

Probabilidade e Processos Estocásticos

Lista de Exemplos – Unidade 02

2.1 – Tabela de Distribuição de Frequências

Questão 1

A tabela a seguir apresenta dados referentes às idades dos funcionários da empresa de engenharia fictícia. A média das idades é uma das informações que se deseja calcular com base no conjunto de dados da tabela 1.

```
import numpy as np
```

```
primitiva = np.array([[18, 38, 36, 56, 52, 38, 39, 56, 54, 20],  
                      [20, 30, 40, 32, 44, 44, 24, 28, 18, 30],  
                      [48, 48, 42, 46, 40, 19, 27, 19, 42, 37],  
                      [44, 34, 60, 18, 19, 36, 56, 26, 22, 28],  
                      [30, 44, 48, 54, 36, 34, 36, 34, 21, 19],  
                      [24, 52, 18, 28, 34, 60, 22, 28, 36, 26],  
                      [24, 40, 27, 55, 60, 38, 24, 44, 36, 34],  
                      [32, 34, 36, 44, 38, 40, 26, 48, 26, 28]  
                      ])
```

```
print(primitiva)
```

```
[[18 38 36 56 52 38 39 56 54 20]  
 [20 30 40 32 44 44 24 28 18 30]  
 [48 48 42 46 40 19 27 19 42 37]  
 [44 34 60 18 19 36 56 26 22 28]  
 [30 44 48 54 36 34 36 34 21 19]  
 [24 52 18 28 34 60 22 28 36 26]  
 [24 40 27 55 60 38 24 44 36 34]  
 [32 34 36 44 38 40 26 48 26 28]]
```

a) Considere a Tabela 1, monte o ROL:

```
sorted_array = np.sort(primitiva, axis= None)
rol = np.reshape(sorted_array, (-1, 10))
print(rol)
```

```
[[18 18 18 18 19 19 19 19 20 20]
 [21 22 22 24 24 24 24 26 26 26]
 [26 27 27 28 28 28 28 28 30 30]
 [30 32 32 34 34 34 34 34 34 36]
 [36 36 36 36 36 36 37 38 38 38]
 [38 39 40 40 40 40 42 42 44 44]
 [44 44 44 44 46 48 48 48 48 52]
 [52 54 54 55 56 56 56 60 60 60]]
```

b) Considere o ROL montado no item “a”. Montar a Tabela de Distribuição de Frequência por Valores (TDF por Valores):

```
unique, counts = np.unique(sorted_array, return_counts=True)
dist = np.asarray((unique, counts)).T
print(dist)
```

```
[[18  4]
 [19  4]
 [20  2]
 [21  1]
 [22  2]
 [24  4]
 [26  4]
 [27  2]
 [28  5]
 [30  3]
 [32  2]
 [34  6]
 [36  7]
 [37  1]
 [38  4]
 [39  1]
 [40  4]
 [42  2]
 [44  6]
 [46  1]
 [48  4]
 [52  2]
 [54  2]
 [55  1]
 [56  3]
 [60  3]]
```

2.2 – Elaboração da TDF

▼ Questão 2 – Considere o ROL (questão 1.a):

▼ a) Calcular a Amplitude Total (AT):

Solução:

$$AT = X_{max} - X_{min}$$

$$AT = 60 - 18$$

$$AT = 42$$

```
x_max = np.max(rol)
x_min = np.min(rol)
at = x_max - x_min
```

```
print(f"\n- X_max: {x_max}\n- X_min: {x_min}")
print(f"- Amplitude Total AT = {x_max} - {x_min} = {at}\n")
```

```
- X_max: 60
- X_min: 18
- Amplitude Total AT = 60 - 18 = 42
```

▼ b) Calcular o Número de Classes (K):

Regra de Sturges

$$K = \sqrt{N}$$

$$K = \sqrt{80}$$

$$K \approx 9$$

```
n = primitiva.shape[0]*primitiva.shape[1]
k = np.sqrt(n)
print(f"Número de Classes (K): {round(k)}")
```

Número de Classes (K): 9

▼ c) Calcular a Amplitude da Classe (h):

$$h \approx \frac{AT}{K}$$

$$Teste : h \times K \geq AT$$

$$h \approx \frac{42}{9} = 5$$

$$Teste : h \times K \geq AT = True$$

```
h = at / k
h = round(h)
print(f"Amplitude da Classe (h): {h}")
```

Amplitude da Classe (h): 5

Teste:

$h \times k \geq at$

True

▼ d) Montar a Tabela de Distribuição de Frequência por Classes (TDF por Classes) de acordo com os valores calculadas:

index	início	fim	frequência	Percentual %
1°	18	23	13	16.25%
2°	23	28	10	12.50%
3°	28	33	10	12.50%
4°	33	38	14	17.50%
5°	38	43	11	13.75%
6°	43	48	07	08.75%
7°	48	52	04	05.00%
8°	52	53	02	02.50%
9°	53	63	09	11.25%

init = 18

```
# Coluna de frequencia da tabela de frequencia
freq_t = np.array([13, 10, 10, 14, 11, 7, 4, 2, 9])
perc = np.zeros((1, 9), dtype="float32")

freq_t.sum()

# Coluna de Percentual da tabela de frequencia
for i, j in enumerate(freq_t):
    perc[0][i] = (100*j)/80

print("\nTabela de Distribuição de Frequências\n")
for i in range(9):
    print(init, end=" ")
    init += 5
    print(init, end="      ")
    print(freq_t[i], end="")
    print(f"          {perc[0][i]}%")
```

Tabela de Distribuição de Frequências

18 23	13	16.25%
23 28	10	12.5%
28 33	10	12.5%
33 38	14	17.5%
38 43	11	13.75%
43 48	7	8.75%
48 53	4	5.0%
53 58	2	2.5%
58 63	9	11.25%

```
# Coluna dos intervalos da tabela de frequencia
```

```
freq_tab = np.array([ [18, 23],
                      [23, 28],
                      [28, 33],
                      [33, 38],
                      [38, 43],
                      [43, 48],
                      [48, 53],
                      [53, 58],
                      [58, 63]])

print(freq_tab)
```

```
[[18 23]
 [23 28]
 [28 33]
 [33 38]
 [38 43]
 [43 48]
```

```
[48 53]
[53 58]
[58 63]]
```

2.3 – Elementos em uma TDF

Questão 3 – Considere o ROL (questão 1.a) e a TDF (questão 2.d):

- a) Definir as Classes (i) de dois intervalos distintos e o número total de classes (K).

Classes:

$$i_2 = 23|28$$

$$i_5 = 38|43$$

```
i_2 = freq_tab[2]
print(f"\n- Classe I2: {i_2}")
```

```
i_5 = freq_tab[5]
print(f"- Classe I5: {i_5}")
```

```
- Classe I2: [28 33]
- Classe I5: [43 48]
```

Total de Classes:

$$K = 9$$

```
print(f"\nTotal de classes k: {round(k)}\n")
```

```
Total de classes k: 9
```

- b) Definir as Limites de Classe de duas classes aleatórias.

Limites das Classes:

Classe i_4

$$l_4 = 38$$

$$L_4 = 43$$

Classe i_7

$$l_7 = 53$$

$$L_7 = 58$$

```
i_4 = freq_tab[4]
```

```
i_7 = freq_tab[7]
```

```
print(f"\n- Classe i_4: {i_4}\n- Limite Inferior da Classe - l_4: {i_4[0]}\n- Limite Superior da Classe - L_4: {i_4[1]}")
```

```
print(f"\n- Classe i_7: {i_7}\n- Limite Inferior da Classe - l_7: {i_7[0]}\n- Limite Superior da Classe - L_7: {i_7[1]}")
```

```
--NORMAL--
```

```
- Classe i_4: [38 43]
- Limite Inferior da Classe - l_4: 38
- Limite Superior da Classe L_4: 43
```

```
- Classe i_7: [53 58]
- Limite Inferior da Classe - l_7: 53
- Limite Superior da Classe L_7: 58
```

▼ c) Amplitude do Intervalo de Classe.

$$h = 5$$

```
hi = i_4[1] - i_4[0]
```

```
print(f"\nAmplitude do Intervalo de Classe h: {hi}\n")
```

```
Amplitude do Intervalo de Classe h: 5
```

▼ d) Amplitude Total da Distribuição.

$$h_d = L_{max} - l_{min}$$

$$h_d = 63 - 18$$

$$h_d = 45$$

```
l_min = freq_tab[:1,:1]
l_max = freq_tab[-1:,-1:]
hd = l_max - l_min

print(f"\nLimite Mínimo: {l_min[0][0]}")
print(f"Limite Máximo: {l_max[0][0]}")
print(f"Amplitude Total da Distribuição: {hd[0][0]}\n")
```

```
Limite Mínimo: 18
Limite Máximo: 63
Amplitude Total da Distribuição: 45
```

▼ e) Amplitude Total (Amplitude Total da Amostra).

$$AT = AA = X_{max} - X_{min}$$

```
X_max = np.max(rol)
X_min = np.min(rol)
AT = X_max - X_min

print(f"- X_max: {X_max},\n- X_min: {X_min}\n- Amplitude Total (AT): .

- X_max: 60,
- X_min: 18
- Amplitude Total (AT): 42
```

▼ f) Ponto Médio da Classe.

Classes	Ponto Médio das Classes
x1	20.5
x2	25.5
x3	30.5

Classes	Ponto Médio das Classes
x4	35.5
x5	40.5
x6	45.5
x7	50.5
x8	55.5
x9	60.5

```
xi = np.zeros((1, 9))

for j, i in enumerate(freq_tab):
    xi[0][j] = i.sum()/2
print(xi.T)

[[20.5]
 [25.5]
 [30.5]
 [35.5]
 [40.5]
 [45.5]
 [50.5]
 [55.5]
 [60.5]]
```

2.3.1 – Tipos de frequências

Questão 4

Considere TDF por Classes montada “questão 2”, elabore uma TDF por Classes, que contenha as seguintes frequências:

- Frequência Simples ou Absoluta (f_i)
- Frequência Relativa (f_{ri})
- Frequência Percentual ($f_i\%$)
- Frequência Acumulada (Simples (f_i), Relativa (f_{ri}), Percentual ($f_i\%$))

- Frequência Simples ou Absoluta (f_i)

```
freq_t

array([13, 10, 10, 14, 11, 7, 4, 2, 9])
```

- Frequência Relativa (f_{ri})

N = Soma de todas as frequências

$$f_{ri} = \frac{f_i}{N}$$

```
freq_relativa = freq_t/freq_t.sum()

print(f"\n- Frequência Relativa:\n")
for i in freq_relativa:
    print(i)

print(f"\nTotal Frequência Relativa: {freq_relativa.sum()}\n")
```

- Frequência Relativa:

```
0.1625
0.125
0.125
0.175
0.1375
0.0875
0.05
0.025
0.1125
```

Total Frequência Relativa: 1.0

▼ - Frequência Percentual ($f_i\%$)

$$f_i\% = f_{ri} \times 100$$

```
freq_percent = freq_relativa*100

print("\nFrequência Percentual:\n")
for i in freq_percent:
    print(f"{i:.2f}%")
print(f"\nTotal: {freq_percent.sum():.0f}%")
```

Frequência Percentual:

```
16.25%
12.50%
12.50%
```

17.50%
 13.75%
 8.75%
 5.00%
 2.50%
 11.25%

Total: 100%

▼ - Frequência Acumulada (Simples (f_i), Relativa (f_{ri}), Percentual ($f_i\%$)

Simples (f_i)

```
freq_simp_ac = np.zeros((1, 9))
```

```
for i in range(10):
    temp = 0
    for j in range(i):
        temp += freq_t[j]
    freq_simp_ac[0][i-1] = temp
```

```
print("\n- Frequência Simples Acumulada:")
for i in freq_simp_ac[0]:
    print(i)
```

```
- Frequência Simples Acumulada:
13.0
23.0
33.0
47.0
58.0
65.0
69.0
71.0
80.0
```

Relativa (f_{ri})

```
freq_relac_ac = np.zeros((1, 9))
```

```
for i in range(10):
    temp = 0
    for j in range(i):
        temp += freq_relativa[j]
    freq_relac_ac[0][i-1] = temp
```

```
print("\n- Frequência Relativa Acumulada:")
for i in freq_relac[0]:
    print(f"{i:.2f}")
```

```
- Frequência Relativa Acumulada:
0.16
0.29
0.41
0.59
0.72
0.81
0.86
0.89
1.00
```

Percentual ($f_i\%$)

```
freq_perc_ac = np.zeros((1, 9))
```

```
for i in range(10):
    temp = 0
    for j in range(i):
        temp += freq_percent[j]
    freq_perc_ac[0][i-1] = temp
```

```
print("\n\n- Frequência Percentual Acumulada:\n")
for i in freq_perc_ac[0]:
    print(f"{i:.2f}%")
```

```
- Frequência Percentual Acumulada:

16.25%
28.75%
41.25%
58.75%
72.50%
81.25%
86.25%
88.75%
100.00%
```

✓ 0s completed at 8:28 PM ● ✕