

# Distribuição de Frequência

**Disciplina: Estatística Aplicada a Engenharia de Software**

Prof. Me. Max Gabriel Steiner

# Introdução – Distribuição de Frequência

- Nesta etapa da disciplina vamos basicamente transformar uma variável numérica em variável categórica.
- Vamos trabalhar com a construção dos gráficos de histograma.
- Em seguida, vamos trabalhar com o estudo de caso voltado para a área de ciência de dados e machine learning que é a geração de regras de associação, onde vamos aprender o básico sobre o algoritmo Apriori, que é o principal algoritmo nesse cenário de regras de associação.
- A ideia é que vocês possam aprender como é feita essa integração entre a distribuição de frequência e das regras de associação, visando compreender algumas das aplicações utilizadas no cenário de aprendizagem de máquina.

# Distribuição de Frequência

- Vamos supor que vamos fazer uma determinada pesquisa, que exista uma população e que desta população vamos retirar uma amostra.
- Vamos supor que queremos medir a altura das pessoas, sendo que existe uma população de 100 pessoas e precisamos extrair uma amostra de 40 pessoas.
- Neste caso podemos também utilizar as técnicas que vimos no módulo de amostragem.

# Distribuição de Frequência

Tabela primitiva

160	165	167	164	160	166	160	161	150	152
173	160	155	164	168	162	161	168	163	156
155	169	151	170	164	155	152	163	160	155
157	156	158	158	161	154	161	156	172	153

- A tabela primitiva é composta pelas alturas das pessoas em centímetros. Nós temos 40 números e cada um desses números representa a altura de determinada pessoa.
- A tabela é chamada de primitiva pelo fato de que nós não fizemos nenhum processamento nessa tabela ainda. Por exemplo, podemos perceber que nem ordenada em crescente ou decrescente ela foi.

# Distribuição de Frequência

- Quando vamos trabalhar com distribuição de frequência, o primeiro passo a se fazer é o de ordenar a tabela.
- Neste caso teremos a tabela ordenada (também conhecida por rol).

Tabela ordenada (rol)

150	151	152	152	153	154	155	155	155	155
156	156	156	157	158	158	160	160	160	160
160	161	161	161	161	162	163	163	164	164
164	165	166	167	168	168	169	170	172	173

- Note que agora fica mais fácil observar os extremos dos dados (maior e menor pessoa).

# Distribuição de Frequência

- O objetivo agora é nós definirmos faixas de valores sempre que trabalhamos com distribuição de frequência. Nós vamos transformar essas alturas em uma faixa, por exemplo, entre 150 até 155, 155 até 160 e assim por diante.
- Tendo a tabela ordenada fica fácil saber os valores máximos e mínimos que vamos chamar de  $X_{\min}$  e  $X_{\max}$ .

Tabela ordenada (rol)

150	151	152	152	153	154	155	155	155	155
156	156	156	157	158	158	160	160	160	160
160	161	161	161	161	162	163	163	164	164
164	165	166	167	168	168	169	170	172	173

$X_{\min}$ : 150

$X_{\max}$ : 173

# Distribuição de Frequência

- Vamos agora aproveitar da tabela ordenada para fazer a distribuição de frequência.
- Notem que na primeira coluna vamos ter a estatura em cm.
- E a frequência que indica quantas vezes que essa altura se repete na base de dados.
- Por exemplo, 150 cm, temos somente 1 pessoa. 151 cm também somente uma pessoa. 152 cm temos 2 pessoas.
- Podemos perceber que essa tabela já nos fornece a distribuição das frequências dos valores, porém, esse formato de tabela acaba sendo muito grande, conforme pessoas com estaturas diferentes vão sendo acrescentadas.

Estatura (cm)	Frequência
150	1
151	1
152	2
153	1
154	1
155	4
156	3
157	1
158	2
160	5
161	4
162	1
163	2
164	3
165	1
166	1
167	1
168	2
169	1
170	1
172	1
173	1
Total	40

# Distribuição de Frequência

- Portanto, o ideal é que façamos a definição de faixas de valores para que a análise fique mais fácil.
- Podemos então transformar a tabela anterior nesta tabela de distribuição de frequências definindo intervalos:

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  --  173	3
Total	40



# Distribuição de Frequência

- Para a construção da tabela basta realizar a soma/contagem das frequências contidas em cada intervalo.
- Note que temos uma barrinha que separa a classe dos valores mínimos e máximo de cada intervalo.
- O valor que está do lado da barrinha entra na contagem e o que está do lado que não há a barrinha não está contido na contagem das frequências.
- **Perceba então que o valor final da classe não entra na contagem atual, apenas na próxima.**
- Por exemplo, o número de frequências do valor 154 está na segunda linha e não na primeira.

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  --  173	3
Total	40

# Distribuição de Frequência

- Vamos aplicar agora os cálculos passo a passo de como você deve definir os intervalos, sabemos que existem várias técnicas estatísticas para isto, mas vamos trabalhar com a técnica mais utilizada que é a aplicação da chamada Fórmula de Sturges que veremos na sequência.
- Antes disso é importante relembrar os conceitos de terminologia da distribuição de frequência.

# Distribuição de Frequência

- Portanto os conceitos da terminologia da distribuição de frequência:

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  --  173	3
Total	40

**Classe:** intervalos de variação da variável representados simbolicamente por  $i$

**Limite de classe**

Exemplo:  $l_1 = 150$  e  $L_2 = 158$

- No limite da classe no caso do exemplo  $l_1$ , o  $n^\circ 1$  indica de que classe estamos falando e o  $l$  por ser minúsculo indica que estamos falando do limite inferior.
- No caso do exemplo  $L_2$ , o  $n^\circ 2$  indica que se trata da classe 2, porém, como o  $L$  é maiúsculo trata-se do limite superior.

# Distribuição de Frequência

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  -- 173	3
Total	40

**Classe:** intervalos de variação da variável representados simbolicamente por  $i$

**Limite de classe**

Exemplo:  $l_1 = 150$  e  $L_2 = 158$

**Amplitude de um intervalo de classe ( $h_i$ )**

$h_i = L_i - l_i$  ( $154 - 150 = 4$ )

**Amplitude total da distribuição (AT)**

$AT = L_{(\max)} - L_{(\min)} = 173 - 150 = 23$

- O valor da amplitude total da distribuição (**AT**), indica que podemos ter um total de 23 alturas diferentes dentro dessa distribuição.

# Distribuição de Frequência

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  -- 173	3
Total	40

**Amplitude amostral (AA)**  $X_{\min}: 150$

$$AA = X_{(\max)} - X_{(\min)} = 173 - 150 = 23 \quad X_{\max}: 173$$

**Ponto médio de uma classe ( $x_i$ )**

$$X_i = (L_i + l_i) / 2 = (158 + 154) / 2 = 156 \text{ cm}$$

**Frequência**

$$f_2 = 9 \text{ (número de elementos na classe 2)}$$

- Atenção aqui, pois neste caso este cálculo será igual ao anterior, porém aqui estamos utilizando os valores de  $X_{\min}$  e  $X_{\max}$ , aqui na amplitude amostral (**AA**) vamos trabalhar com os valores da amostra (da tabela amostral) e não com os valores máximo e mínimo da classe da distribuição de frequência, uma vez que por exemplo, o nosso valor de altura máximo poderia ser 172, ou seja,  $X_{\max}$  seria 172, enquanto que o  $L_{\max}$  continuaria sendo 173, por conta dos intervalos das classes.

# Distribuição de Frequência

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  -- 173	3
Total	40

**Amplitude amostral (AA)**  $X_{\min}$ : 150

$AA = X_{(\max)} - X_{(\min)} = 173 - 150 = 23$   $X_{\max}$ : 173

**Ponto médio de uma classe ( $x_i$ )**

$X_i = (L_i + l_i) / 2 = (158 + 154) / 2 = 156 \text{ cm}$

**Frequência**

$f_2 = 9$  (número de elementos na classe 2)

- O ponto médio de uma classe apontado por  $X_i$  apresenta realmente o ponto médio da classe analisada.
- Por fim, a frequência é denotada por  $f$  minúsculo. O  $f_4$  por exemplo indica o número da frequência da classe 4.

# Distribuição de Frequência

- Agora que já compreendemos os conceitos da terminologia da distribuição de frequência, podemos então verificar como que serão definidos os intervalos de classe.
- Vamos utilizar então a Fórmula de Sturges para determinar o **número de classes** representado pela letra  $i$ . “ $n$ ” é a quantidade total de elementos  $n=40$ .

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  -- 173	3
Total	40

Determinar o número de classes

Fórmula de Sturges ( $i = 1 + 3.3 \log n$ )

$$1 + 3.3 * \log(40)$$

$$1 + 3.3 * 1.6 = 6.28$$

# Distribuição de Frequência

- Tendo o número de classes agora precisamos fazer o cálculo da amplitude, de quanto em quanto que vamos contar os valores, por exemplo 150 até 154, 154 até 158 e assim por diante.
- Vamos determinar a amplitude através da fórmula abaixo onde temos “h” como amplitude, temos AA que representa a amplitude amostral. Portanto:

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  -- 173	3
Total	40

Determinar a amplitude do intervalo de classe

$$h = AA / i \text{ (sempre arredondar para cima)}$$

$$23 / 6 = 3,83 \text{ (arredondado = 4)}$$

**Amplitude amostral (AA)**

$$AA = X_{(\max)} - X_{(\min)} = 173 - 150 = 23$$

- Por fim, precisamos fazer a contagem da **frequência** para cada classe.



# Distribuição de Frequência - Python

➤ Chegou a hora de fazermos a implementação passo a passo da distribuição de frequência manual, aplicando o passo a passo que vimos anteriormente, para então em seguida, utilizarmos as bibliotecas python.

➤ No google colab:

```
import numpy as np #para fazermos a manipulação de vetores e op matemática
```

```
import matplotlib.pyplot as plt #para gerarmos gráficos no python
```

```
import pandas as pd #para carregarmos uma base de dados e dataframes
```

```
import seaborn as sns #biblioteca gráfica
```

# Distribuição de Frequência - Python

- Precisamos criar o nosso dataset primitivo com os dados da tabela do slide:

Tabela primitiva

160	165	167	164	160	166	160	161	150	152
173	160	155	164	168	162	161	168	163	156
155	169	151	170	164	155	152	163	160	155
157	156	158	158	161	154	161	156	172	153

```
dados=np.array([160, 165, 167, 164, 160, 166, 160, 161, 150, 152, 173, 160, 155, 164,  
168, 162, 161, 168, 163, 156, 155, 169, 151, 170, 164, 155, 152, 163, 160, 155, 157,  
156, 158, 158, 161, 154, 161, 156, 172, 153])
```

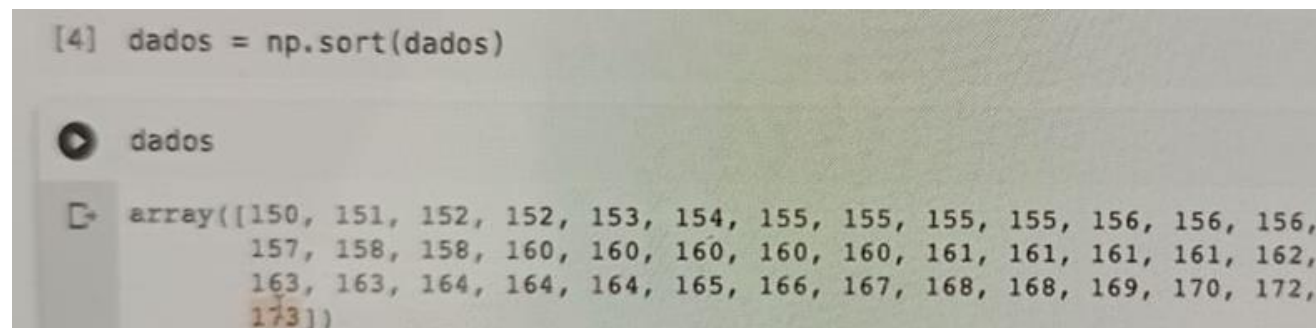
# Distribuição de Frequência - Python

- Precisamos agora definirmos o nosso rol, ou seja, os dados ordenados, precisamos ordenar em ordem os dados, portanto:

```
dados=np.sort(dados)
```

- Note que se solicitarmos a impressão da variável dados, eles já estão ordenados:

```
print(dados)
```



```
[4] dados = np.sort(dados)
```

dados

```
array([150, 151, 152, 152, 153, 154, 155, 155, 155, 155, 156, 156, 156, 157, 158, 158, 160, 160, 160, 160, 160, 161, 161, 161, 161, 162, 163, 163, 164, 164, 164, 165, 166, 167, 168, 168, 169, 170, 172, 173])
```

- Podemos agora buscar os valores mínimos e máximos, para isso:

```
minimo=dados.min()
```

```
print(minimo)
```

```
maximo=dados.max()
```

```
print(maximo)
```

# Distribuição de Frequência - Python

- Vamos agora visualizar a quantidade única de pessoas que temos em cada uma dessas alturas:

`np.unique(dados, return_counts=True)`

```
np.unique(dados, return_counts=True)

(array([150, 151, 152, 153, 154, 155, 156, 157, 158, 160, 161, 162, 163,
        164, 165, 166, 167, 168, 169, 170, 172, 173]),
 array([1, 1, 2, 1, 1, 4, 3, 1, 2, 5, 4, 1, 2, 3, 1, 1, 1, 2, 1, 1, 1, 1]))
```

- Podemos perceber que ele vai retornar as frequências, e esse array de dados vale como aquela primeira tabela de distribuição de frequências que vimos anteriormente:

Estatura (cm)	Frequência
150	1
151	1
152	2
153	1
154	1
155	4
156	3
157	1
158	2
160	5
161	4
162	1
163	2
164	3
165	1
166	1
167	1
168	2
169	1
170	1
172	1
173	1
Total	40

# Distribuição de Frequência - Python

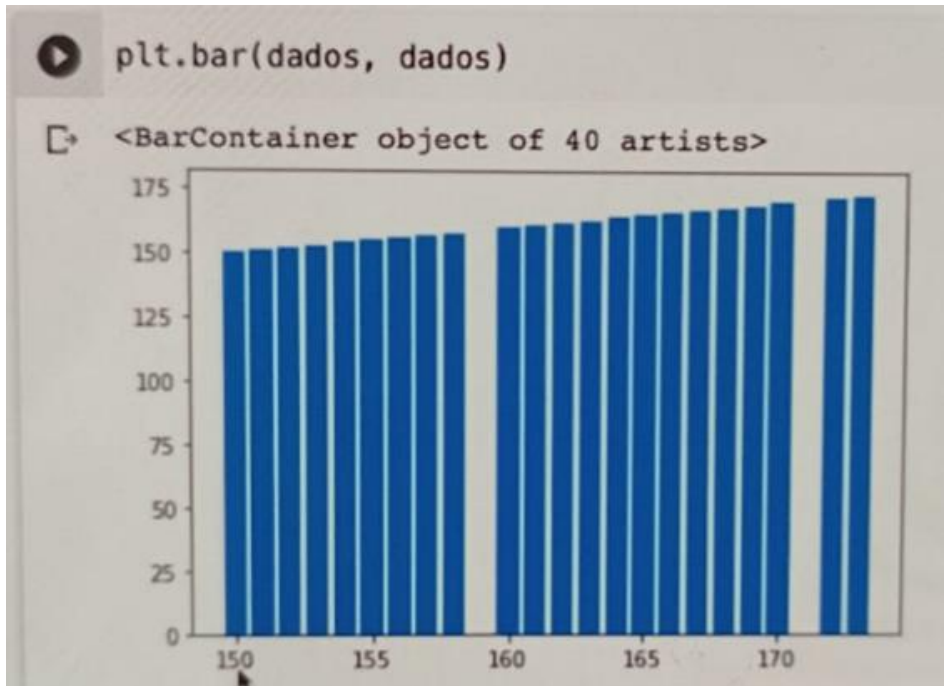
➤ Apenas lembrando que nosso objetivo é gerar essa tabela com as faixas:

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  --  173	3
Total	40

# Distribuição de Frequência - Python

- Podemos agora gerar um gráfico com os dados:

```
plt.bar(dados, dados)
```



- Percebemos que esse gráfico não mostra ainda uma visão clara da distribuição dos valores.

# Distribuição de Frequência - Python

- **Após a obtenção do rol precisamos agora definir o número de classes, para isso precisamos utilizar a Fórmula de Sturges**

Determinar o número de classes

Fórmula de Sturges ( $i = 1 + 3.3 \log n$ )

$$1 + 3.3 * \log(40)$$

$$1 + 3.3 * 1.6 = 6.28$$

```
n= len(dados)
```

```
print(n) #irá retornar n=40, que é a quantidade de pessoas que temos na base de dados.
```

- **Podemos agora aplicar a fórmula de Sturges de fato:**

```
i = 1 + 3.3*np.log10(n)
```

```
print(i) #i irá retornar exatamente o valor esperado de 6,28
```

# Distribuição de Frequência - Python

## ➤ Precisamos agora arredondar o valor de i

`i=round(i)`

`Print(i)` #i irá retornar o valor arredondado de 6 que indica exatamente o número de classes que queremos na nossa distribuição.

## ➤ Precisamos agora calcular a amplitude do intervalo, através da fórmula:

Determinar a amplitude do intervalo de classe

$h = AA / i$  (sempre arredondar para cima)

$23 / 6 = 3,83$  (arredondado = 4)

**Amplitude amostral (AA)**

$$AA = X_{(\max)} - X_{(\min)} = 173 - 150 = 23$$

## ➤ No Python:

`AA=maximo – minimo`

`print(AA)` #irá retornar o valor de 23, amplitude do intervalo

`h=AA/i`

`print(h)` #irá retornar o valor de 3,8333



# Distribuição de Frequência - Python

## ➤ Precisamos agora arredondar o valor de h

```
import math
```

```
h=math.ceil(h) #função utilizada para arredondar o número sempre para cima.
```

```
print(h) #percebemos que agora temos o valor de h sendo igual a 4 que indica a amplitude do nosso intervalo de classe.
```

## ➤ Agora podemos efetivamente construir a distribuição de frequência!

## ➤ É necessário definirmos os intervalos:

```
intervalos=np.arange(minimo, maximo, step=h) #importante perceber que a partição step vai indicar o tamanho do passo que será dado em cada intervalo, vamos utilizar o valor de h que se refere a nossa amplitude.
```

```
print(intervalos)
```

```
intervalos = np.arange(minimo, maximo, step = h)
intervalos
array([150, 154, 158, 162, 166, 170])
```


# Distribuição de Frequência - Python

## ➤ Observando a saída “intervalos”:

```
intervalos = np.arange(minimo, maximo, step = h)
intervalos
```

```
array([150, 154, 158, 162, 166, 170])
```

## ➤ Podemos perceber que ele finalizou em 170, porém precisamos que o nosso intervalo vá até 173:



Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  -- 173	3
Total	40

## ➤ Para isso:

`intervalos=np.arange(minimo, máximo + 2, step=h) #impprint(intervalos)`

```
intervalos = np.arange(minimo, maximo + 2, step = h)
intervalos
```

```
array([150, 154, 158, 162, 166, 170, 174])
```

# Distribuição de Frequência - Python

- Notemos que o valor final terá que ser 174, diferente do esperado 173:

```
intervalos = np.arange(minimo, maximo + 2, step = h)
intervalos
```

```
array([150, 154, 158, 162, 166, 170, 174])
```

- Contudo isso não é um problema, pois basta considerar que não teremos a barrinha na última classe, igual nas classes anteriores:

Estatura (cm)	Frequência
150  -- 154	5
154  -- 158	9
158  -- 162	11
162  -- 166	7
166  -- 170	5
170  -- 173	3
Total	40

# Distribuição de Frequência - Python

- **Nosso objetivo agora é de contar quantos elementos existem em cada intervalo de classe, portanto, vamos percorrer nossa base de dados inteira e vamos fazer um if, se o valor estiver entre 150 e 153, nós vamos fazer a contagem de cada uma das frequências.**

intervalo1, intervalo2, intervalo3, intervalo4, intervalo5, intervalo6 = 0, 0, 0, 0, 0, 0  
#codificação utilizada para que cada uma dessas variáveis inicie no valor de zero.

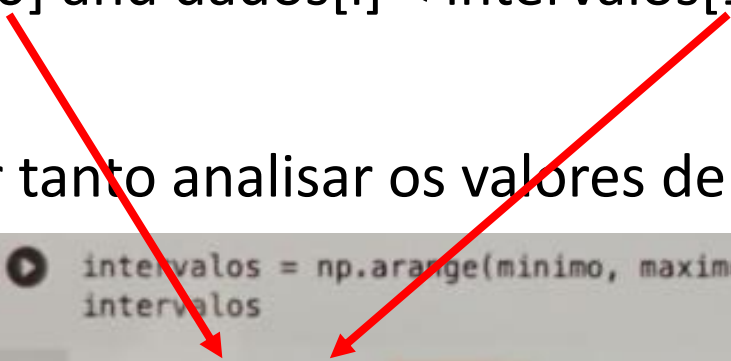
for i in range(n):

if dados[i] >= intervalos[0] and dados[i] < intervalos[1]:

intervalo1 +=1

#o código utilizado vai por tanto analisar os valores de 150 até 153

```
intervalos = np.arange(minimo, maximo, step = h)  
intervalos  
array([150, 154, 158, 162, 166, 170])
```



# Distribuição de Frequência - Python

## ➤ Continuando...

intervalo1, intervalo2, intervalo3, intervalo4, intervalo5, intervalo6 = 0, 0, 0, 0, 0, 0  
#codificação utilizada para que cada uma dessas variáveis inicie no valor de zero.

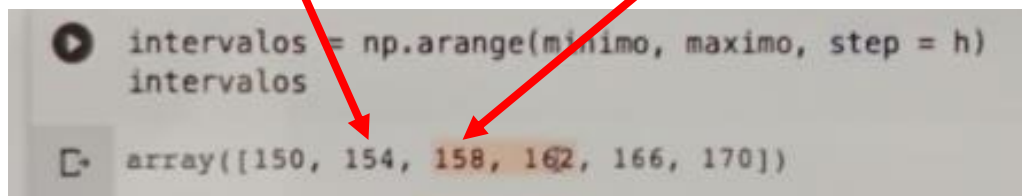
```
for i in range(n):
```

```
    if dados[i] >= intervalos[0] and dados[i] < intervalos[1]:
```

```
        intervalo1 +=1
```

```
    elif dados[i] >= intervalos[1] and dados[i] < intervalos[2]:
```

```
        intervalo2 +=1
```



```
intervalos = np.arange(minimo, maximo, step = h)  
intervalos  
array([150, 154, 158, 162, 166, 170])
```

# Distribuição de Frequência - Python

## ➤ Continuando...

intervalo1, intervalo2, intervalo3, intervalo4, intervalo5, intervalo6 = 0, 0, 0, 0, 0, 0 #codificação utilizada para que cada uma dessas variáveis inicie no valor de zero.

```
for i in range(n):
```

```
    if dados[i] >= intervalos[0] and dados[i] < intervalos[1]:
```

```
        intervalo1 +=1
```

```
    elif dados[i] >= intervalos[1] and dados[i] < intervalos[2]:
```

```
        intervalo2 +=1
```

```
    elif dados[i] >= intervalos[2] and dados[i] < intervalos[3]:
```

```
        intervalo3 +=1
```

```
    elif dados[i] >= intervalos[3] and dados[i] < intervalos[4]:
```

```
        intervalo4 +=1
```

```
    elif dados[i] >= intervalos[4] and dados[i] < intervalos[5]:
```

```
        intervalo5 +=1
```

```
    elif dados[i] >= intervalos[5] and dados[i] < intervalos[6]:
```

```
        intervalo6 +=1
```

# Distribuição de Frequência - Python

➤ **Agora vamos adicionar em uma lista cada um dos valores dos intervalos:**

```
lista_intervalos=[]  
lista_intervalos.append(intervalo1)  
lista_intervalos.append(intervalo2)  
lista_intervalos.append(intervalo3)  
lista_intervalos.append(intervalo4)  
lista_intervalos.append(intervalo5)  
lista_intervalos.append(intervalo6)  
print(lista_intervalos)
```

➤ **Podemos verificar que a saída da impressão nos fornece os mesmos valores de frequência da tabela:**

```
[5, 9, 11, 7, 5, 3]
```

# Distribuição de Frequência - Python

- **Vamos agora criar uma outra variável chamada de `lista_classes` para definirmos um string que fique mais fácil para identificar os valores das classes, exemplo de 150 até 154, 154 até 158, etc:**

```
lista_classes=[] #representa a variável vazia.
```

```
for i in range(len(lista_intervalos)): #assim vamos percorrer cada um dos intervalos que criamos no slide anterior.
```

```
    lista_classes.append(str(intervalos[i]) + '-' + str(intervalos[i+1]))
```

```
print(lista_classes)
```

```
lista_classes
```

```
['150-154', '154-158', '158-162', '162-166', '166-170', '170-174']
```

- **Podemos perceber que agora na variável `lista_classes` nós temos as classes e na variável `lista_intervalos` nós temos as quantidades/frequências.**





# Distribuição de Frequência - Python

- Por fim, agora sim podemos gerar um gráfico de barras, diferente do anteriormente gerado, agora sim teremos um gráfico que irá mostrar as frequências:

```
plt.bar(lista_classes, lista_intervalos) #sendo lista_classes eixo x do gráfico e lista_intervalos eixo y.
```

```
plt.title('Distribuição de frequência – histograma') #colocar um título no gráfico
```

```
plt.xlabel('intervalos') #representa o rótulo do eixo x
```

```
plt.ylabel('valores'); #representa o rótulo do eixo y, o ; do final é para ele não mostrar as mensagens do matplotlib
```

