

Python Code for QSS Chapter 6: Probability

Kosuke Imai, Python code by Jeff Allen

First Printing

```
[ ]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from math import comb, exp, factorial, log
```

Section 6.1: Probability

Section 6.1.1: Frequentist versus Bayesian

Section 6.1.2: Definition and Axioms

Section 6.1.3: Permutations

```
[ ]: def birthday(k):
    logdenom = k * log(365) + log(factorial(365 - k)) # log denominator
    lognumer = log(factorial(365)) # log numerator
    # P(at least two have the same bday) = 1 - P(nobody has the same bday)
    pr = 1 - exp(lognumer - logdenom) # transform back
    return pr

k = pd.Series(np.arange(1, 51))

bday = k.apply(birthday) # apply the function to each element of k

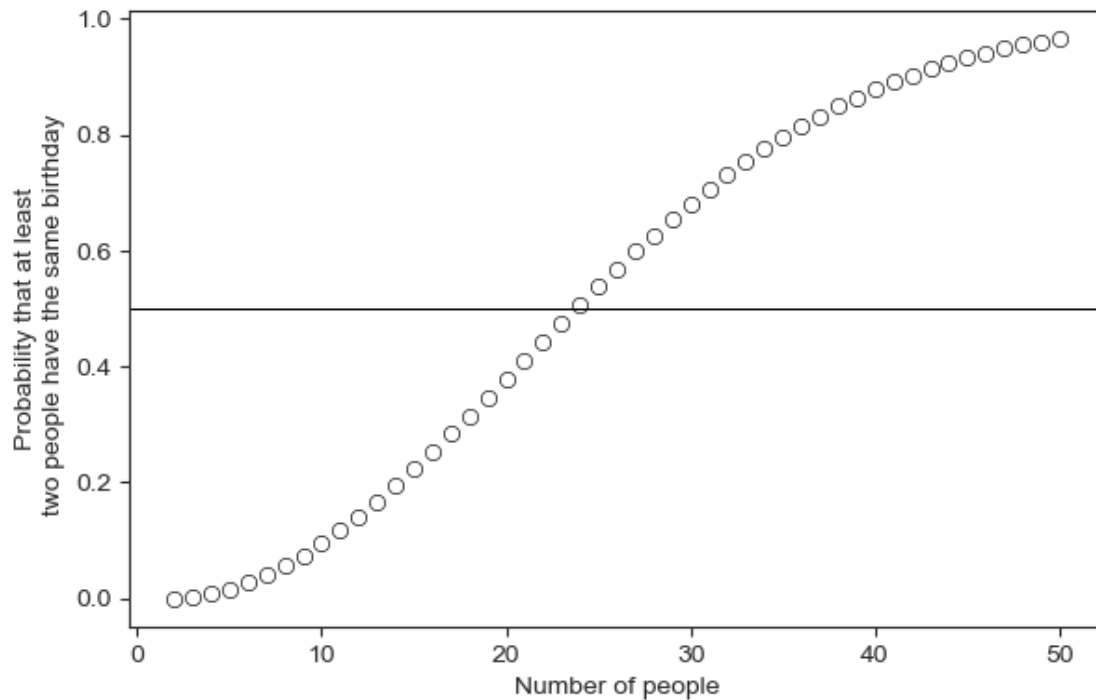
bday.index = k # add labels

sns.set_style('ticks')

sns.relplot(
    x=k, y=bday, color='white', edgecolor='black', height=4, aspect=1.5
).set(ylabel='Probability that at least\n two people have the same birthday',
      xlabel='Number of people').despine(right=False, top=False)

# horizontal line at 0.5
plt.axhline(0.5, color='black', linewidth=0.75)
```

```
[ ]: <matplotlib.lines.Line2D at 0x18e6a345840>
```



```
[ ]: bday.loc[20:25]
```

```
[ ]: 20    0.411438
      21    0.443688
      22    0.475695
      23    0.507297
      24    0.538344
      25    0.568700
      dtype: float64
```

Section 6.1.4: Sampling With and Without Replacement

```
[ ]: k = 23 # number of people
      sims = 10000 # number of simulations
      event = 0 # initialize counter

      for i in range(sims):
          days = np.random.choice(np.arange(1,366), size=k, replace=True)
          days_unique = np.unique(days) # number of unique days
          '''
          if there are duplicates, the number of unique birthdays will be less than
          the number of birthdays, which is 'k'
          '''
          if len(days_unique) < len(days):
```

```
        event += 1

answer = event / sims
answer
```

```
[ ]: 0.5032
```

Section 6.1.5: Combinations

```
[ ]: comb(84, 6)
```

```
[ ]: 406481544
```

Section 6.2: Conditional Probability

In Progress