

LISTA 12 - RESOLUÇÃO

March 2, 2022

ESTATÍSTICA APLICADA A COMPUTAÇÃO

LISTA 12

ALUNA: Maria Eduarda Pereira de Souza Melo

QUESTÃO1: O conjunto de dados dados1.csv possui notas de 100 alunos do curso de Economia da FEA-USP, em prova da disciplina Introdução à Probabilidade e à Estatística, 1999. Construa uma distribuição de frequências e um gráfico para essas notas.

```
[1]: import pandas as pd
import numpy as np
d1= pd.read_csv('/home/eduarda/Área de Trabalho/dados1.csv', sep=';',
               decimal=',', names=['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'])
dados1 = pd.DataFrame(d1)
dados1
```

```
[1]:
```

	A	B	C	D	E	F	G	H	I	J
0	3.5	4.0	5.5	6.0	5.0	5.5	5.0	5.5	4.0	10.0
1	6.5	9.5	4.0	7.0	7.5	3.0	4.5	5.0	2.5	6.0
2	5.0	6.5	3.5	4.5	8.5	4.0	8.0	7.0	6.0	7.5
3	8.5	6.0	9.0	6.0	6.5	7.5	5.5	6.5	8.0	8.5
4	4.5	7.5	8.0	3.0	4.0	8.0	4.5	5.5	6.0	6.0
5	7.5	3.5	3.0	7.0	1.5	4.5	10.0	5.5	2.5	10.0
6	4.0	6.5	7.5	5.5	7.0	7.5	6.0	6.5	6.5	5.5
7	6.5	5.0	5.5	7.5	8.0	6.5	5.0	7.0	6.0	5.5
8	3.0	5.0	3.5	6.0	6.5	6.0	8.0	5.5	7.5	6.0
9	2.5	7.5	9.0	6.0	6.5	3.5	4.5	7.0	5.0	5.0

```
[2]: hist, bin_edges = np.histogram(dados1, bins='auto')
x = np.around(bin_edges, decimals=2)
```

```
[3]: relativa =(hist/hist.sum())*100
cumulativa = (hist/hist.sum())*np.diff(bin_edges)
df = pd.DataFrame(list(zip(hist,relativa, cumulativa)), columns =
               ['Absoluta', 'Relativa %', 'cumulativa'])
df.index = [ '1.5 - 2.44', '2.44 - 3.39', '3.39 - 4.33', '4.33 - 5.28', '5.28 - 6.22',
            '6.22 - 7.17', '7.17 - 8.11', '8.11 - 9.06', '9.06 -10']
```

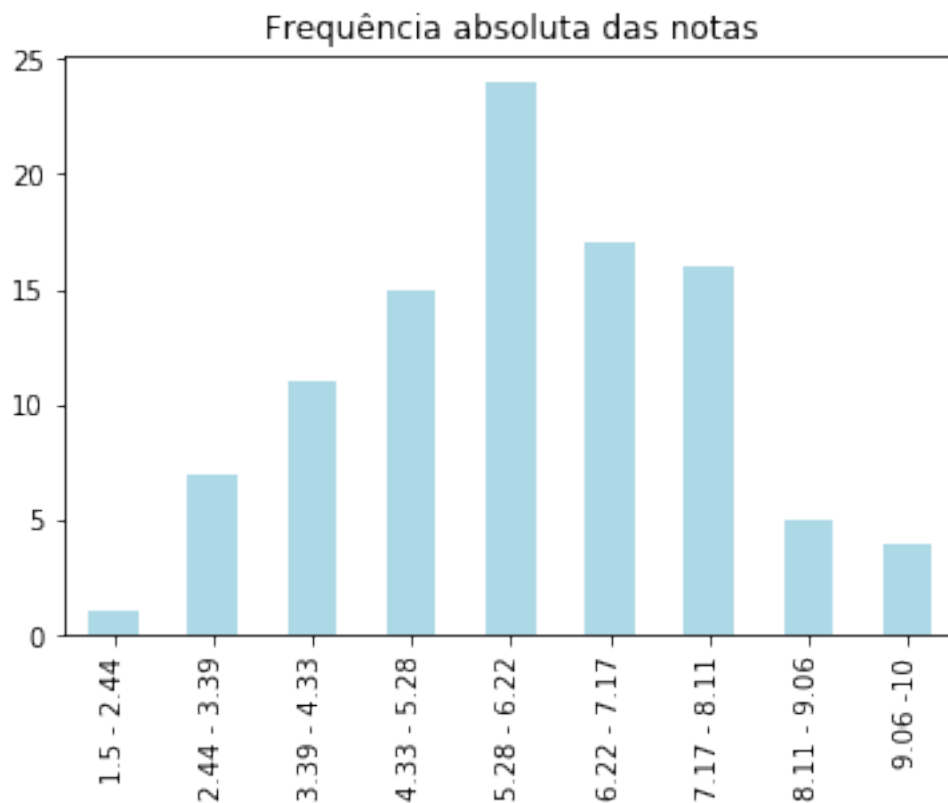
```
df
```

```
[3]:
```

	Absoluta	Relativa %	cumulativa
1.5 - 2.44	1	1.0	0.009444
2.44 - 3.39	7	7.0	0.066111
3.39 - 4.33	11	11.0	0.103889
4.33 - 5.28	15	15.0	0.141667
5.28 - 6.22	24	24.0	0.226667
6.22 - 7.17	17	17.0	0.160556
7.17 - 8.11	16	16.0	0.151111
8.11 - 9.06	5	5.0	0.047222
9.06 -10	4	4.0	0.037778

```
[4]: from matplotlib import pyplot as plt
%matplotlib inline
df['Absoluta'].plot.bar(title='Frequência absoluta das notas',color='#ADD8E6')
```

```
[4]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef2451220>
```



2. O conjunto de dados dados2.csv mostra os dados brutos de uma amostra de 27 imóveis anunciados para venda nos anúncios de um site especializado. Nesse exemplo,

cada observação é um imóvel e cada variável é um atributo dos imóveis (bairro, tipo, número de quartos, preço). Com base nesses dados: Importação dos dados:

```
[5]: d2= pd.read_csv('/home/eduarda/Área de Trabalho/dados2.csv',sep=';')
      dados2 = pd.DataFrame(d2)
      dados2
```

```
[5]:
```

	Bairro	Tipo	N de quartos	Preço (mil)
0	Barra	Apto.	2	165
1	Barra	Apto.	3	240
2	Barra	Cobt.	3	158
3	Barra	Sala	0	90
4	Botafogo	Apto.	2	270
5	Catete	Apto.	1	250
6	Centro	Sala	1	110
7	Copacabana	Apto.	2	210
8	Copacabana	Apto.	2	200
9	Copacabana	Apto.	3	280
10	Flamengo	Apto.	>=4	330
11	Flamengo	Cobt.	1	180
12	Gávea	Apto.	3	290
13	Jacarepaguá	Apto.	>=4	320
14	Ipanema	Apto.	3	276
15	Lagoa	Apto.	3	254
16	Laranjeiras	Apto.	2	220
17	Laranjeiras	Apto.	2	219
18	Leblon	Apto.	>=4	340
19	Leblon	Apto.	3	260
20	Maracanã	Apto.	>=4	338
21	Recreio	Cobt.	3	290
22	São Conrado	Casa	>=4	430
23	Tijuca	Apto.	2	200
24	Tijuca	Apto.	2	210
25	Tijuca	Casa	>=4	370
26	Vila Isabel	Apto.	2	230

(a) classifique cada variável do conjunto; Bairro->Variavel qualitativa . Tipo->Variavel qualitativa. Numero de quartos-> Variavel quantitativa. Preço-> Variavel quantitativa.

(b) monte uma distribuição de frequência para cada uma das variáveis;

0.0.1 BAIRRO:

```
[6]: absoluta=dados2['Bairro'].value_counts()
      relativa= dados2['Bairro'].value_counts(normalize=True)*100
      cumulativa=np.cumsum(dados2['Bairro'].value_counts())
      ddf = pd.DataFrame(list(zip(absoluta,relativa, cumulativa)), columns =
      ↳['Absoluta','Relativa %','cumulativa'])
```

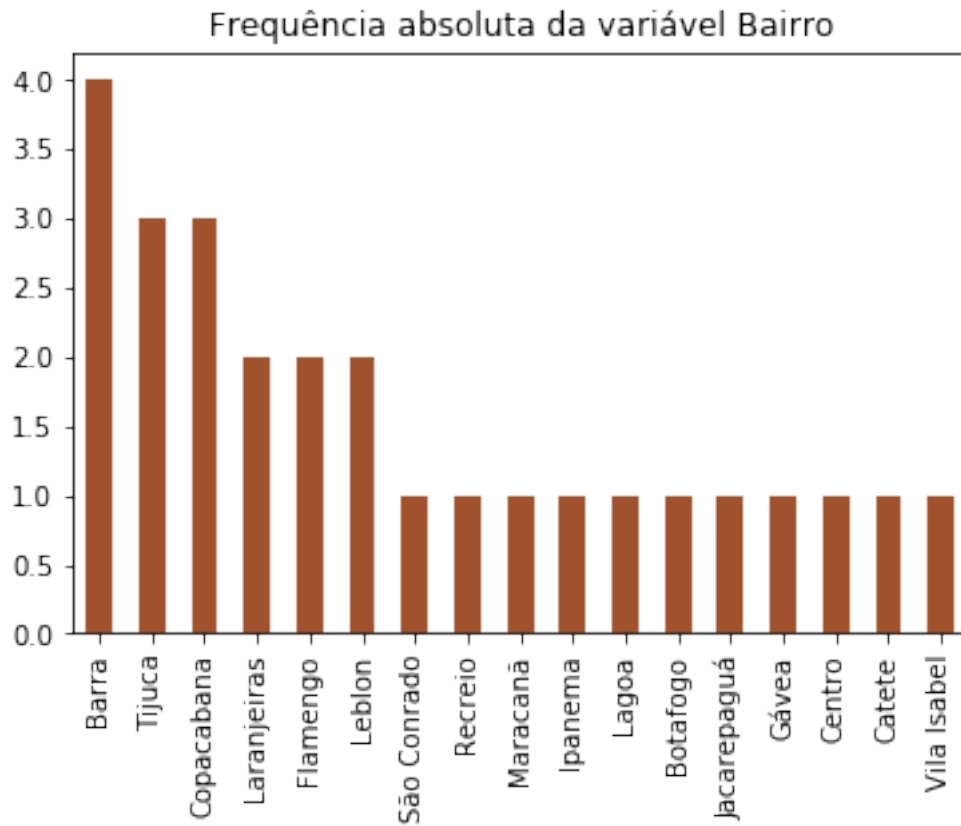
```
ddf.index = absoluta.index
ddf
```

```
[6]:
```

	Absoluta	Relativa %	cumulativa
Barra	4	14.814815	4
Tijuca	3	11.111111	7
Copacabana	3	11.111111	10
Laranjeiras	2	7.407407	12
Flamengo	2	7.407407	14
Leblon	2	7.407407	16
São Conrado	1	3.703704	17
Recreio	1	3.703704	18
Maracanã	1	3.703704	19
Ipanema	1	3.703704	20
Lagoa	1	3.703704	21
Botafogo	1	3.703704	22
Jacarepaguá	1	3.703704	23
Gávea	1	3.703704	24
Centro	1	3.703704	25
Catete	1	3.703704	26
Vila Isabel	1	3.703704	27

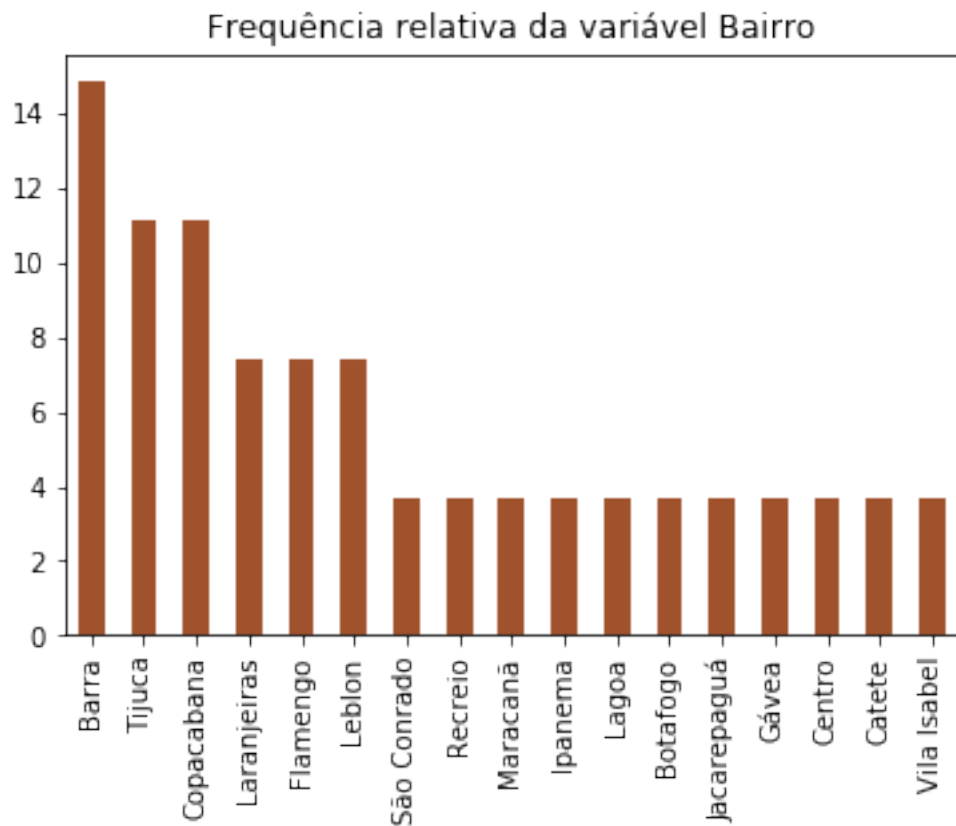
```
[7]: %matplotlib inline
absoluta.plot.bar(title='Frequência absoluta da variável Bairro',
→color='#A0522D')
```

```
[7]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef0397040>
```



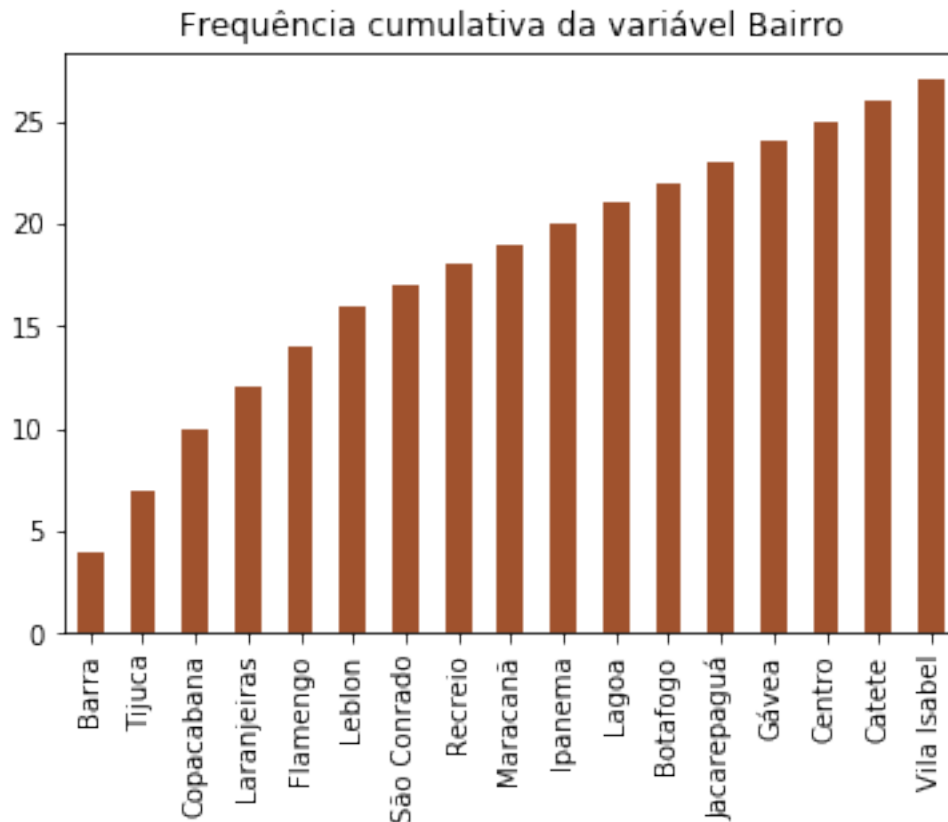
```
[8]: %matplotlib inline
      relativa.plot.bar(title='Frequência relativa da variável Bairro',
                        ↪color='#A0522D')
```

```
[8]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef02b6af0>
```



```
[9]: %matplotlib inline
      cumulativa.plot.bar(title='Frequência cumulativa da variável Bairro',
                           ↪color='#A0522D')
```

```
[9]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef0257df0>
```



0.0.2 TIPO:

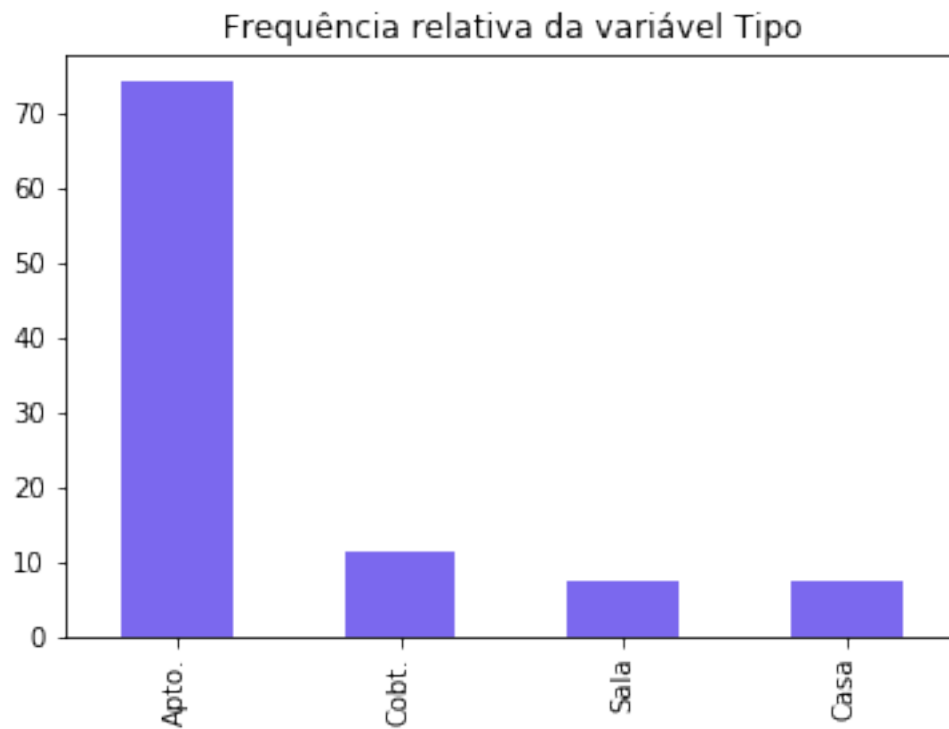
```
[10]: absoluta=dados2['Tipo'].value_counts()
relativa= dados2['Tipo'].value_counts(normalize=True)*100
cumulativa = np.cumsum(dados2['Tipo'].value_counts())
df = pd.DataFrame(list(zip(absoluta,relativa,cumulativa)), columns =_
    ↳['Absoluta','Relativa %','cumulativa'])
df.index = absoluta.index
df
```

```
[10]:
```

	Absoluta	Relativa %	cumulativa
Apto.	20	74.074074	20
Cobt.	3	11.111111	23
Sala	2	7.407407	25
Casa	2	7.407407	27

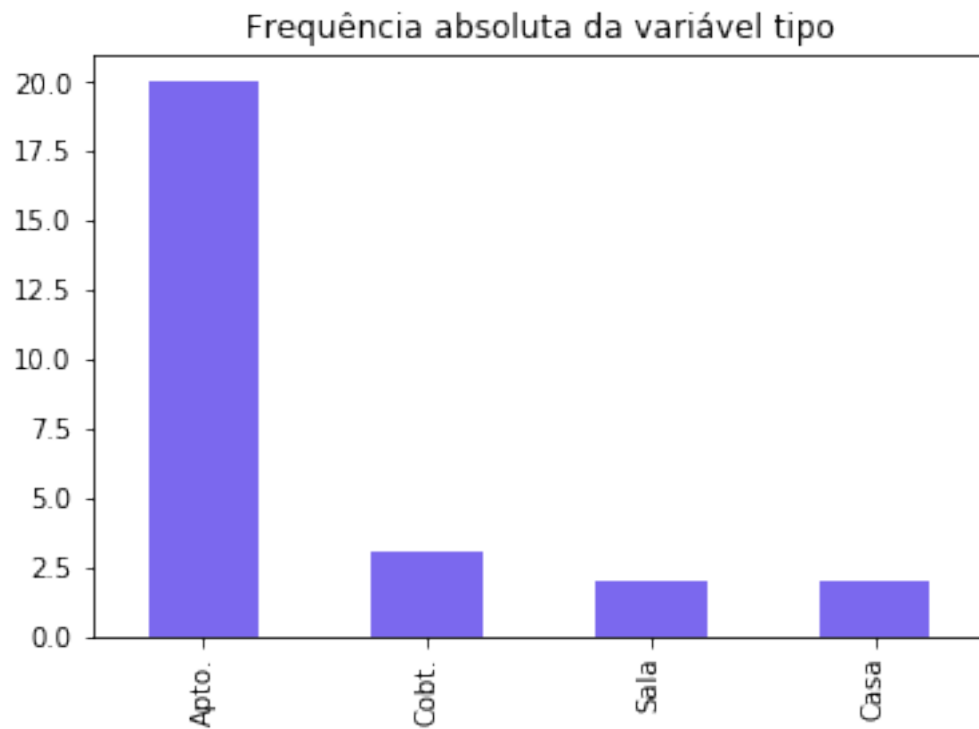
```
[11]: %matplotlib inline
relativa.plot.bar(title='Frequência relativa da variável Tipo',color='#7B68EE')
```

```
[11]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef016c2b0>
```



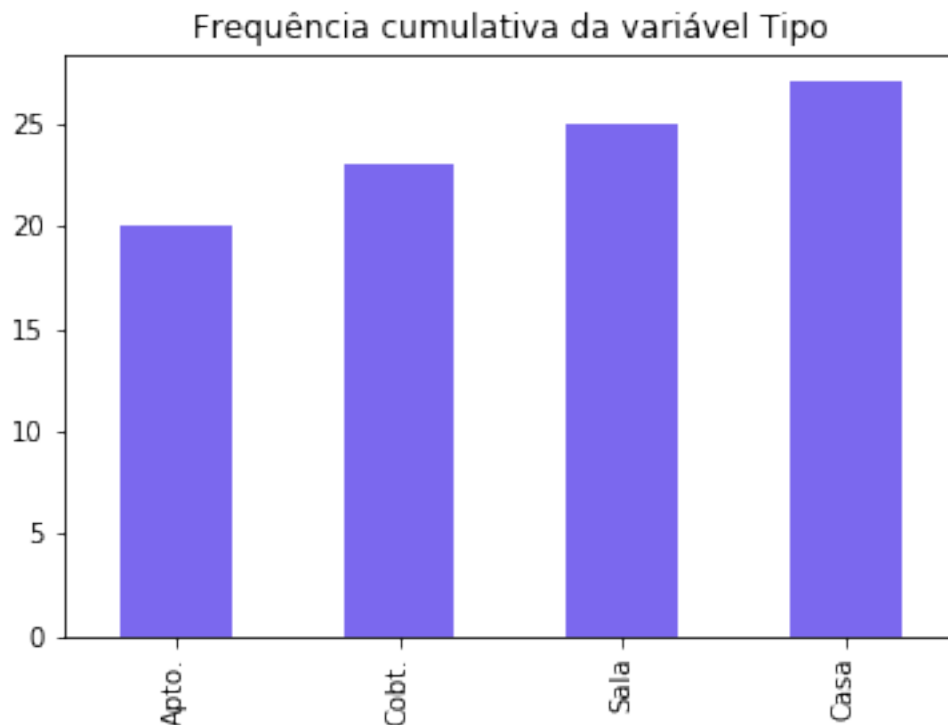
```
[12]: %matplotlib inline
absoluta.plot.bar(title='Frequência absoluta da variável tipo',color='#7B68EE')
```

```
[12]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef0146340>
```

```
[13]: %matplotlib inline
cumulativa.plot.bar(title='Frequência cumulativa da variável_
↳Tipo',color='#7B68EE')
```

```
[13]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef00b6160>
```



2.Distribuição de frequências para as variáveis quantitativas Numero de quartos e preço.

0.0.3 PREÇO:

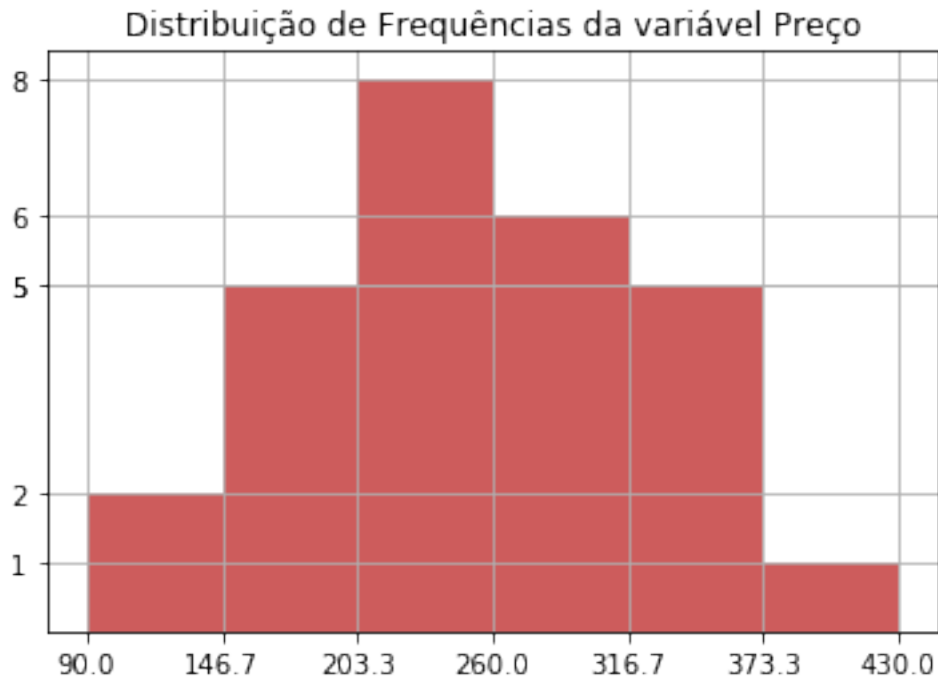
```
[14]: Distribuição_Preço= dados2['Preço (mil)']
hist, bin_edges = np.histogram(Distribuição_Preço, bins='auto')
relativa =(hist/hist.sum())*100
cumulativa = (hist/hist.sum())*np.diff(bin_edges)
df = pd.DataFrame(list(zip(hist,relativa, cumulativa)), columns =_
↳ ['Absoluta', 'Relativa %', 'cumulativa'])
df.index = ['90 - 146.67', '146.67 - 203.33', ' 203.33 - 260', '260 - 316.67',_
↳ '316.67 - 373.33', '373.33 - 430']
df
```

```
[14]:
```

	Absoluta	Relativa %	cumulativa
90 - 146.67	2	7.407407	4.197531
146.67 - 203.33	5	18.518519	10.493827
203.33 - 260	8	29.629630	16.790123
260 - 316.67	6	22.222222	12.592593
316.67 - 373.33	5	18.518519	10.493827
373.33 - 430	1	3.703704	2.098765

```
[15]: from matplotlib import pyplot as plt
      %matplotlib inline
      fi, classes,x= plt.hist(Distribuição_Preço, bins='auto', color='#CD5C5C')
      plt.xticks(classes)
      plt.yticks(fi)
      plt.grid()
      plt.title('Distribuição de Frequências da variável Preço')
```

```
[15]: Text(0.5, 1.0, 'Distribuição de Frequências da variável Preço')
```



0.0.4 NÚMERO DE QUARTOS:

```
[16]: absoluta=dados2['N de quartos'].value_counts()
      relativa= dados2['N de quartos'].value_counts(normalize=True)*100
      cumulativa = np.cumsum(dados2['N de quartos'].value_counts())
      df = pd.DataFrame(list(zip(absoluta,relativa,cumulativa)), columns = ['Absoluta', 'Relativa', 'cumulativa'])
      df.index = absoluta.index
      df
```

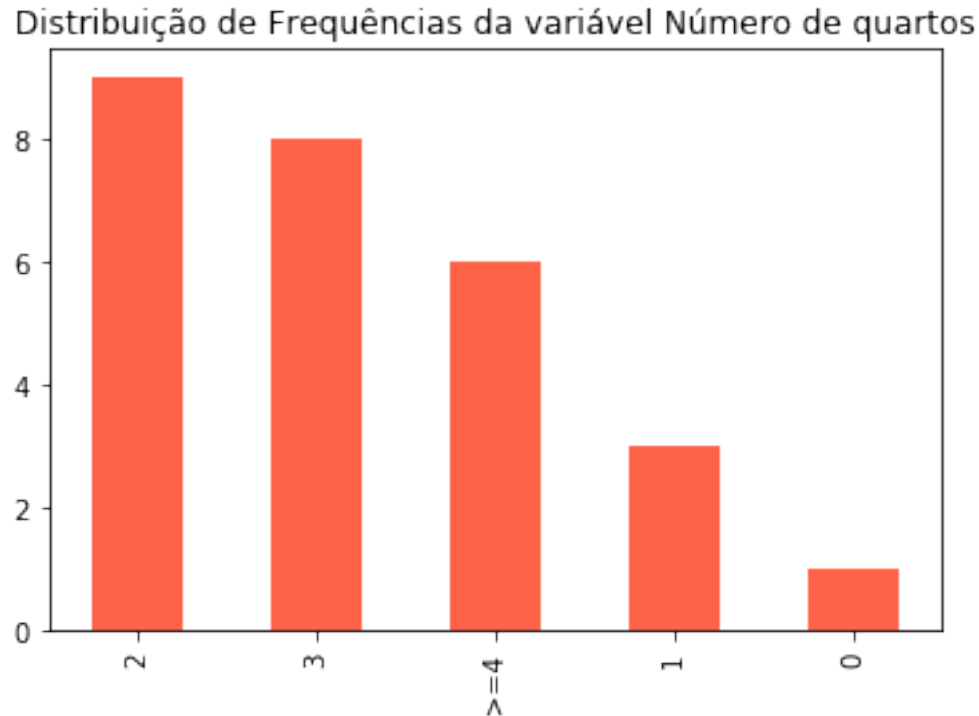
```
[16]:
```

	Absoluta	Relativa	cumulativa
2	9	33.333333	9
3	8	29.629630	17
>=4	6	22.222222	23

1	3	11.111111	26
0	1	3.703704	27

```
[17]: %matplotlib inline
absoluta.plot.bar(title = 'Distribuição de Frequências da variável Número de_
↳quartos', color = '#FF6347')
```

```
[17]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7ef0009ac0>
```



3. O conjunto de dados dados3.xls traz informações sobre um questionário aplicado aos alunos do primeiro ano do ensino médio de uma escola fornecendo informações sobre as seguintes variáveis: Turma: Qualitativa.

Gênero: Qualitativo.

Idade: Quantitativa.

Alt: Quantitativa.

Peso: Quantitativa.

Filhos: Quantitativa.

Fuma: Qualitativa.

Tolr: Qualitativa.

Exerc: Quantitativa.

Cine: Quantitativa.

OpCine: Qualitativa.

TV: Quantitativa.

OpTV: Qualitativa.

```
[18]: d3= pd.read_excel('/home/eduarda/Área de Trabalho/dados3.xls')
dados3 = pd.DataFrame(d3)
dados3
```

```
[18]:
```

	Turma	Sexo	Idade	Alt	Peso	Filhos	Fuma	Toler	Exerc	Cine	OpCine	TV	\
0	A	F	17	1.60	60.5	2	NAO	P	0	1	B	16	
1	A	F	18	1.69	55.0	1	NAO	M	0	1	B	7	
2	A	M	18	1.85	72.8	2	NAO	P	5	2	M	15	
3	A	M	25	1.85	80.9	2	NAO	P	5	2	B	20	
4	A	F	19	1.58	55.0	1	NAO	M	2	2	B	5	
5	A	M	19	1.76	60.0	3	NAO	M	2	1	B	2	
6	A	F	20	1.60	58.0	1	NAO	P	3	1	B	7	
7	A	F	18	1.64	47.0	1	SIM	I	2	2	M	10	
8	A	F	18	1.62	57.8	3	NAO	M	3	3	M	12	
9	A	F	17	1.64	58.0	2	NAO	M	2	2	M	10	
10	A	F	18	1.72	70.0	1	SIM	I	10	2	B	8	
11	A	F	18	1.66	54.0	3	NAO	M	0	2	B	0	
12	A	F	21	1.70	58.0	2	NAO	M	6	1	M	30	
13	A	M	19	1.78	68.5	1	SIM	I	5	1	M	2	
14	A	F	18	1.65	63.5	1	NAO	I	4	1	B	10	
15	A	F	19	1.63	47.4	3	NAO	P	0	1	B	18	
16	A	F	17	1.82	66.0	1	NAO	P	3	1	B	10	
17	A	M	18	1.80	85.2	2	NAO	P	3	4	B	10	
18	A	F	20	1.60	54.5	1	NAO	P	3	2	B	5	
19	A	F	18	1.68	52.5	3	NAO	M	7	2	B	14	
20	A	F	21	1.70	60.0	2	NAO	P	8	2	B	5	
21	A	F	18	1.65	58.5	1	NAO	M	0	3	B	5	
22	A	F	18	1.57	49.2	1	SIM	I	5	4	B	10	
23	A	F	20	1.55	48.0	1	SIM	I	0	1	M	28	
24	A	F	20	1.69	51.6	2	NAO	P	8	5	M	4	
25	A	F	19	1.54	57.0	2	NAO	I	6	2	B	5	
26	B	F	23	1.62	63.0	2	NAO	M	8	2	M	5	
27	B	F	18	1.62	52.0	1	NAO	P	1	1	M	10	
28	B	F	18	1.57	49.0	2	NAO	P	3	1	B	12	
29	B	F	25	1.65	59.0	4	NAO	M	1	2	M	2	
30	B	F	18	1.61	52.0	1	NAO	P	2	2	M	6	
31	B	M	17	1.71	73.0	1	NAO	P	1	1	B	20	
32	B	F	17	1.65	56.0	3	NAO	M	2	1	B	14	
33	B	F	17	1.67	58.0	1	NAO	M	4	2	B	10	

34	B	M	18	1.73	87.0	1	NAO	M	7	1	B	25
35	B	F	18	1.60	47.0	1	NAO	P	5	1	M	14
36	B	M	17	1.70	95.0	1	NAO	P	10	2	M	12
37	B	M	21	1.85	84.0	1	SIM	I	6	4	B	10
38	B	F	18	1.70	60.0	1	NAO	P	5	2	B	12
39	B	M	18	1.73	73.0	1	NAO	M	4	1	B	2
40	B	F	17	1.70	55.0	1	NAO	I	5	4	B	10
41	B	F	23	1.45	44.0	2	NAO	M	2	2	B	25
42	B	M	24	1.76	75.0	2	NAO	I	7	0	M	14
43	B	F	18	1.68	55.0	1	NAO	P	5	1	B	8
44	B	F	18	1.55	49.0	1	NAO	M	0	1	M	10
45	B	F	19	1.70	50.0	7	NAO	M	0	1	B	8
46	B	F	19	1.55	54.5	2	NAO	M	4	3	B	3
47	B	F	18	1.60	50.0	1	NAO	P	2	1	B	5
48	B	M	17	1.80	71.0	1	NAO	P	7	0	M	14
49	B	M	18	1.83	86.0	1	NAO	P	7	0	M	20

	OpTV
0	R
1	R
2	R
3	R
4	R
5	R
6	R
7	R
8	R
9	R
10	N
11	R
12	R
13	N
14	R
15	R
16	N
17	R
18	R
19	M
20	R
21	R
22	R
23	R
24	N
25	R
26	R
27	R
28	R

```

29    R
30    N
31    R
32    R
33    R
34    B
35    R
36    N
37    R
38    R
39    R
40    B
41    R
42    N
43    R
44    R
45    R
46    R
47    R
48    R
49    B

```

0.0.5 Alt

```

[19]: alt= dados3['Alt']
hist, bin_edges = np.histogram(alts, bins='auto')
relativa =(hist/hist.sum())*100
cumulativa = (hist/hist.sum())*np.diff(bin_edges)
df = pd.DataFrame(list(zip(hist,relativa, cumulativa)), columns =
    ['Absoluta','Relativa %','cumulativa'])
df.index = ['1.45 - 1.50714286', '1.50714286 -1.56428571', '1.56428571 - 1.
    62142857', '1.62142857 - 1.67857143', '1.67857143 - 1.73571429', '1.73571429
    - 1.79285714', '1.79285714 - 1.85']

df

```

```

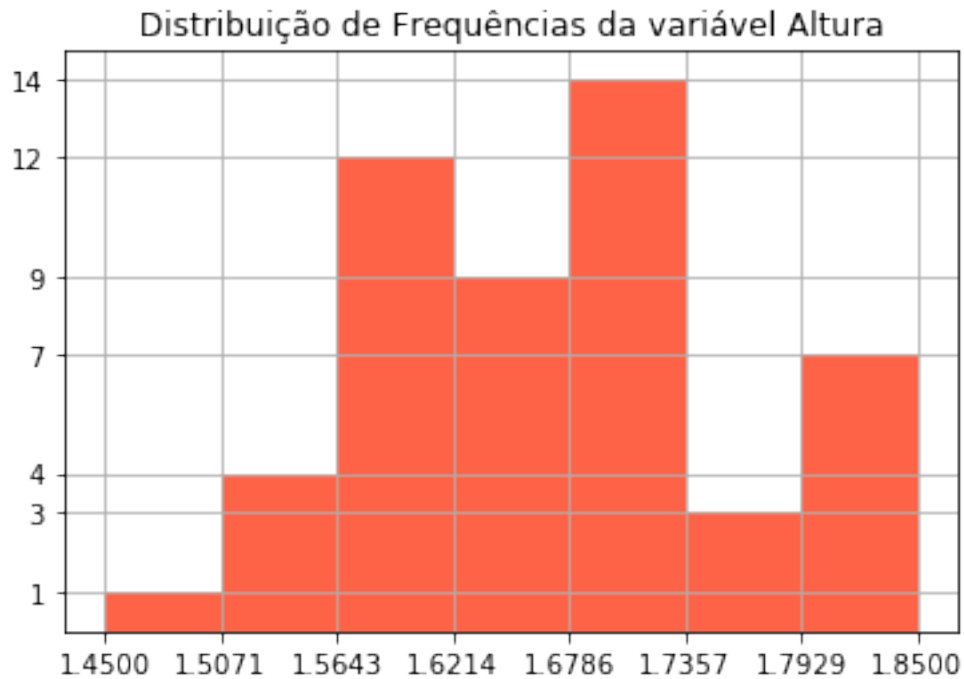
[19]:

```

	Absoluta	Relativa %	cumulativa
1.45 - 1.50714286	1	2.0	0.001143
1.50714286 -1.56428571	4	8.0	0.004571
1.56428571 - 1.62142857	12	24.0	0.013714
1.62142857 - 1.67857143	9	18.0	0.010286
1.67857143 - 1.73571429	14	28.0	0.016000
1.73571429 - 1.79285714	3	6.0	0.003429
1.79285714 - 1.85	7	14.0	0.008000

```
[20]: from matplotlib import pyplot as plt
      %matplotlib inline
      fi, classes,x= plt.hist(alt, bins='auto',color='#FF6347')
      plt.xticks(classes)
      plt.yticks(fi)
      plt.grid()
      plt.title('Distribuição de Frequências da variável Altura')
```

```
[20]: Text(0.5, 1.0, 'Distribuição de Frequências da variável Altura')
```



0.0.6 Peso

```
[21]: peso = dados3['Peso']
      hist, bin_edges = np.histogram(peso, bins='auto')
      relativa =(hist/hist.sum())*100
      cumulativa = (hist/hist.sum())*np.diff(bin_edges)
      df = pd.DataFrame(list(zip(hist,relativa, cumulativa)), columns =_
      ↪ ['Absoluta', 'Relativa %', 'cumulativa'])
      df.index=['44 - 51.29', '51.29 - 58.57', '58.57 - 65.86', '73.14 - 80.43', '80.43_
      ↪ - 87.71', '87.71 - 95', '95']
      df
```

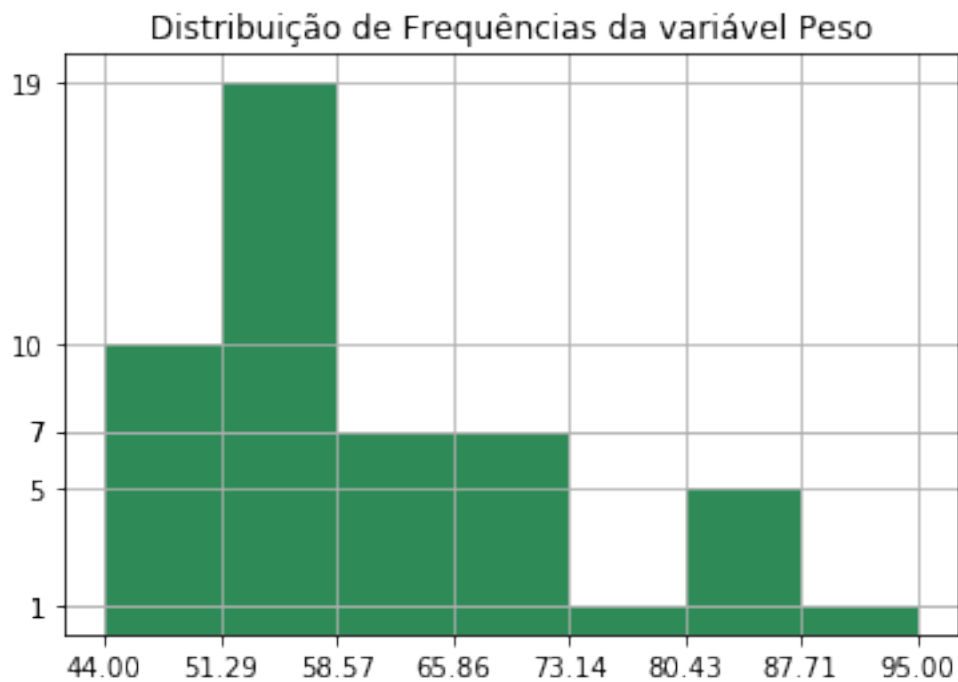
```
[21]:
```

	Absoluta	Relativa %	cumulativa
44 - 51.29	10	20.0	1.457143

51.29 - 58.57	19	38.0	2.768571
58.57 - 65.86	7	14.0	1.020000
73.14 - 80.43	7	14.0	1.020000
80.43 - 87.71	1	2.0	0.145714
87.71 - 95	5	10.0	0.728571
95	1	2.0	0.145714

```
[22]: %matplotlib inline
fi, classes,x= plt.hist(peso, bins='auto', color = '#2E8B57')
plt.xticks(classes)
plt.yticks(fi)
plt.grid()
plt.title('Distribuição de Frequências da variável Peso')
```

```
[22]: Text(0.5, 1.0, 'Distribuição de Frequências da variável Peso')
```



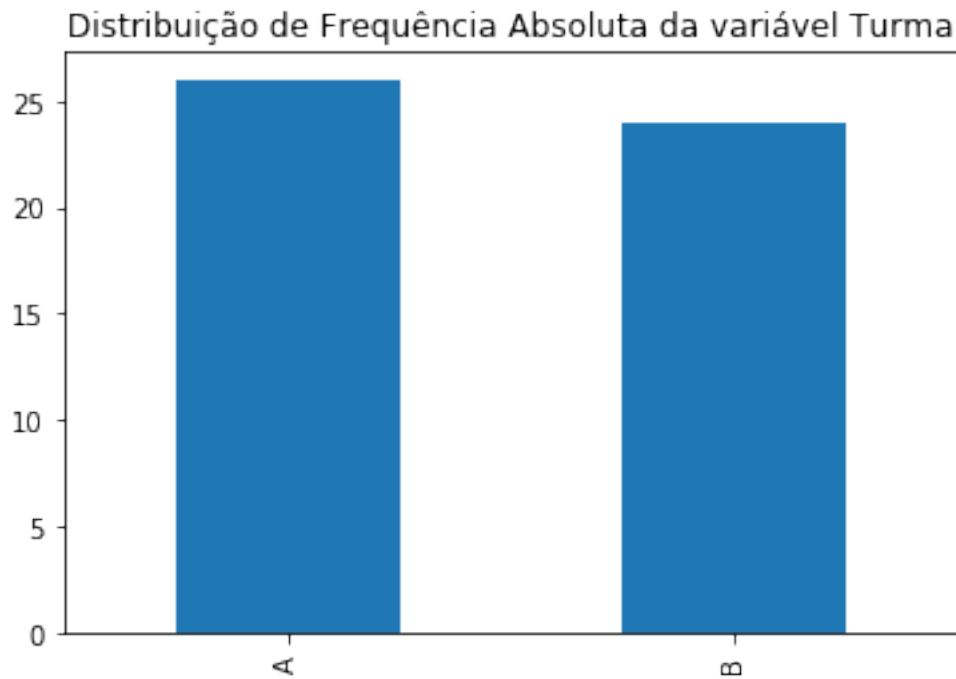
0.0.7 Turma

```
[23]: absoluta = dados3['Turma'].value_counts()
df = pd.DataFrame(absoluta)
df
```

```
[23]: Turma
A      26
```

```
[24]: %matplotlib inline
absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável_
↳Turma')
```

```
[24]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefe35100>
```



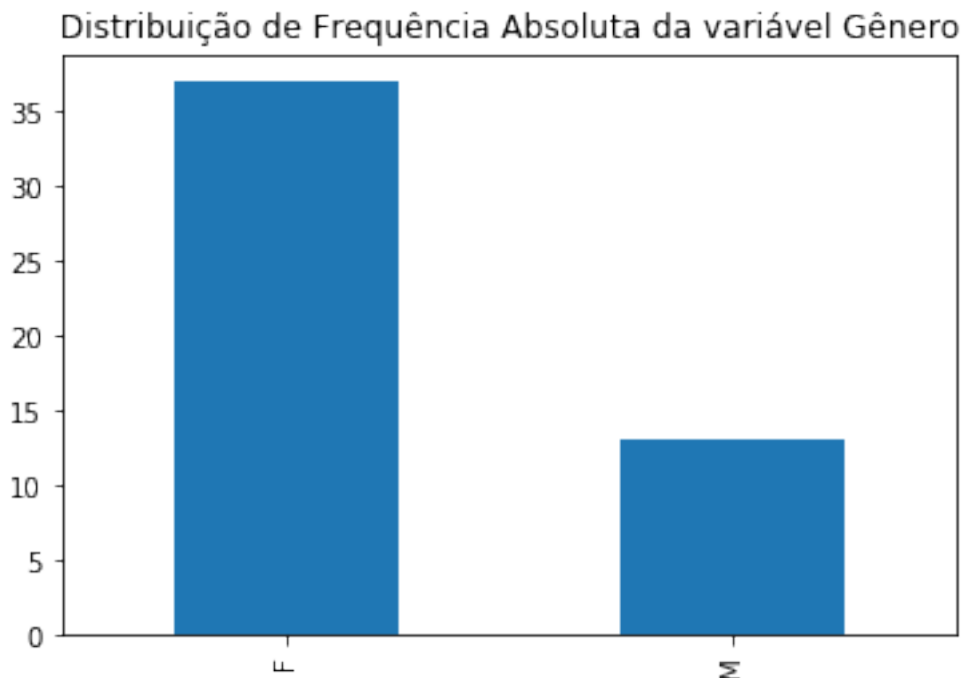
0.0.8 Gênero

```
[25]: absoluta = dados3['Sexo'].value_counts()
df = pd.DataFrame(absoluta)
df
```

```
[25]:      Sexo
F      37
M      13
```

```
[26]: %matplotlib inline
absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável_
↳Gênero')
```

```
[26]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefe04730>
```



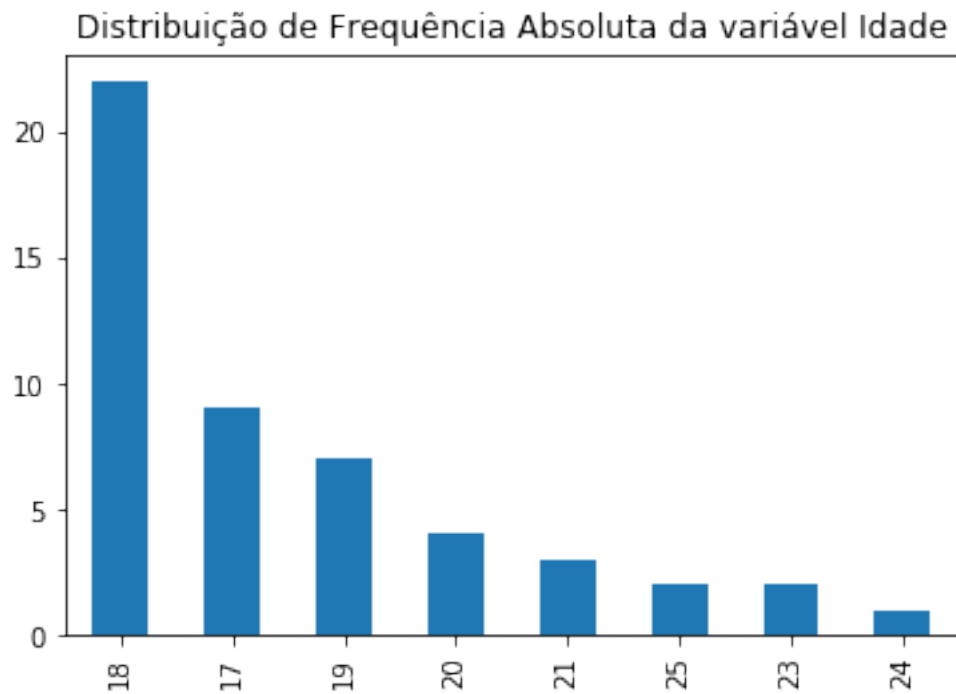
0.0.9 Idade

```
[27]: absoluta = dados3['Idade'].value_counts()
df = pd.DataFrame(absoluta)
df
```

```
[27]:      Idade
18      22
17       9
19       7
20       4
21       3
25       2
23       2
24       1
```

```
[28]: %matplotlib inline
absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável_
↳ Idade')
```

```
[28]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefdd50a0>
```



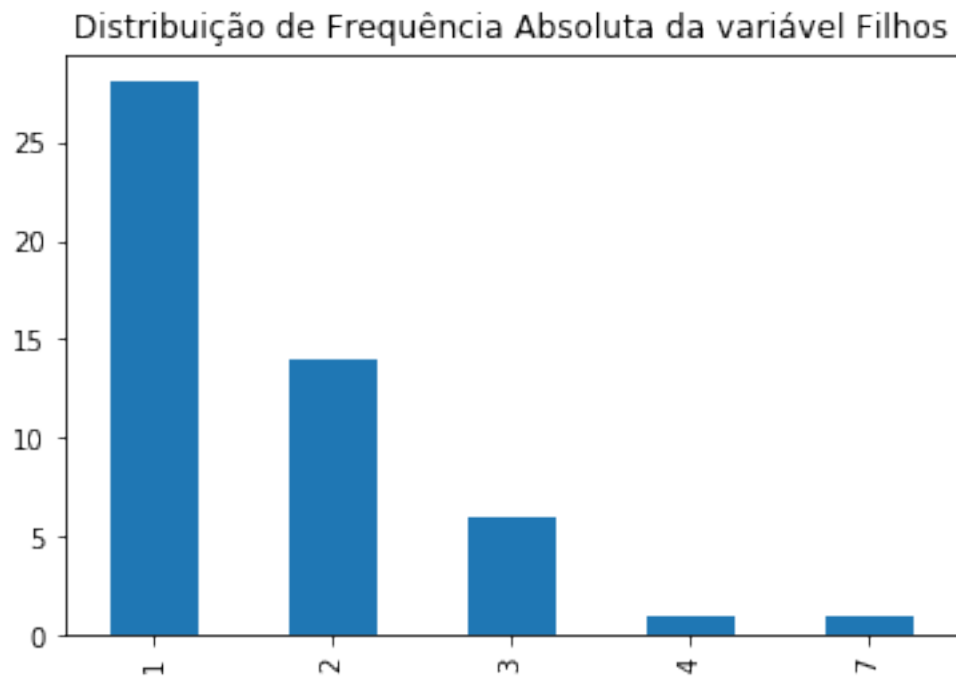
0.0.10 Filhos

```
[29]: absoluta = dados3['Filhos'].value_counts()
      df = pd.DataFrame(absoluta)
      df
```

```
[29]:      Filhos
      1      28
      2      14
      3       6
      4       1
      7       1
```

```
[30]: %matplotlib inline
      absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável_
      ↪Filhos')
```

```
[30]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefd48370>
```



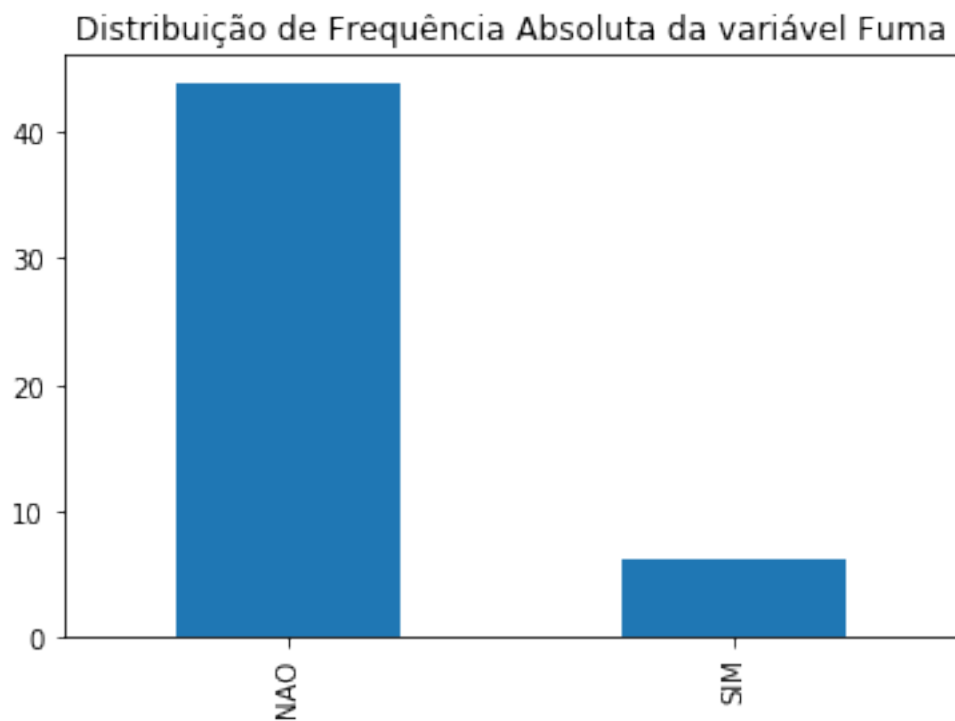
0.0.11 Fuma

```
[31]: absoluta = dados3['Fuma'].value_counts()  
df = pd.DataFrame(absoluta)  
df
```

```
[31]:      Fuma  
NAO    44  
SIM     6
```

```
[32]: %matplotlib inline  
absoluta.plot.bar(title='Distribuição de Frequência Absoluta da variável Fuma')
```

```
[32]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefeda8b0>
```



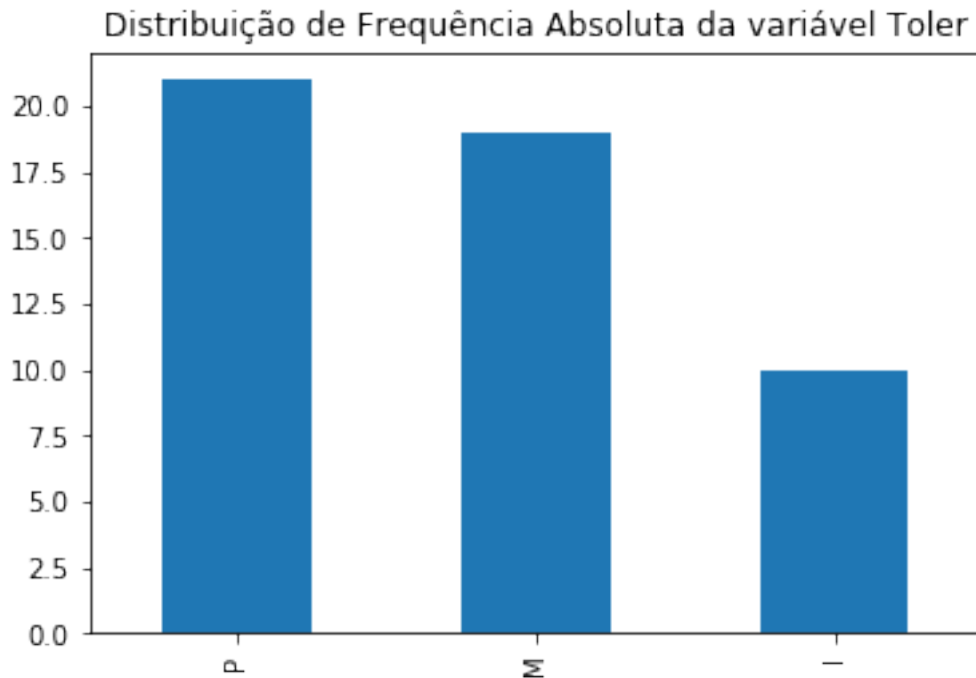
0.0.12 Toler

```
[33]: absoluta = dados3['Toler'].value_counts()
      df = pd.DataFrame(absoluta)
      df
```

```
[33]:   Toler
      P    21
      M    19
      I    10
```

```
[34]: %matplotlib inline
      absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável_
      ↳Toler')
```

```
[34]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefc65940>
```



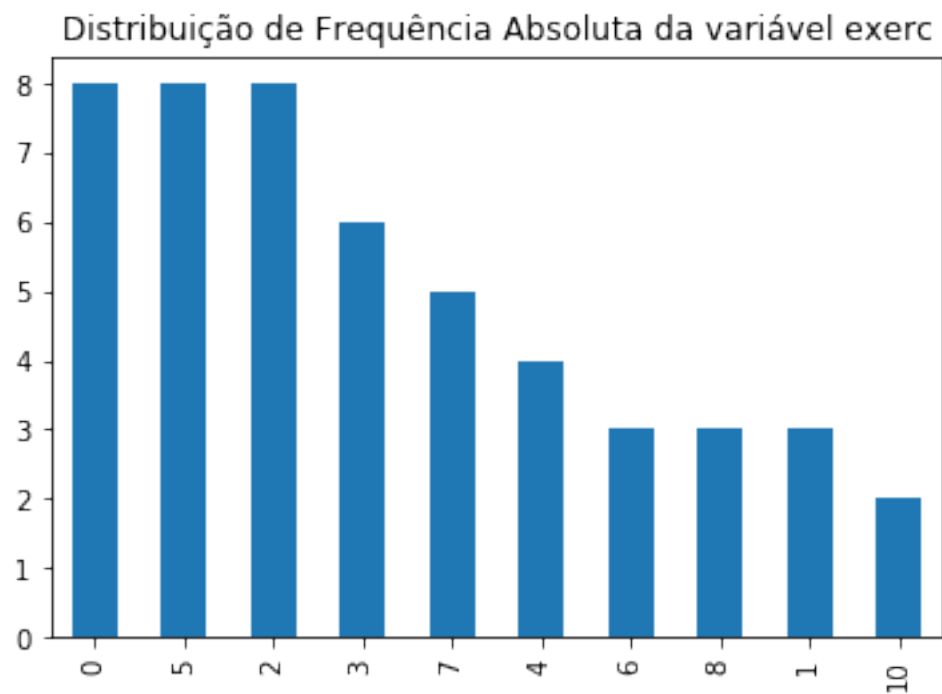
0.0.13 Exerc

```
[35]: absoluta = dados3['Exerc'].value_counts()
df = pd.DataFrame(absoluta)
df
```

```
[35]:      Exerc
0         8
5         8
2         8
3         6
7         5
4         4
6         3
8         3
1         3
10        2
```

```
[36]: %matplotlib inline
absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável_
↳exerc')
```

```
[36]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefc331c0>
```



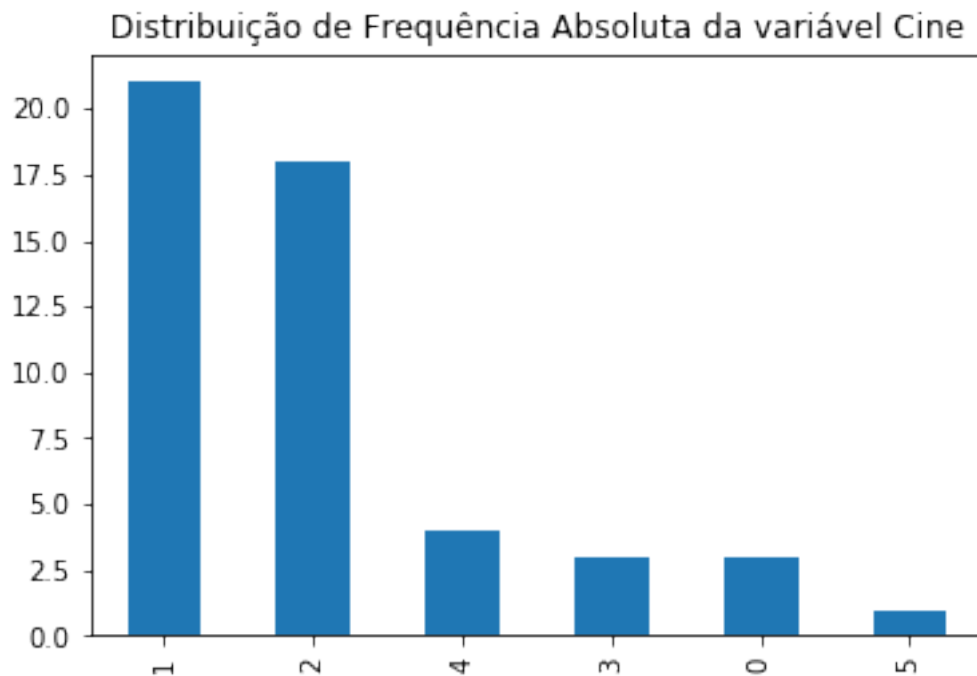
0.0.14 Cine

```
[37]: absoluta = dados3['Cine'].value_counts()  
df = pd.DataFrame(absoluta)  
df
```

```
[37]: Cine  
1    21  
2    18  
4     4  
3     3  
0     3  
5     1
```

```
[38]: %matplotlib inline  
absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável Cine')
```

```
[38]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefbb5af0>
```

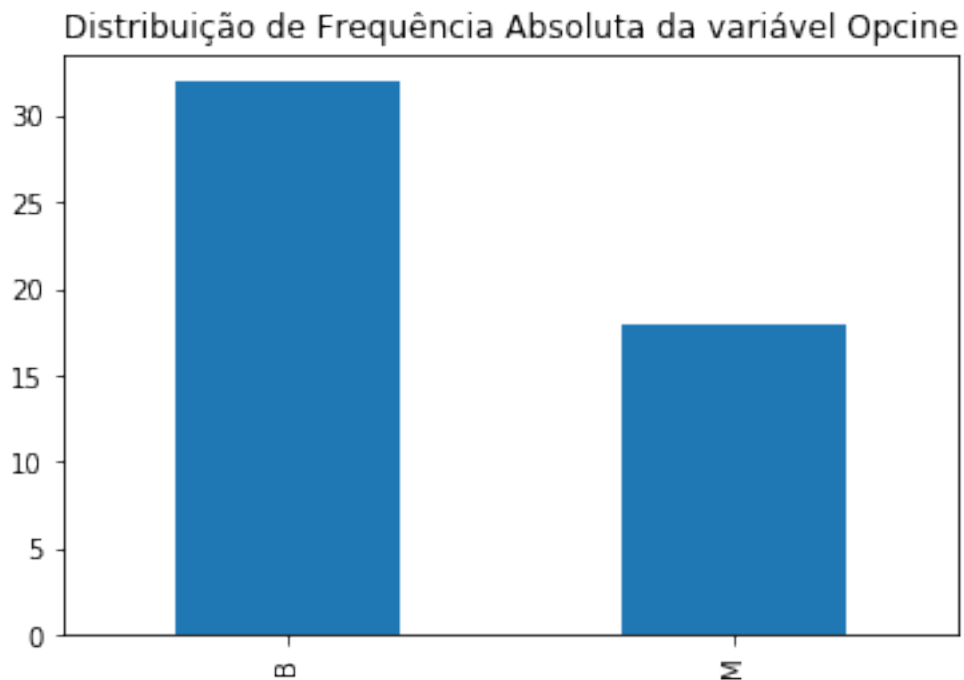
0.0.15 OpCine

```
[39]: absoluta = dados3['OpCine'].value_counts()
      df = pd.DataFrame(absoluta)
      df
```

```
[39]: OpCine
      B      32
      M      18
```

```
[40]: %matplotlib inline
      absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável OpCine')
```

```
[40]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefb21280>
```



0.0.16 TV

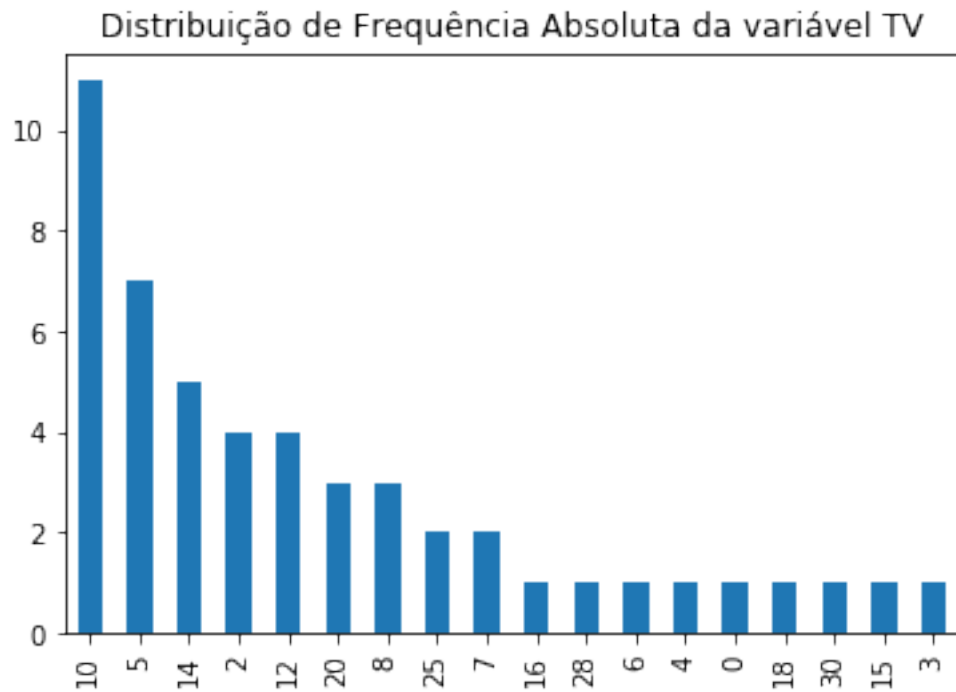
```
[41]: absoluta = dados3['TV'].value_counts()
      df = pd.DataFrame(absoluta)
      df
```

```
[41]: TV
      10  11
      5   7
     14   5
      2   4
     12   4
     20   3
      8   3
     25   2
      7   2
     16   1
     28   1
      6   1
      4   1
      0   1
     18   1
     30   1
     15   1
```

3 1

```
[42]: %matplotlib inline
absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável TV')
```

```
[42]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefaeaa90>
```



0.0.17 OpTV

```
[43]: absoluta = dados3['OpTV'].value_counts()
df = pd.DataFrame(absoluta)
df
```

```
[43]:   OpTV
R     39
N      7
B      3
M      1
```

```
[44]: %matplotlib inline
absoluta.plot.bar(title = 'Distribuição de Frequência Absoluta da variável OpTV')
```

```
[44]: <matplotlib.axes._subplots.AxesSubplot at 0x7f7eefa813d0>
```

