

Métodos Computacionais da Física A

Aluno: Leandra Wanderley Rodrigues Machado - Matrícula: 00302222
IF-UFRGS

23 de setembro de 2021

Resumo

Tarefa "Jogando dados" da aula 7 da disciplina de Métodos Computacionais da Física A

1 Introdução

O experimento "Jogando dados" consiste em simular a soma de 2 dados jogados N vezes e avaliar os resultados obtidos. Este experimento foi realizado com o intuito de se obter um conhecimento mais aprofundado das ferramentas de cálculo e simulação na linguagem Python e, ao mesmo tempo, ganhar um entendimento básico de algumas formas de análise estatística.

2 Método

Para a realização do experimento, foi elaborado um programa em Python que gera um array com o resultado da soma das jogadas, logo em seguida calcula a média dos resultados e seu desvio padrão.

A primeira e mais básica das funções elaboradas consiste em simular o resultado de 2 dados, gerando 2 número aleatórios N vezes e adicionar a sua soma a um array:

```
def gera_array(n):  
  
    lista = []  
  
    for i in range (n):  
        dado_1 = randint(1,6)  
        dado_2 = randint(1,6)  
        temp = dado_1 + dado_2  
  
        lista.append(temp)  
  
    return np.array(lista)
```

A segunda função realiza o cálculo da média dos valores contidos em um determinado array, seguindo a equação: $\langle Y \rangle = \frac{1}{N} \sum_{i=1}^N Y_i$

```
def gera_media( M ):  
  
    sum = 0  
  
    for i in range (len(M)):  
        sum += M[i]  
  
    return sum/(len(M))
```

A terceira função retorna o desvio padrão como sendo o resultado das operações descritas pela equação: $\sigma_Y = \sqrt{\frac{1}{N} \sum_{i=1}^N (Y_i - \langle Y \rangle)^2}$

```
def gera_desvio( N ):

    sum = 0
    mean = gera_media(N)

    for i in range (len(N)):
        sum += ( ( N[i] - mean)**2)
    sum /= (len(N))

    return (sum)**0.5
```

3 Resultados

Rodadndo o programa elaborado para este experimento, obtemos os seguintes resultados:

```
l = gera_array(1000)

med = gera_media(l)
print(med)

6.934

desv = gera_desvio(l)
print(desv)

2.451865412293259

np.mean(l)

6.934

np.std(l)

2.4518654122932606
```

Figura 1: Resultado das funções

Adicionando ao final do programa o comando a seguir, é possível obter um histograma do array gerado

```
plt.pyplot.hist(l,bins=[2,3,4,5,6,7,8,9,10,11,12,13],rwidth=0.6,align="left")
```

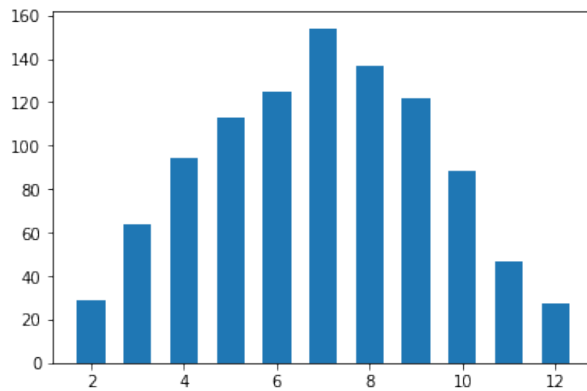


Figura 2: Histograma de Y

4 Conclusões

Ao observar os resultados obtidos, podemos concluir que o código elaborado teve exatamente o comportamento esperado. O cálculo da média em ambas as instâncias teve o mesmo resultado e o cálculo do desvio padrão diferiu apenas a partir da 14ª casa após a vírgula. Podemos concluir também que a simulação feita das 1000 jogadas de dados manteve as características esperadas, vendo-se claramente uma curva normal com pico na média $\langle Y \rangle$ e uma declividade em direção aos extremos respeitando o desvio σ_Y .