simulating_coin_flips

November 28, 2017

1 Simulating Coin Flips

```
In [1]: import numpy as np
        import matplotlib.pyplot as plt
        % matplotlib inline

In [2]: # outcome of one coin flip
        np.random.randint(2)

Out[2]: 1

In [3]: # outcomes of ten thousand coin flips
        np.random.randint(2, size=10000)

Out[3]: array([0, 0, 1, ..., 0, 1, 0])

In [4]: # mean outcome of ten thousand coin flips
        np.random.randint(2, size=10000).mean()

Out[4]: 0.49840000000000001

In [5]: # outcome of one coin flip
        np.random.choice([0, 1])

Out[5]: 1

In [6]: # outcome of ten thousand coin flips
        np.random.choice([0, 1], size=10000)

Out[6]: array([0, 0, 1, ..., 1, 0, 1])

In [7]: # mean outcome of ten thousand coin flips
        np.random.choice([0, 1], size=10000).mean()

Out[7]: 0.49409999999999998

In [8]: # outcomes of ten thousand biased coin flips
        np.random.choice([0, 1], size=10000, p=[0.8, 0.2])

Out[8]: array([0, 1, 0, ..., 0, 0, 0])

In [9]: # mean outcome of ten thousand biased coin flips
        np.random.choice([0, 1], size=10000, p=[0.8, 0.2]).mean()

Out[9]: 0.1953
```