Tarefa Aula 8 - Métodos Computacionais da Física A

Aluno: Matheus Coelho Frank - Matrícula: 00221444
IF-UFRGS

17 de março de 2022

Resumo

Este relatório aborda o problema da distribuição dos valores resultantes das somas dos lançamentos de dois dados não viciados, ambos de seis lados, e seus respectivos dados estatísticos.

1 Introdução

O problema da soma dos valores de dois dados lançados simultaneamente, e as informações estatísticas obtidas a partir de n-lançamentos, à primeira vista, pode parecer de valor científico irrisório. Entretanto, há de se argumentar que assíduos jogadores de cassino, pricipalmente das modalidades que utilizam dados diretamente, (como na clássica Mesa de Dados) podem desfrutar de *insights* obtidos com este breve estudo.

Figura 1: Mesa de Dados.

2 Método

Afim de analisar o problema exposto, foram utilizados métodos computacionais para simular o lançamento dos dados, (por meio de um programa escrito em linguagem de programação Python) e as definições estatísticas de Valor Médio e Desvio Padrão.

Estabelecido isso, e utilizando-se de um gerador de números aleatórios de 1 à 6 - embutido no pacote Numpy do Python - para cada lançamento simultâneo, foi possível modelar a situação para quaisquer número n de lançamentos e coletar dados a partir de funções específicas, descritas a seguir:

Seja Y o resultado da soma dos lançamentos de dois dados X_1 e X_2, temos:

$$Y = X_1 + X_2 \tag{1}$$

A partir disso, comtemplado um ambiente de n-lançamentos, é possível calcular a o Valor Médio

$$Y_{médio} = \frac{1}{n}\sum_{i=1}^{n} Y_i \tag{2}$$

e o Desvio Padrão

$$\sigma_Y = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - Y_{médio})^2} \tag{3}$$

Posteriormente, estas expressões foram inseridas no código do programa.

Código utilizado em **python**:

```
import array
import numpy as np
import matplotlib.pyplot as plt
def roll_dices(n):
  d1=np.random.randint(1,7,n)
  d2=np.random.randint(1,7,n)
  y=d1+d2
  return y
def mean(y):
  sum=np.sum(y)
  y_mean=(1/n)*sum
```

```
  return y_mean
def std(y,y_mean):
  n=len(y)
  dev=np.sqrt((1/n)*np.sum((y_mean-y)**2))
  return dev
n=int(input('Insira o número de vezes que os dados serão jogados: \n'))
y=roll_dices(n)
y_mean=mean(y)
y_std=std(y,y_mean)

histograma = plt.figure()
plt.suptitle('Histograma de Lançamento de Dados')
plt.title(f'n = {n}')
plt.xlabel(f'Somas Possíveis')
plt.ylabel(f'Número de Ocorrências')
hist=np.zeros(11)
for i in y:
    hist[i-2]+=1
x = np.arange(2,13)
plt.bar(x,hist)
plt.savefig('histograma.png')
plt.show()

print(f'\n O valor médio obtido em {n} lançamentos é: {y_mean:.10f}')
print(f'\n O desvio padrão obtido em {n} lançamentos é: {y_std:.10f}')
print(f'\n o valor médio obtido de acordo com o módulo np.mean é:
{np.mean(y):.10f}')
print(f'\n o desvio padrão obtido de acordo com o módulo np.std é:
{np.std(y):.10f}')
print(f'\n O array y, contendo todos as somas dos dados 1 e 2, para
os lançamentos é:\n \n{y}')
```

3 Resultados

Os resultados obtidos para o Valor Médio e o Desvio Padrão em um espaço amostral de 1000 lançamentos, respectivamente, foram:

- Valor Médio:
$$Y_{médio} = 6.9250000000$$
- Desvio Padrão:
$$\sigma_Y = 2.4148240102$$

A distribuição dos valores obtidos em forma de histograma:

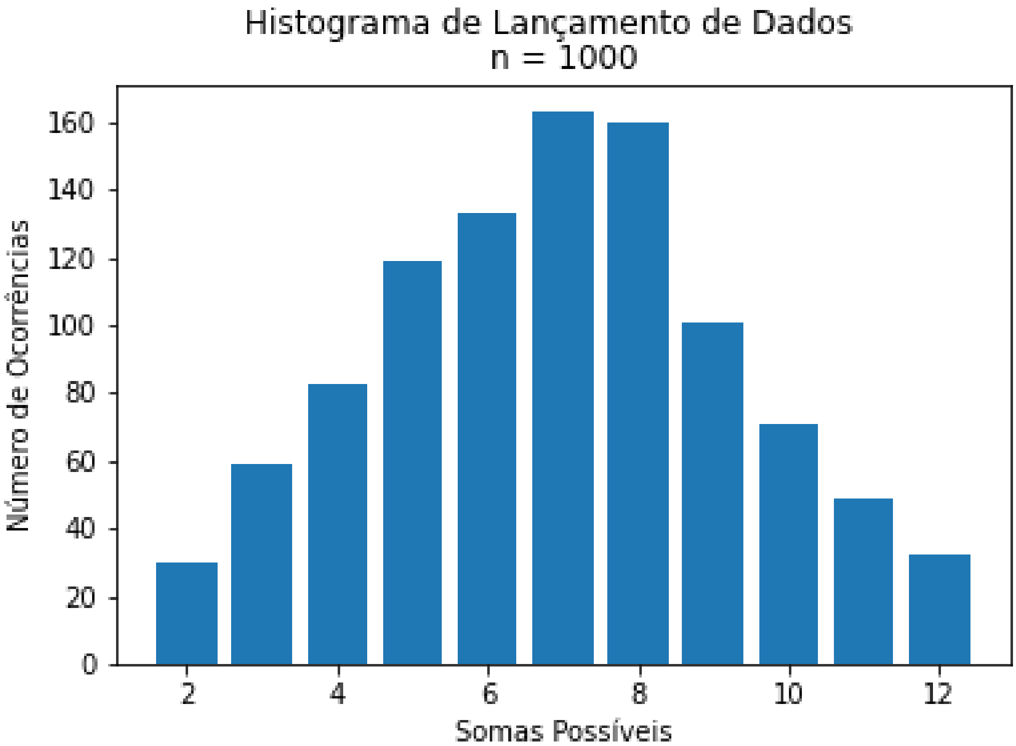

Figura 2: Histograma da Distribuição dos Lançamentos.

4 Conclusões

É possível observar que a soma com o maior número de ocorrências é 7. Isto é esperado porque -como os dados não são viciados- todas as faces tem a mesma probabilidade de caírem viradas para cima, logo, podemos escrever.

Probabilidade das Faces Caírem Viradas para Cima:

$$Face_1 = Face_2 = Face_3 = Face_4 = Face_5 = Face_6 = 1/6 \approx 16.7\%$$

Como estamos interessados na soma dos valores de dois lançamentos, podemos escrever 7 como:

$$1 + 6 = 7,\ 2 + 5 = 7,\ 3 + 4 = 7,\ 4 + 3 = 7,\ 5 + 2 = 7,\ 6 + 1 = 7,$$

Estas combinações somam 6 de um total de $6 * 6 = 36$ eventos possíveis.

$$\text{Logo, } 6/36 = 1/6 \approx 16.6\% \text{ de probabilidade que ocorram.}$$

Note que a probabilidade de uma soma ocorrer, portanto, diminui conforme diminuem as combinações possíveis dos dados.

$$2 + 6 = 8,\ 3 + 5 = 8,\ 4 + 4 = 8,\ 5 + 3 = 8,\ 6 + 2 = 8,$$

Totalizando 5 eventos possíveis de um total de 36 temos $5/36 \approx 13.8\%$ de probabilidade para que a soma 8 ocorra.

Podemos, então, finalmente concluir que a probabilidade de determinada soma Y é dada pela expressão:

$$Prob_Y = \left(\frac{6}{36} - \frac{|7 - Y|}{36}\right)\%, \text{ no intervalo } I = \{2, 12\}, \text{ com } Y \in \mathbb{N}. \tag{4}$$

Digno de nota também, é o fato que os resultados das expressões utilizadas fornecem pelo menos 10 casas decimais de precisão quando comparadas aos resultados das funções `numpy.mean` e `numpy.std`.

Referências

[1] Documentação Numpy em https://numpy.org/doc/

[2] Voices at: https://bit.ly/3wixroe