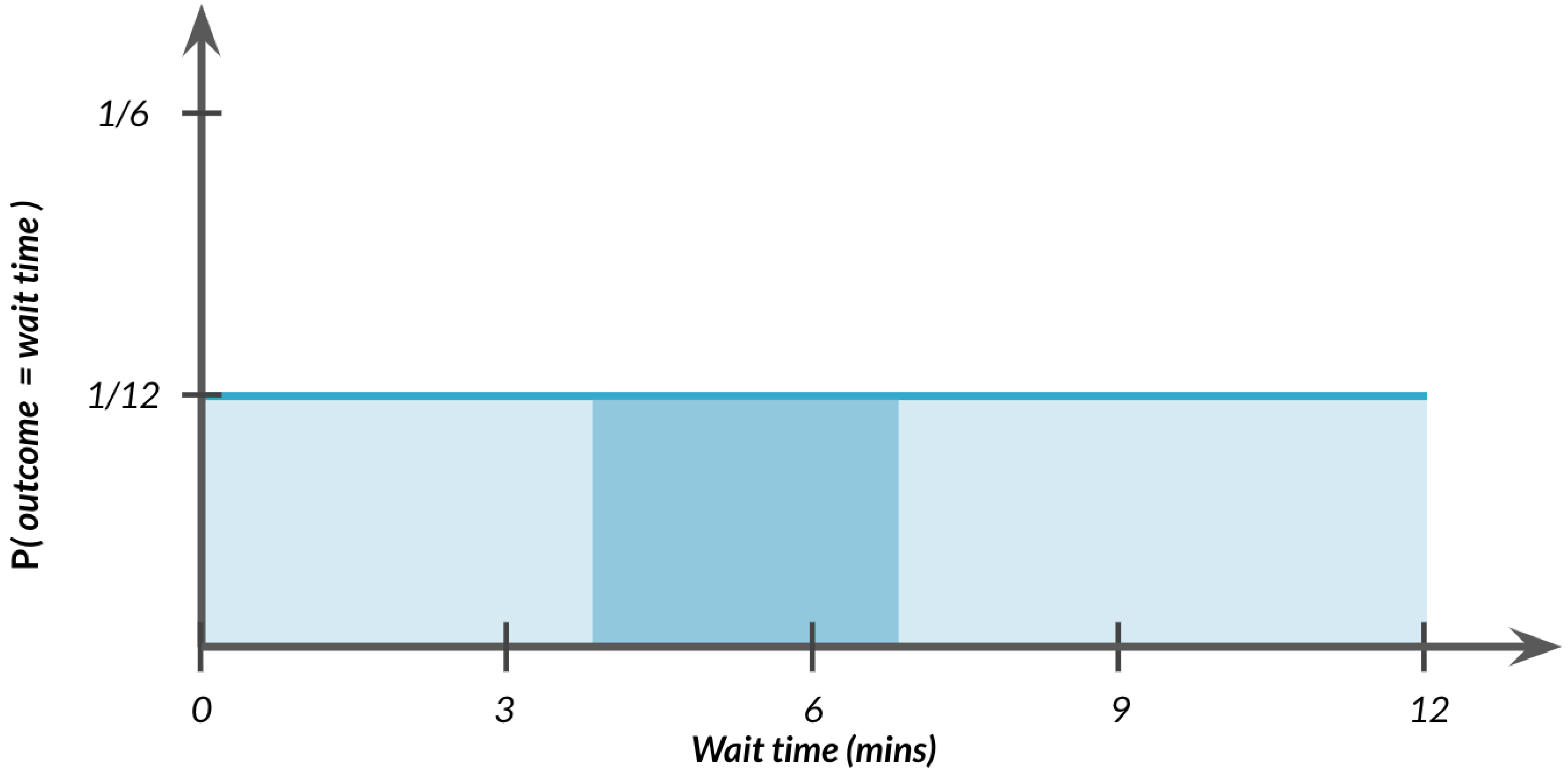


Continuous uniform distribution



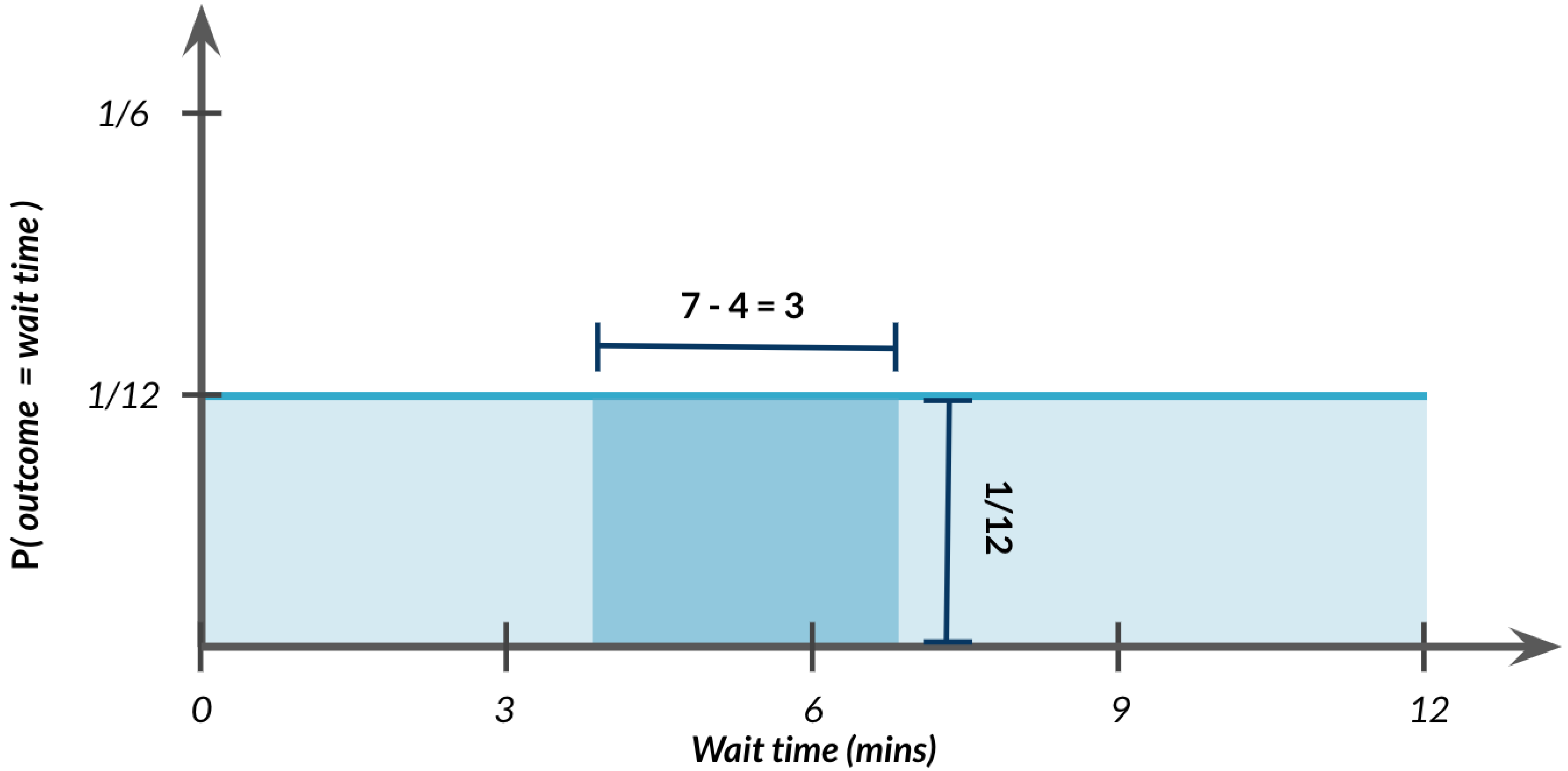
Probability still = area

$$P(4 \leq \text{wait time} \leq 7) = \ ?$$



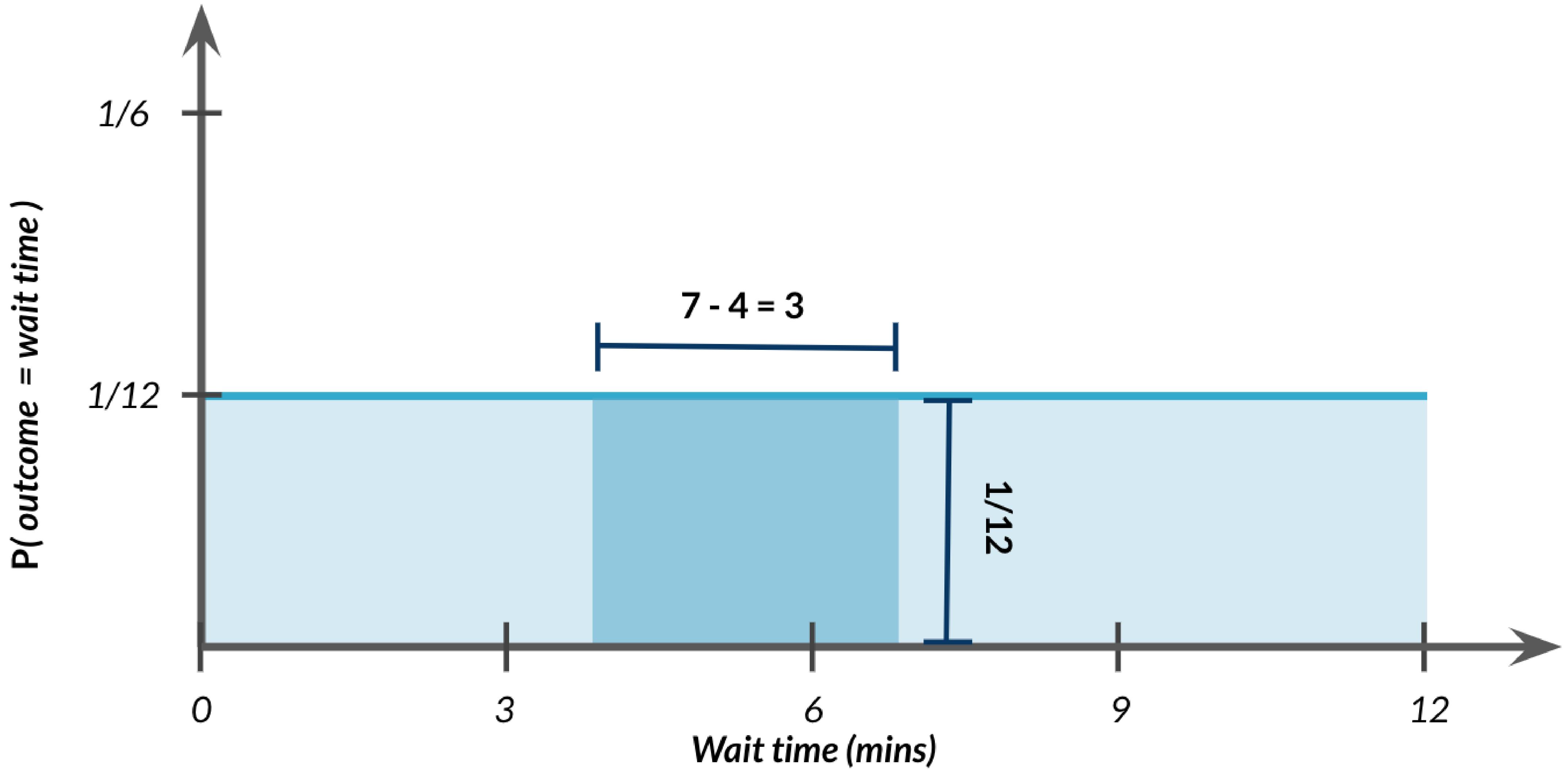
Probability still = area

$$P(4 \leq \text{wait time} \leq 7) = \ ?$$



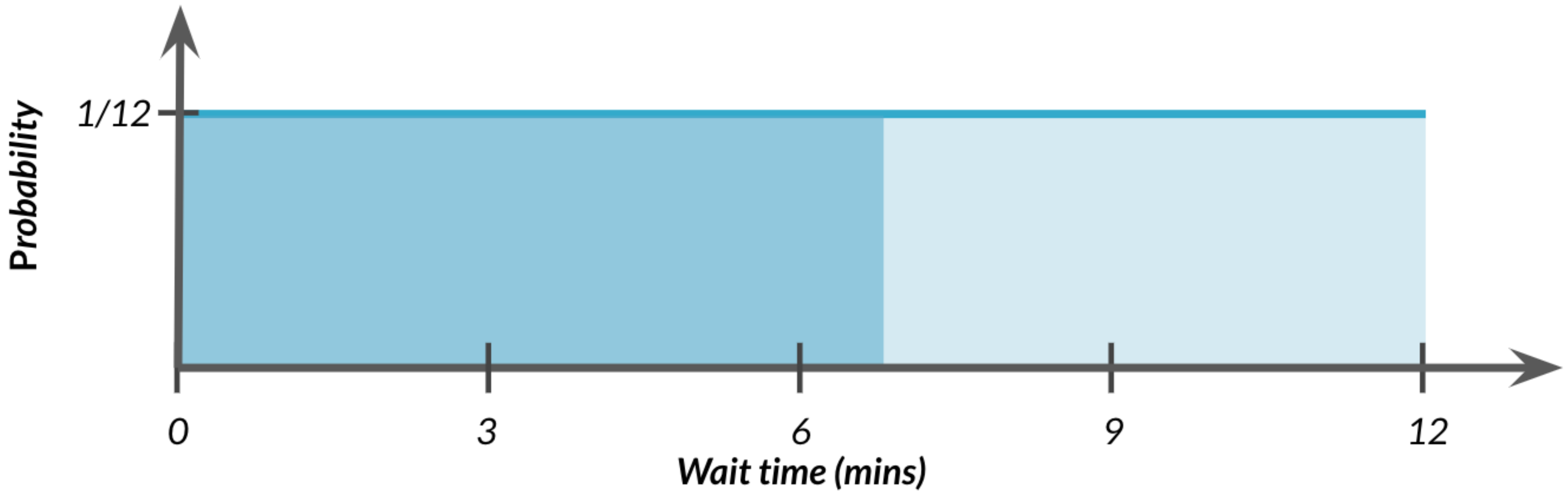
Probability still = area

$$P(4 \leq \text{wait time} \leq 7) = 3 \times 1/12 = 3/12$$



Uniform distribution in Python

$$P(\text{wait time} \leq 7)$$

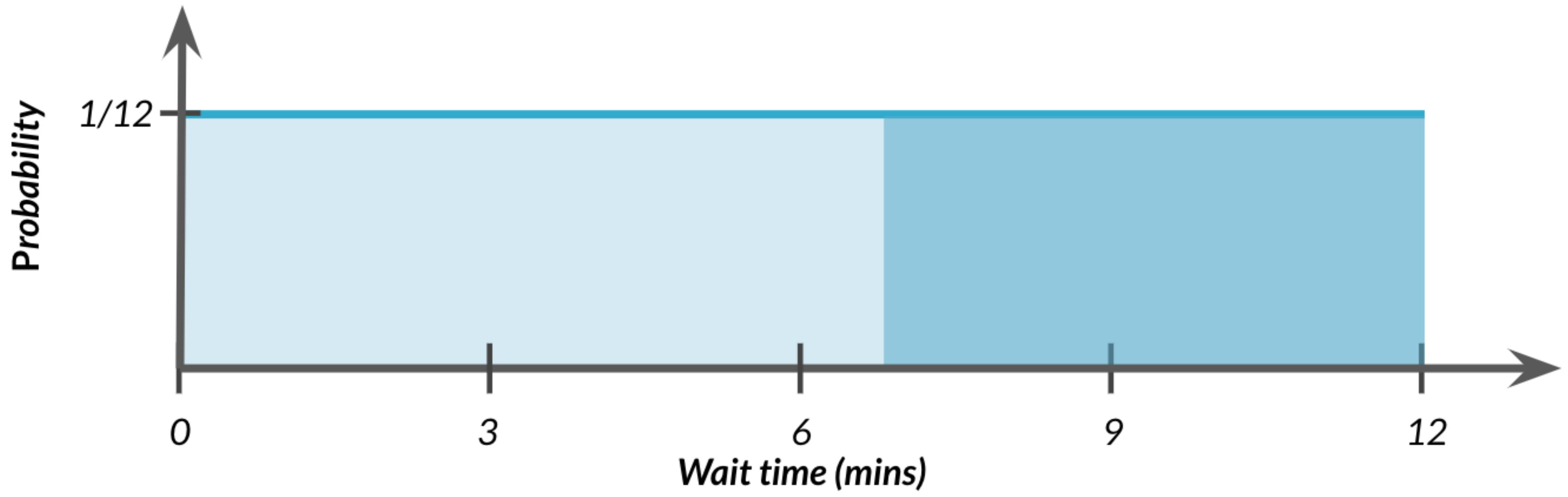



```
from scipy.stats import uniform
uniform.cdf(7, 0, 12)
```

```
0.5833333
```

"Greater than" probabilities

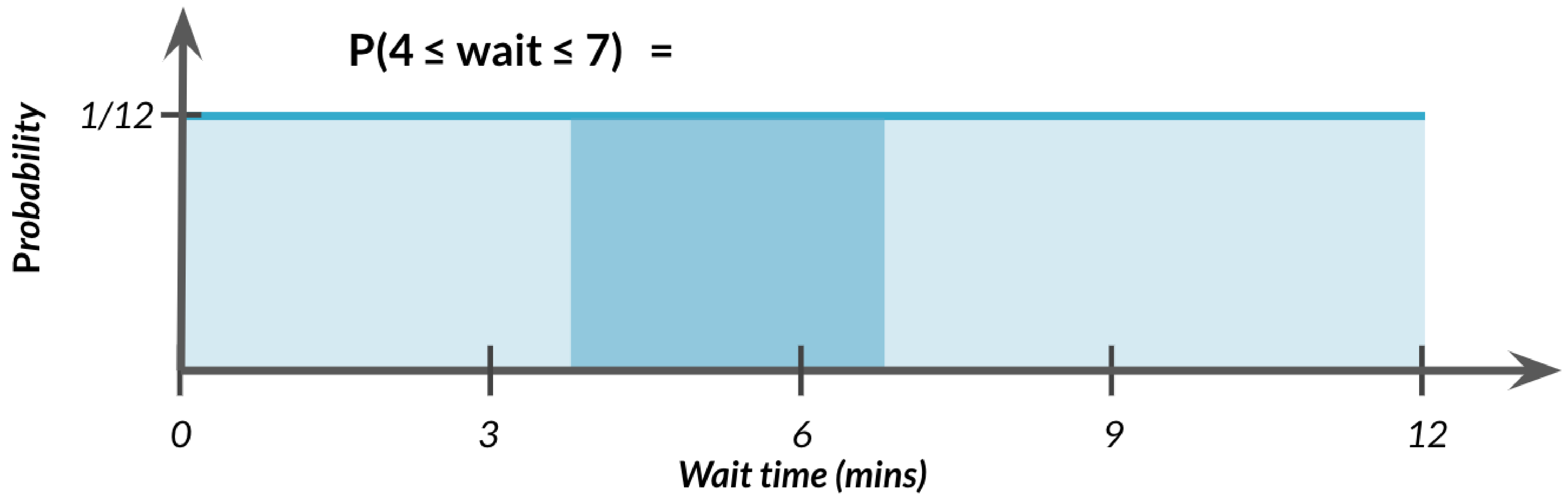
$$P(\text{wait time} \geq 7) = 1 - P(\text{wait time} \leq 7)$$



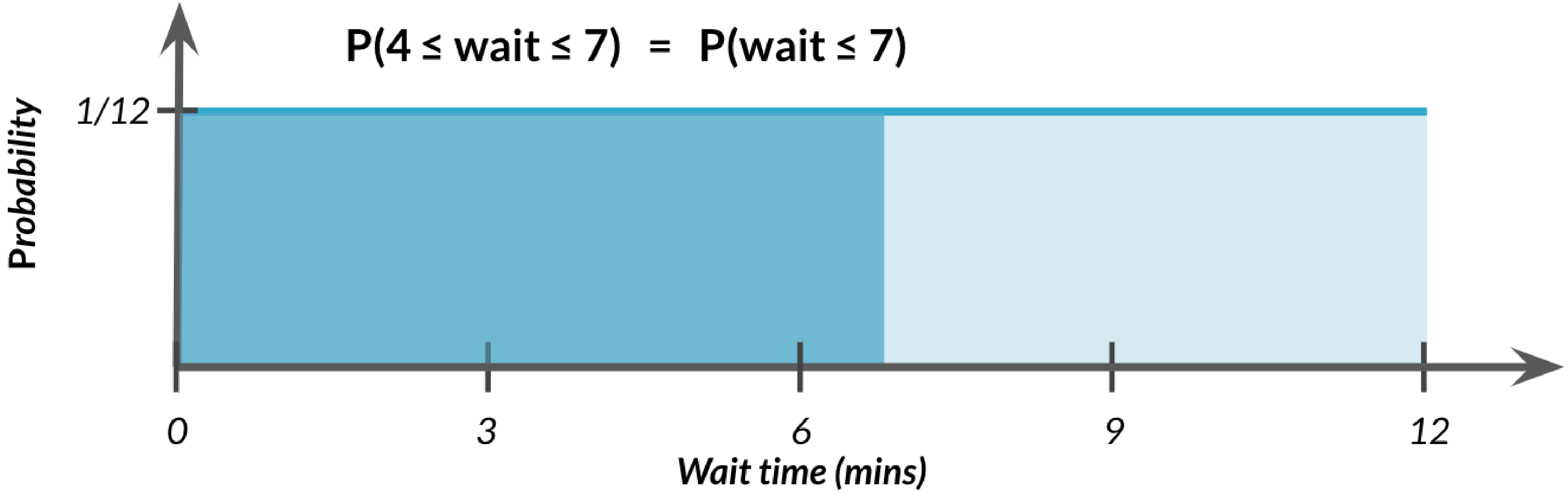

```python
from scipy.stats import uniform
1 - uniform.cdf(7, 0, 12)
```

```
0.4166667
```

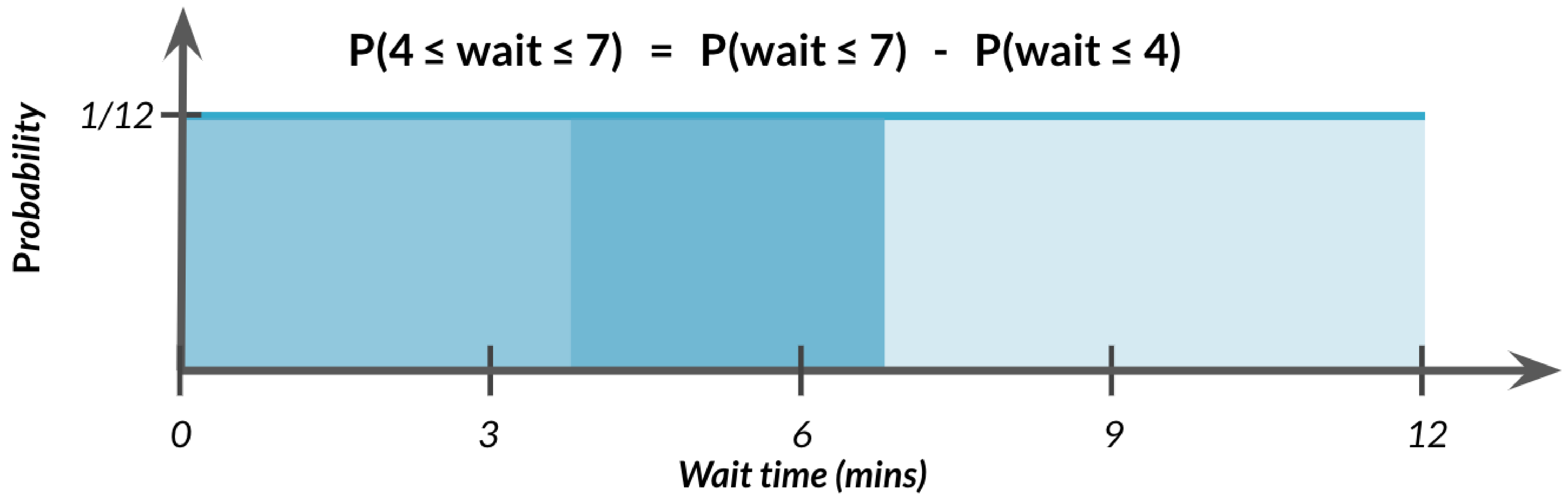
$$P(4 \le \text{wait time} \le 7)$$



$$P(4 \leq \text{wait time} \leq 7)$$



$$P(4 \leq \text{wait time} \leq 7)$$

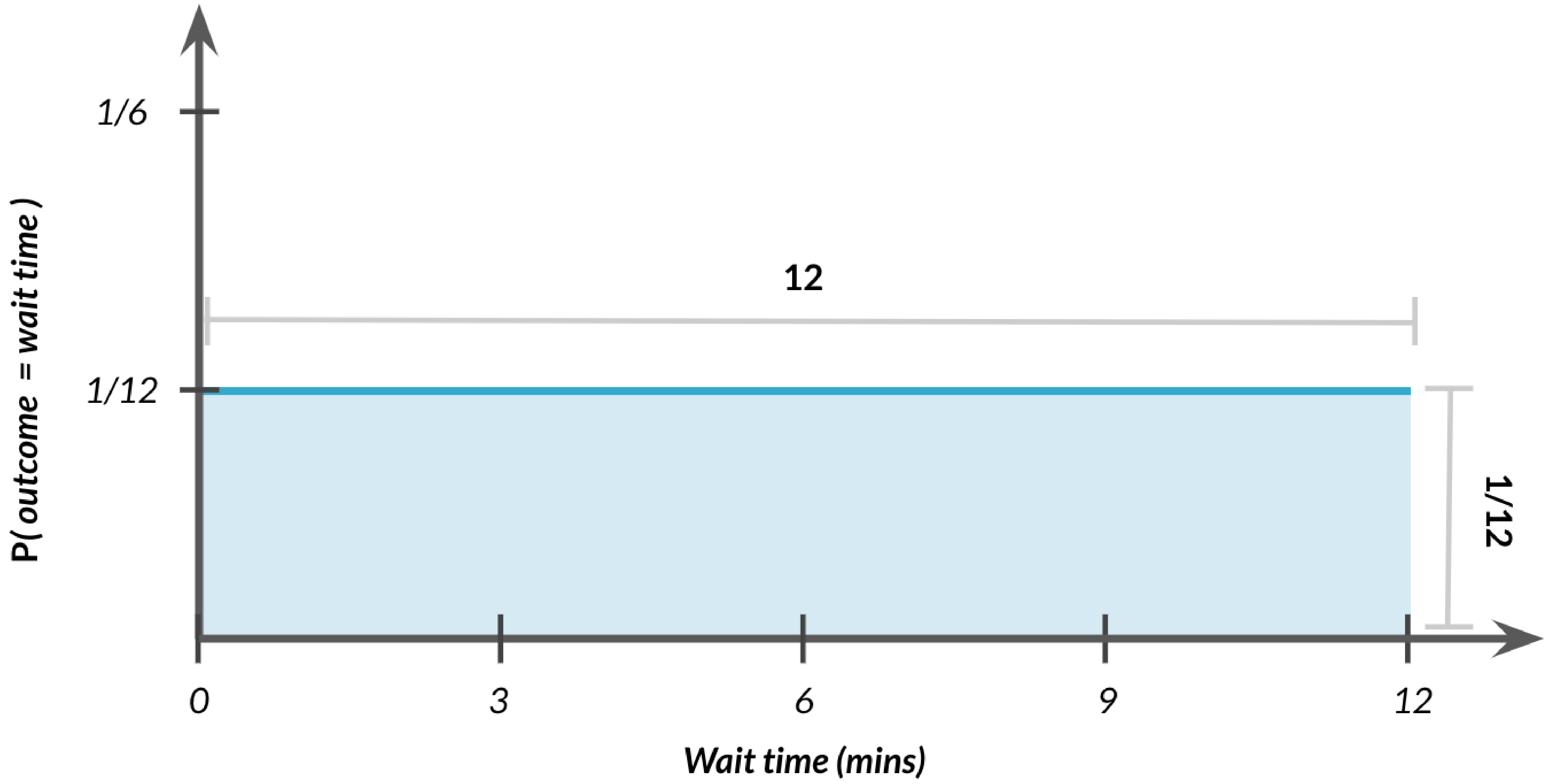



```python
from scipy.stats import uniform
uniform.cdf(7, 0, 12) - uniform.cdf(4, 0, 12)
```

```
0.25
```

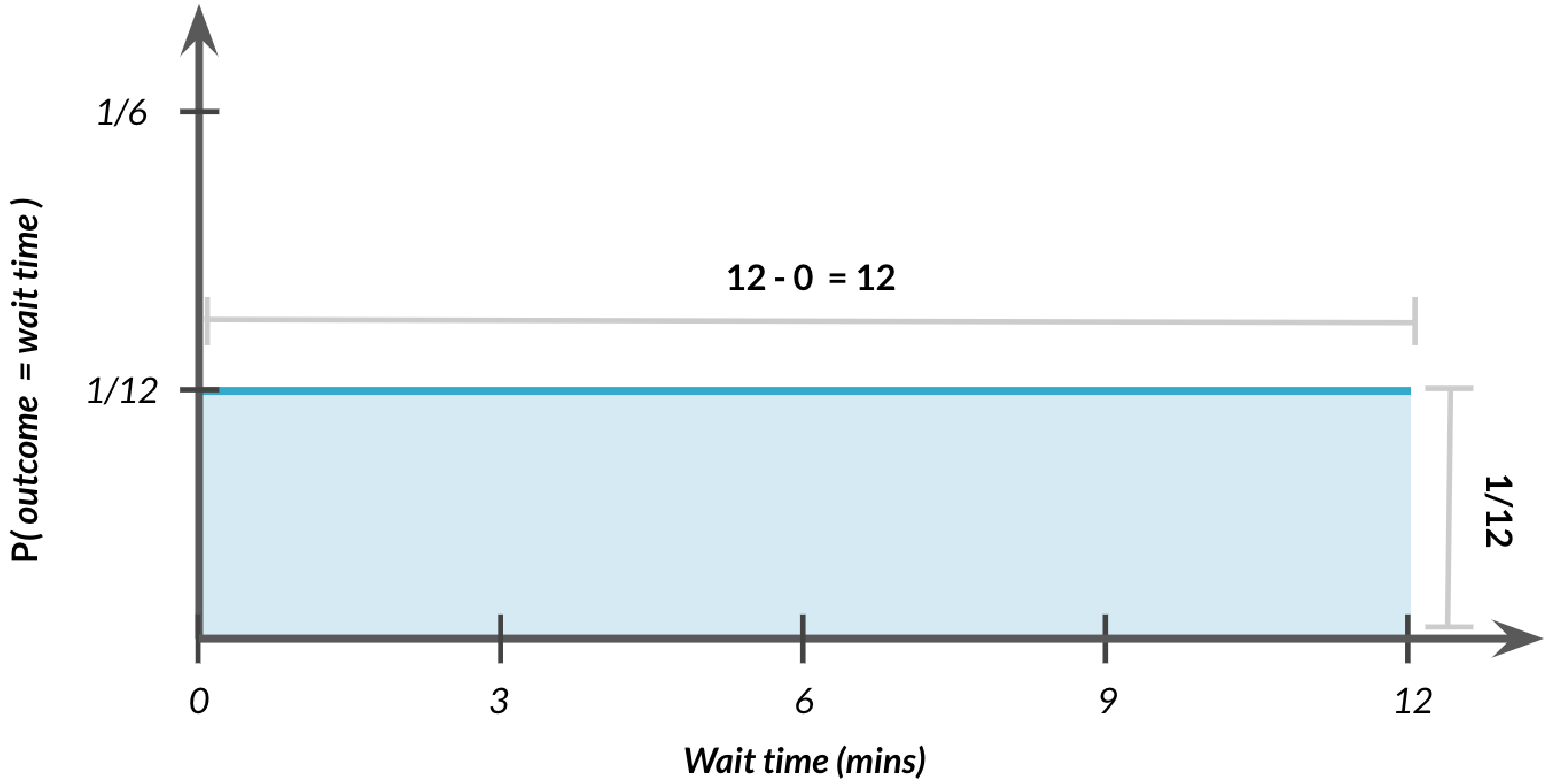
Total area = 1

$$P(0 \leq \text{wait time} \leq 12) = \ ?$$



Total area = 1

$$P(0 \leq \text{outcome} \leq 12) = 12 \times 1/12 = 1$$

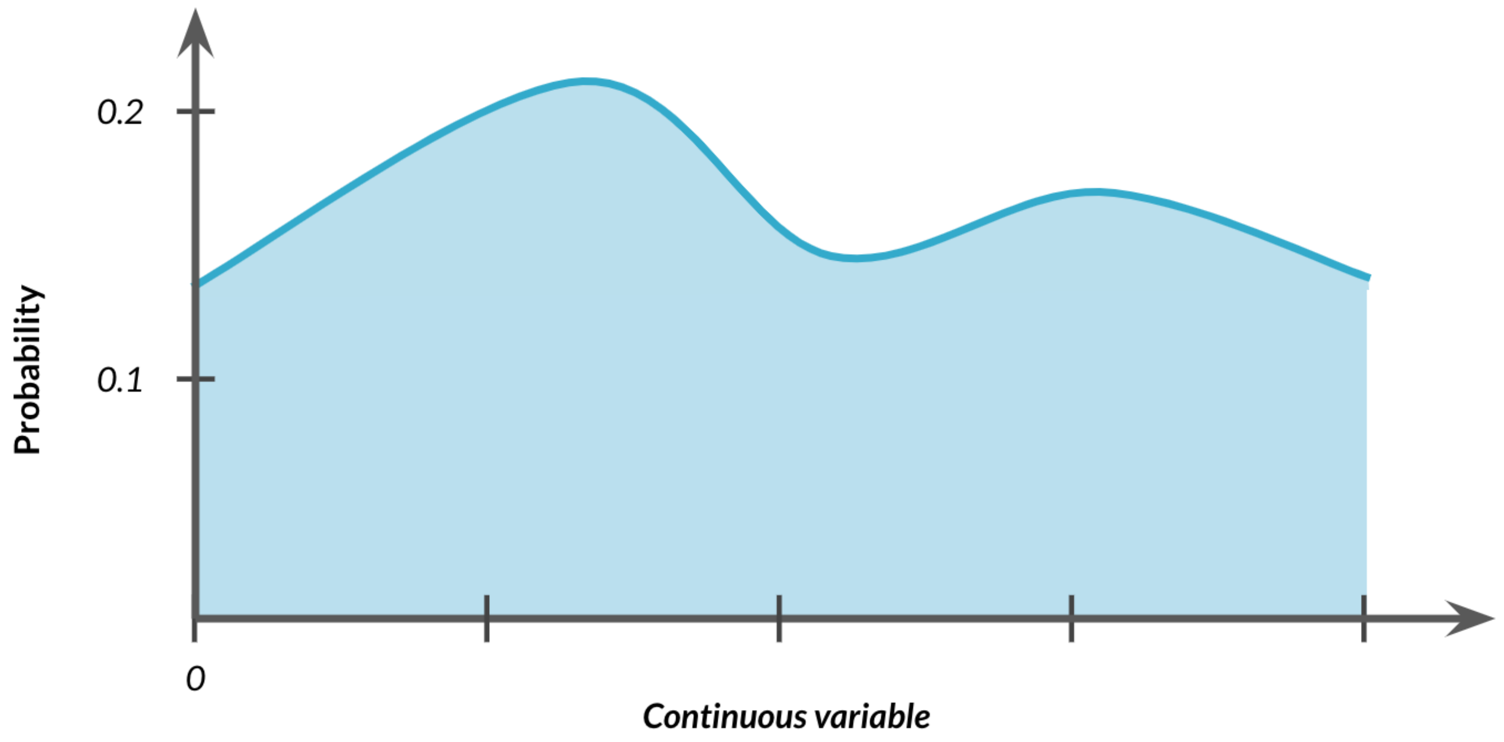


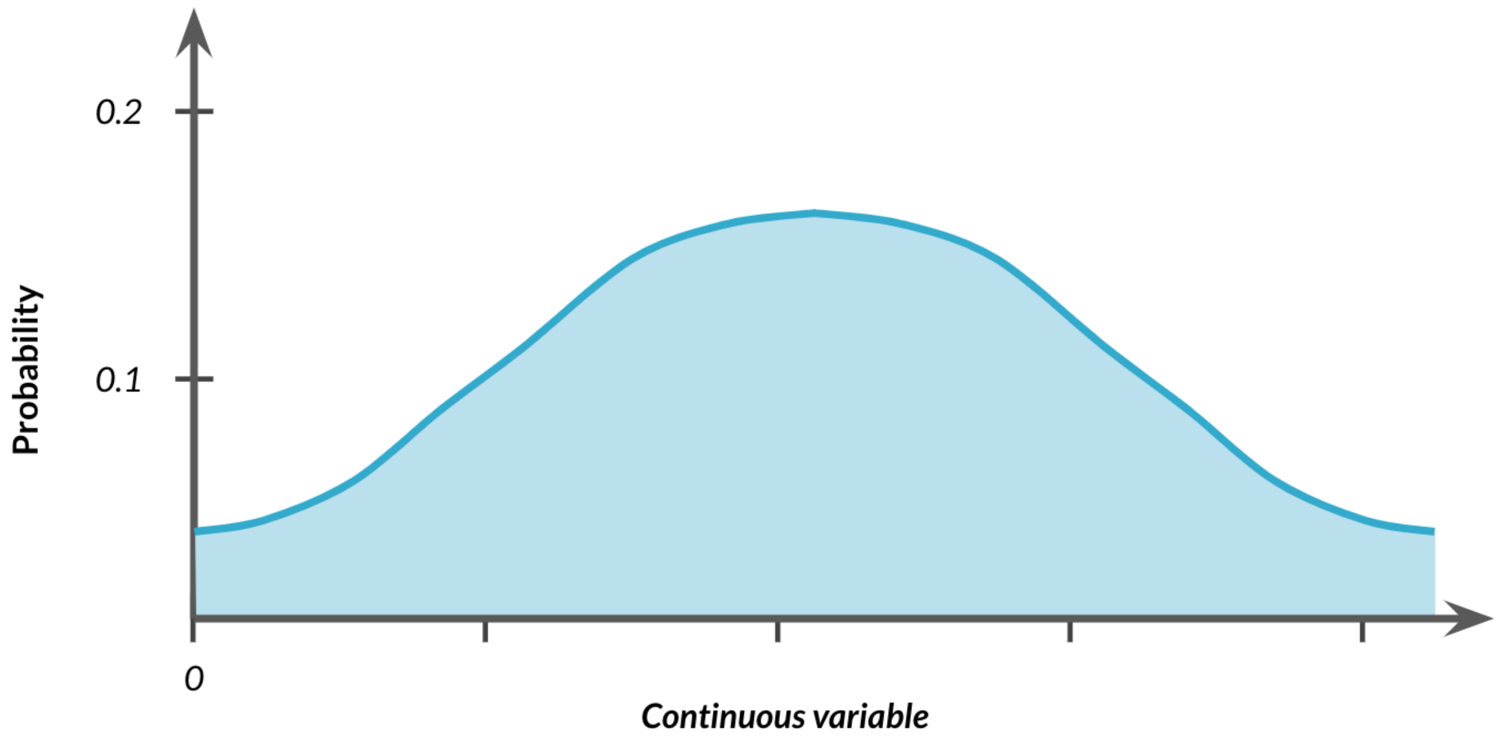
Generating random numbers according to uniform distribution

```
from scipy.stats import uniform
uniform.rvs(0, 5, size=10)
```

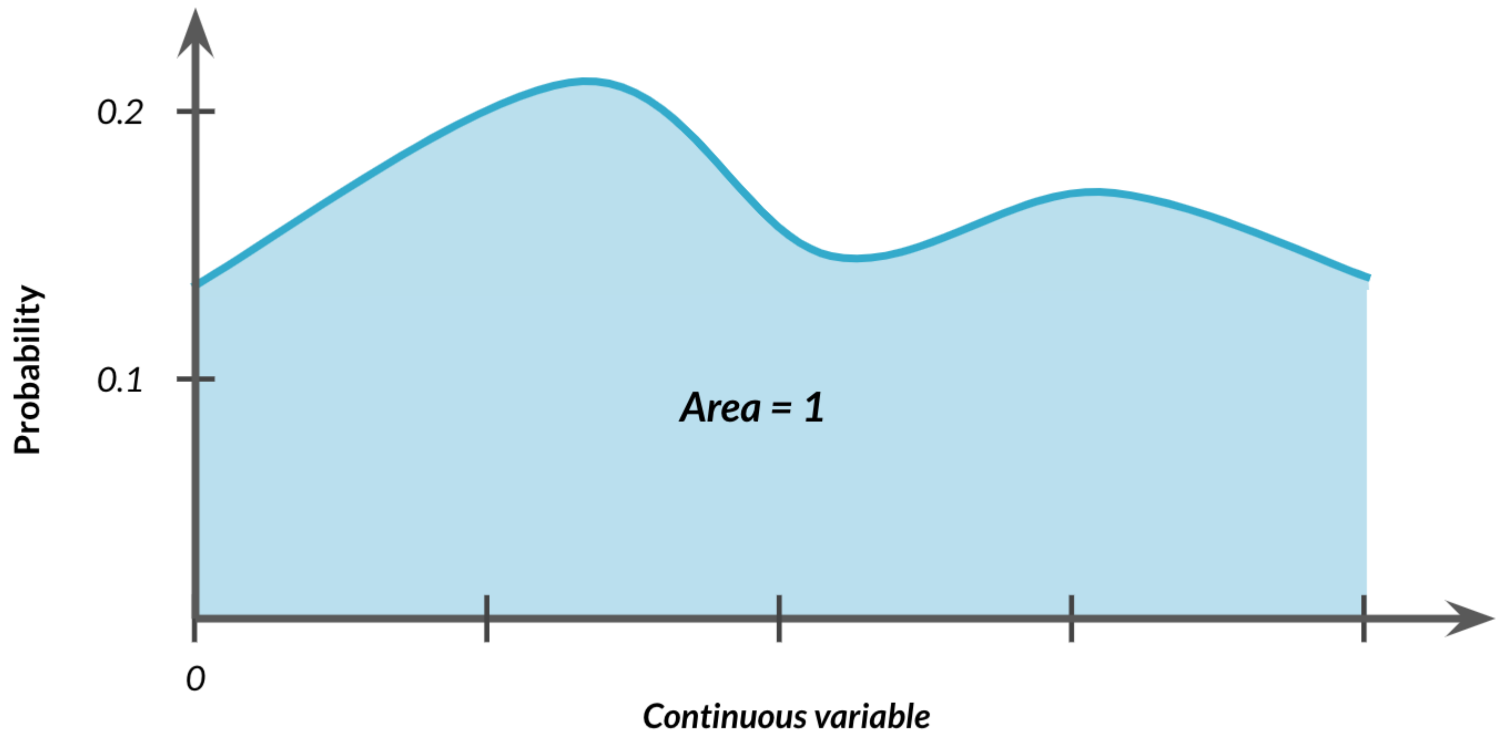
```
array([1.89740094, 4.70673196, 0.33224683, 1.0137103 , 2.31641255,
       3.49969897, 0.29688598, 0.92057234, 4.71086658, 1.56815855])
```

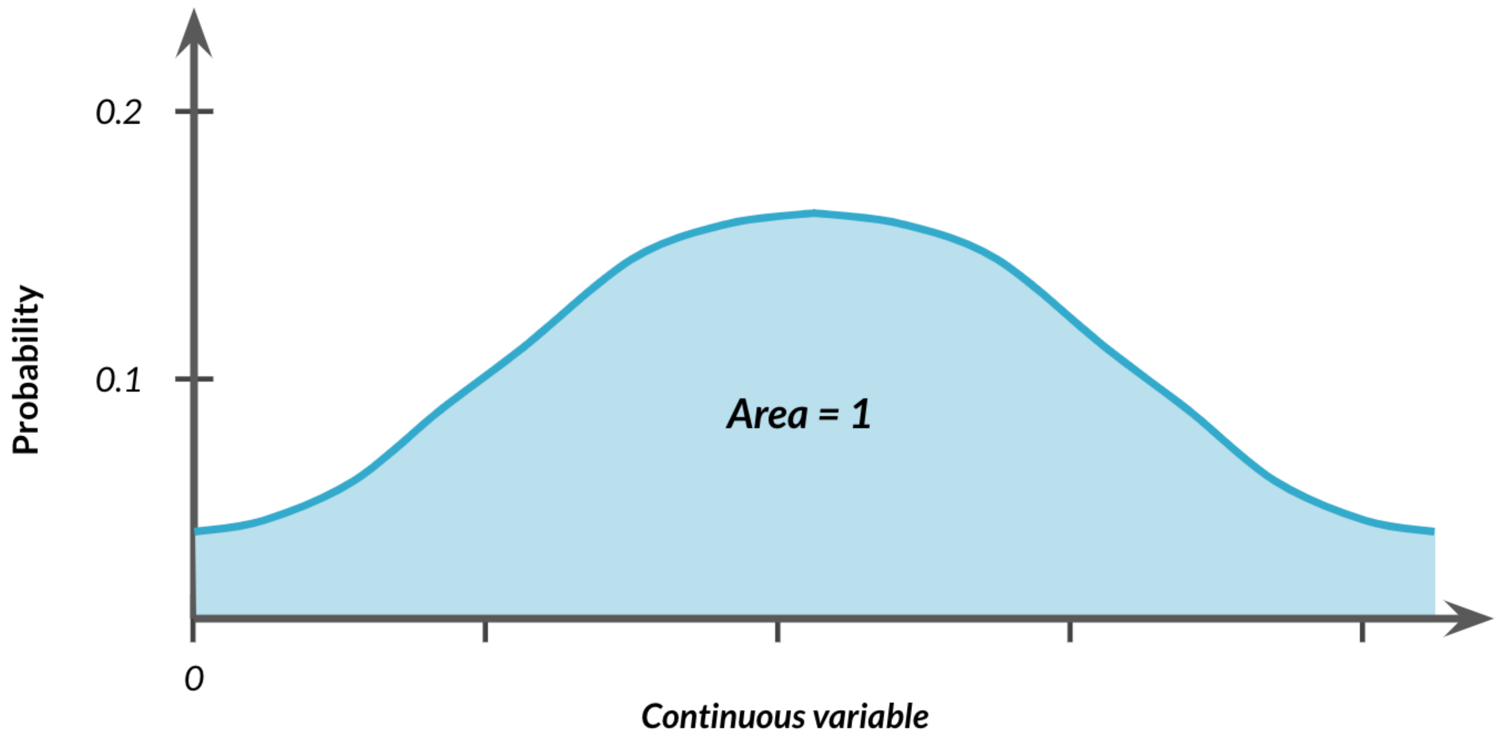
Other continuous distributions





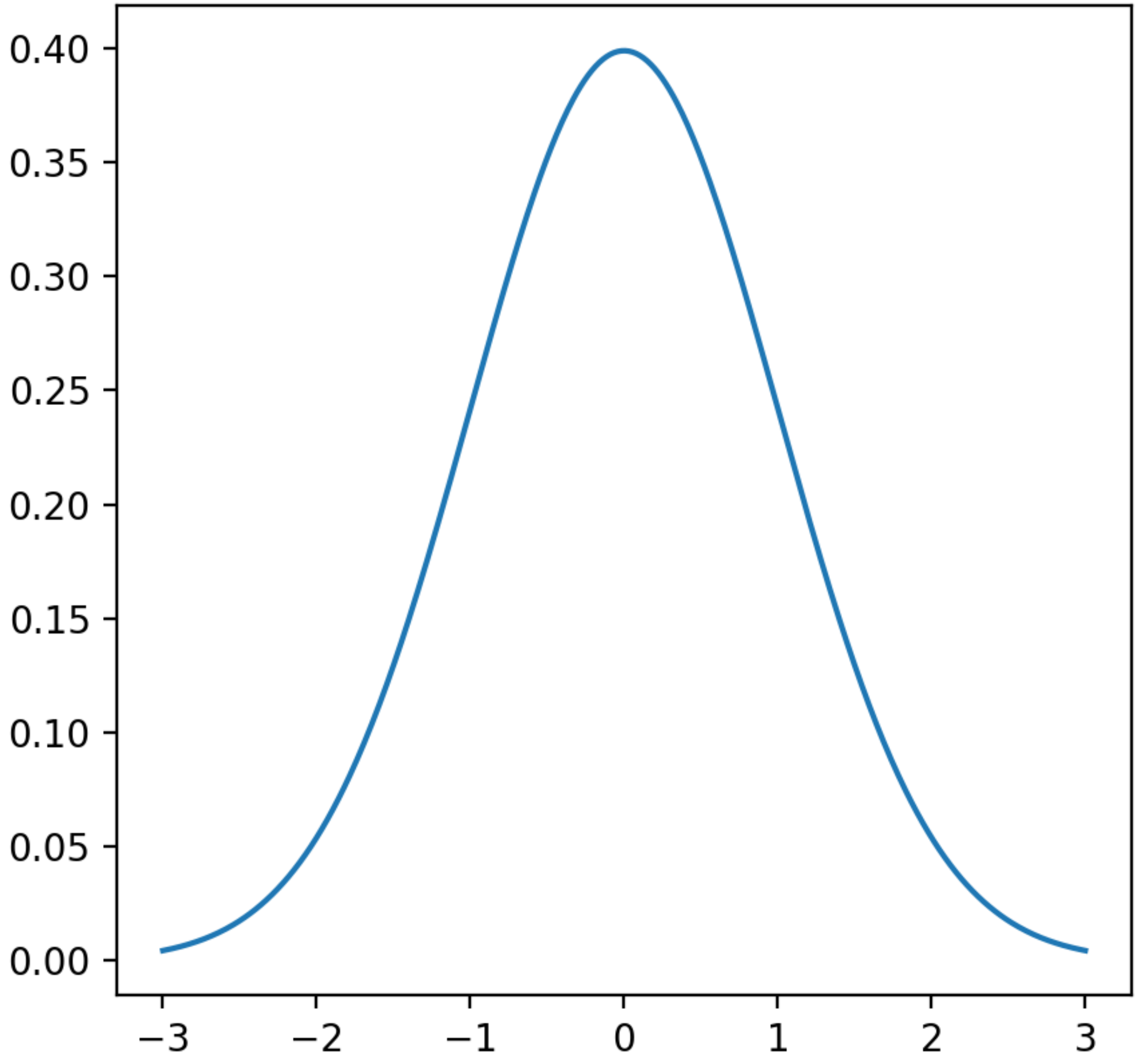
Other continuous distributions



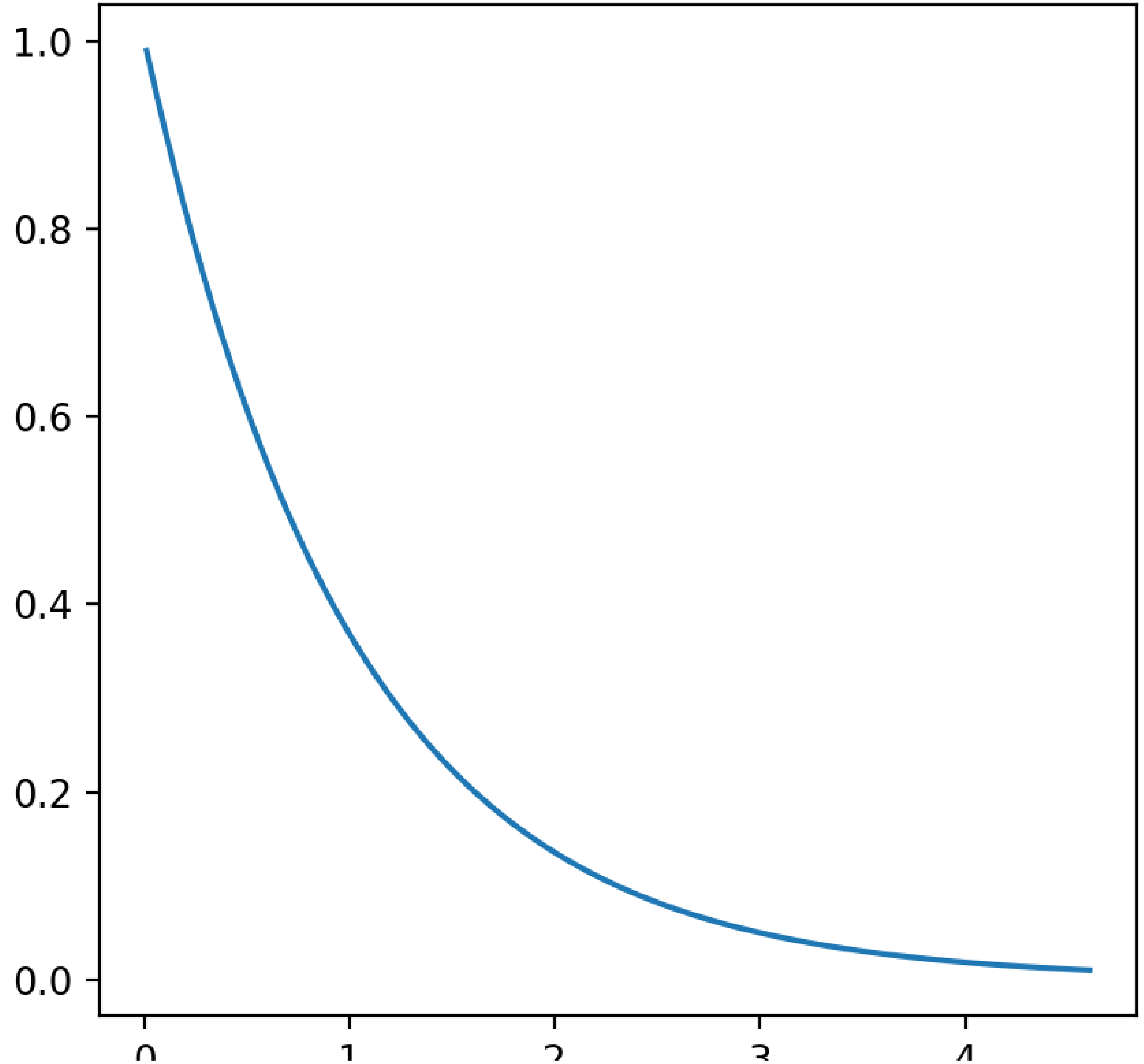


Other special types of distributions

Normal distribution



Exponential distribution



Let's practice!

The binomial distribution

INTRODUCTION TO STATISTICS IN PYTHON

Maggie Matsui
Content Developer, DataCamp

Coin flipping

50% 50%

H T

Binary outcomes

H	T
1	0
Success	Failure
Win	Loss

A single flip

`binom.rvs(# of coins, probability of heads/success, size=# of trials)`

`1` = head, `0` = tails

```
from scipy.stats import binom
binom.rvs(1, 0.5, size=1)
```

```
array([1])
```

One flip many times

```
binom.rvs(1, 0.5, size=8)
```

```
array([0, 1, 1, 0, 1, 0, 1, 1])
```

```
binom.rvs(1, 0.5, size = 8)
```

Flip 1 coin with 50% chance of success 8 times

Many flips one time

```
binom.rvs(8, 0.5, size=1)
```

```
array([5])
```

```
binom.rvs(8, 0.5, size = 1)
```

Flip 8 coins with 50% chance of success 1 time

Many flips many times

```
binom.rvs(3, 0.5, size=10)
```

```
array([0, 3, 2, 1, 3, 0, 2, 2, 0, 0])
```

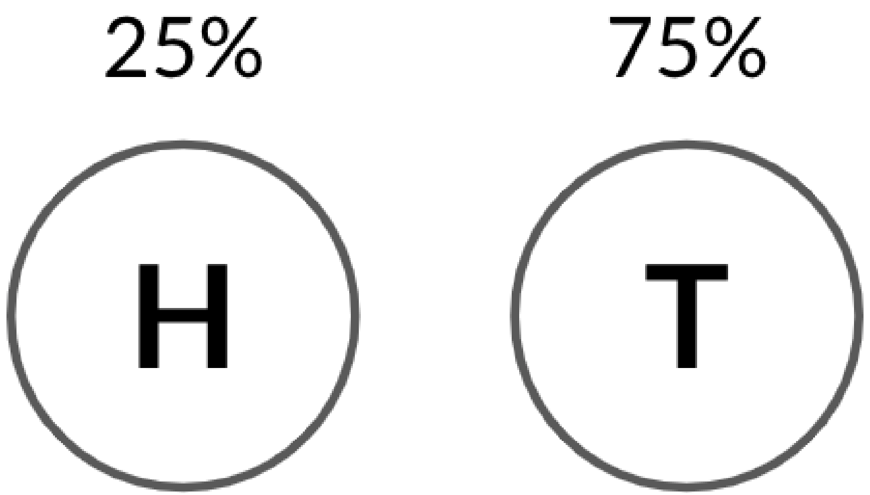
```
binom.rvs(3, 0.5, size = 10)
```

Flip 3 coins with 50% chance of success 10 times

Other probabilities

```
binom.rvs(3, 0.25, size=10)
```

```
array([1, 1, 1, 1, 0, 0, 2, 0, 1, 0])
```



Binomial distribution

Probability distribution of the number of successes in a sequence of independent trials

E.g. Number of heads in a sequence of coin flips

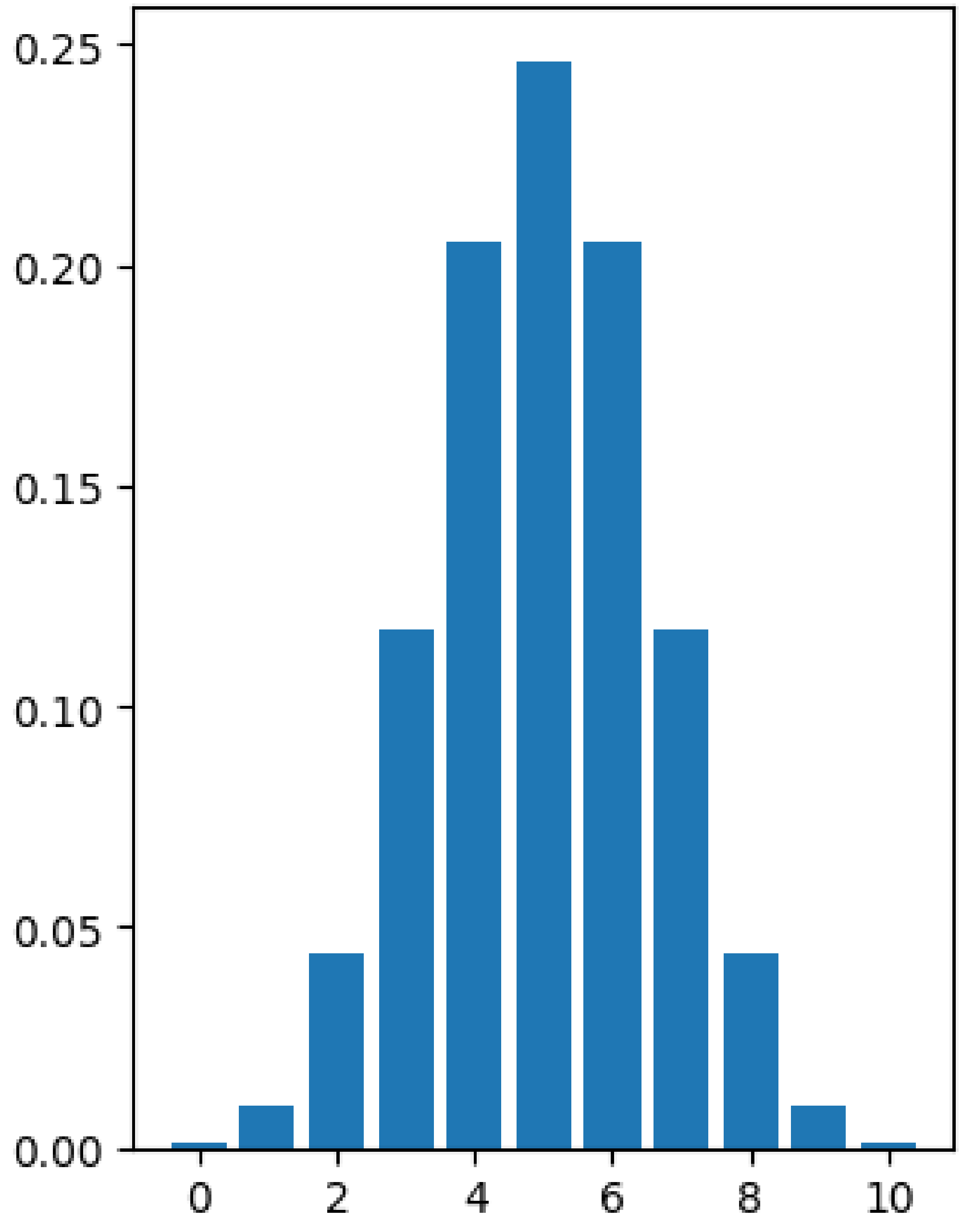
Described by n and p

- n: total number of trials
- p: probability of success

p **n**

```
binom.rvs(3, 0.5, size = 10)
```

Flip 3 coins with 50% chance of success 10 times



What's the probability of 7 heads?

$P(\text{heads} = 7)$

```
# binom.pmf(num heads, num trials, prob of heads)
binom.pmf(7, 10, 0.5)
```

```
0.1171875
```

What's the probability of 7 or fewer heads?

$P(\text{heads} \leq 7)$

```
binom.cdf(7, 10, 0.5)
```

```
0.9453125
```

What's the probability of more than 7 heads?

$P(\text{heads} > 7)$

```
1 - binom.cdf(7, 10, 0.5)
```

```
0.0546875
```

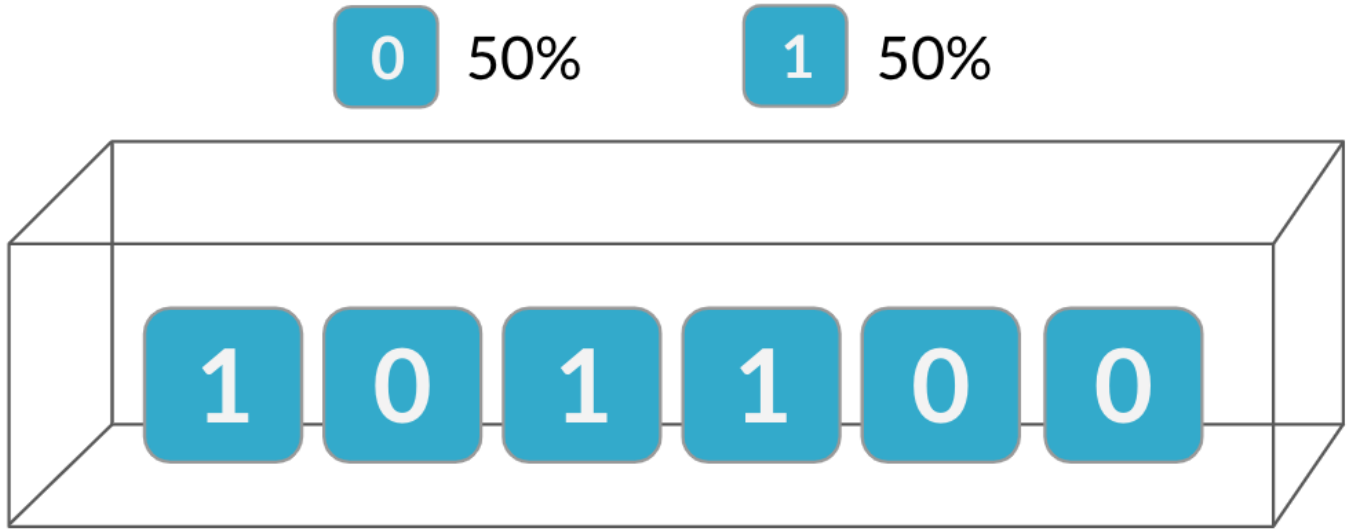
Expected value

$$\text{Expected value} = n \times p$$

Expected number of heads out of 10 flips $= 10 \times 0.5 = 5$

Independence

The binomial distribution is a probability distribution of the number of successes in a sequence of ***independent*** *trials*

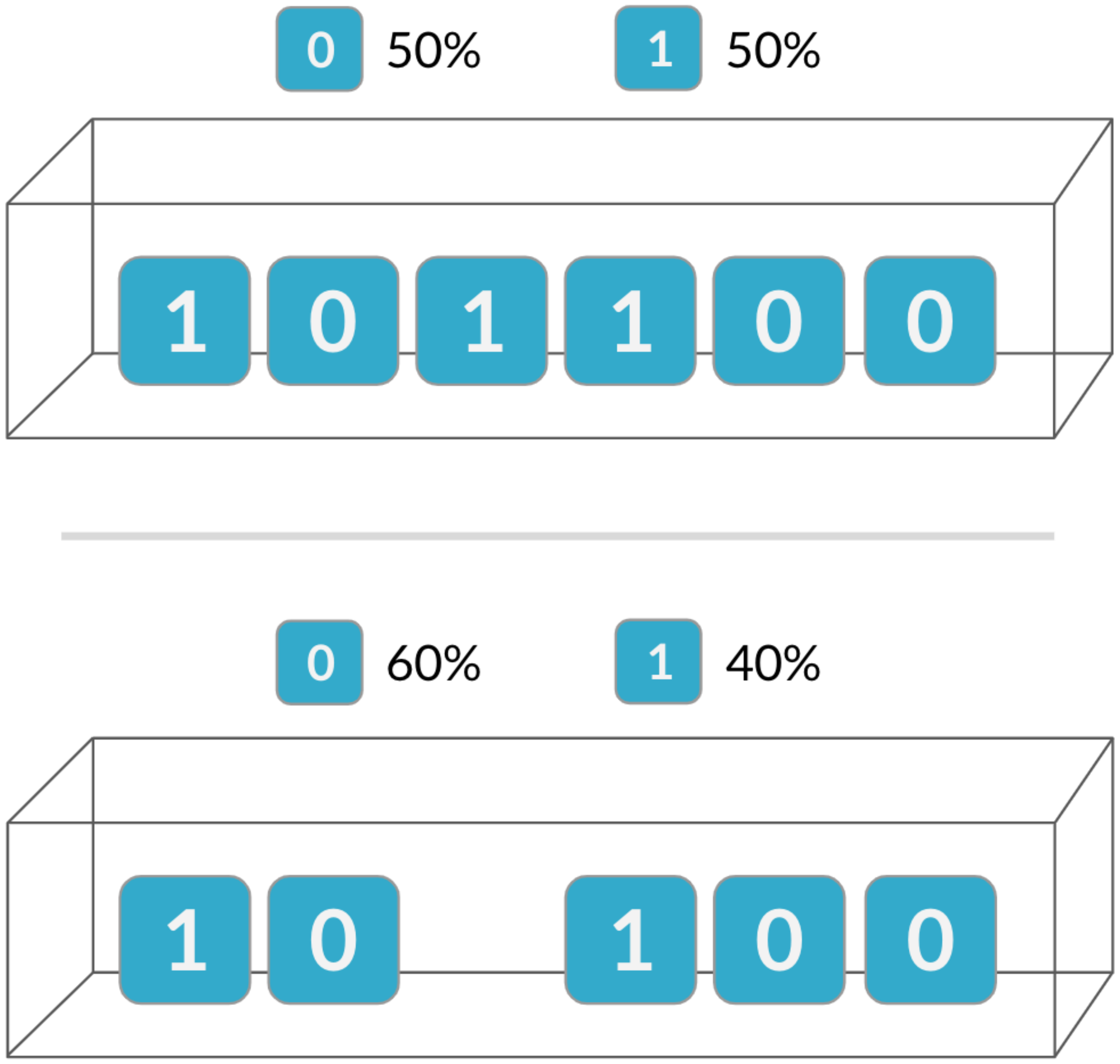


Independence

*The binomial distribution is a probability distribution of the number of successes in a sequence of **independent** trials*

Probabilities of second trial are altered due to outcome of the first

If trials are not independent, the binomial distribution does not apply!



Let's practice!

INTRODUCTION TO STATISTICS IN PYTHON