Lista #3

Isabella B. — 11810773

---

1. Seja a bola preta representada como uma vitória (i.e. a variável de interesse), temos:

   a. Para um sorteio de duas categorias de objetos sem reposição, precisamos considerar a alteração das quantidades entre cada novo sorteio, e sua distribuição (hipergeométrica) é dada por

   $$p(X,n) = \frac{\binom{4}{X}\binom{6}{6+4-X}}{\binom{6+4}{n}}$$

   b. Considerando a situação onde há reposição, temos uma variável binomial, dada por

   $$p(X) = \binom{6+4}{X}\left(\frac{4}{6+4}\right)^X\left(\frac{6}{6+4}\right)^{6+4-X},$$

   e que se mantém constante para qualquer n.

2. Para uma distribuição geométrica temos a fórmula

   $$p(X) = (1-p)^{X-1}p = 0.2^{X-1}\cdot 0.8$$

   de tal forma que:

   a.
   $$\sum_{X=4}^{\infty} 0.2^{X-1}\cdot 0.8$$

   o que é uma soma convergente, já que trata-se de uma série geométrica com razão $0.2 = r < 1$ e, portanto, pode ser somada usando

   $$\frac{p(4)}{(1-r)} = \frac{0.8\cdot 0.2^3}{0.8} = 10^{-3}\cdot 8 = 0.008$$

b.

$$
\begin{aligned}
&\sum_{X=4}^{7} 0.2^{X-1} \cdot 0.8 + \sum_{X=10}^{\infty} 0.2^{X-1} \cdot 0.8 \\
=&0.8 \cdot 0.2^3 + 0.8 \cdot 0.2^4 + 0.8 \cdot 0.2^5 + 0.8 \cdot 0.2^6 + \frac{0.8 \cdot 0.2^{10}}{0.8} \\
=&0.8 \cdot 0.2^3(1 + 0.2 + 0.2^2 + 0.2^3) + 0.2^{10} \\
=&(10^{-1} \cdot 8)(10^{-3} \cdot 8)(1 + 0.2 + 0.04 + 0.008) + 10^{-10} \cdot 2^{10} \\
=&1.248 \cdot 64 \cdot 10^{-4} + 2^{10} \cdot 10^{-10} \\
=&(1248 \cdot 64 + 1.024)10^{-7} \\
\approx&0.00799
\end{aligned}
$$

c.

$$
\begin{aligned}
&\sum_{X=3}^{5} 0.2^{X-1} \cdot 0.8 + \sum_{X=7}^{10} 0.2^{X-1} \cdot 0.8 \\
=&(0.8 \cdot 0.2^2 + 0.8 \cdot 0.2^3 + 0.8 \cdot 0.2^4) + (0.8 \cdot 0.2^6 + 0.8 \cdot 0.2^7 + 0.8 \cdot 0.2^8 + 0.8 \cdot 0.2^9) \\
=&0.8 \cdot 0.2^2(1 + 0.2 + 0.2^2 + 0.2^4 + 0.2^5 + 0.2^6 + 0.2^7) \\
=&2^5 \cdot 10^{-3}(1 + 0.2 + 0.04 + 0.0016 + 0.00032 + 0.000064 + 0.0000128) \\
=&0.032 \cdot (1.2419968) \\
\approx&0.0397
\end{aligned}
$$

Simulando resultados usando o script `ep3.py`, temos:

```
$ ./ep3.py 0.8 4 -
0.00798
$ ./ep3.py 0.8 4 - 7 10 -
0.00797
$ ./ep3.py 0.8 3 - 5 7 - 10
0.03969
```

O que nos dá uma margem de erro de $< 1\%$ para 10^6 iterações (hardcoded).