

Lista3aRespostas

December 3, 2021

ESTATÍSTICA APLICADA

LISTA 3a

ALUNO: Maria Eduarda Pereira de Souza Melo

1-

Lista: é uma estrutura de dados indexados e armazenados em sequência, onde cada elemento possui uma posição que é identificada por um índice.

Dicionário: é uma estrutura que compreendem um conjunto de pares: chave e valor. Cada chave individual possui um valor associado.

Array Numpy: é uma tabela multidimensional de elementos do mesmo tipo, indexados por uma tupla de inteiros positivos. As dimensões são chamadas de eixos (axes).

Séries Pandas: é um array unidimensional capaz de armazenar qualquer tipo de dados com rótulos ou índice de eixo. É a estrutura de dados para uma única coluna de um DataFrame.

2-

```
[1]: import pandas as pd
import numpy as np

dtt=pd.DataFrame(np.random.randint(1,35,size=(7, 5)),columns=list('VWXYZ'))
dtt
```

```
[1]:      V    W    X    Y    Z
0  22  22  26  33    3
1  34  33  14  17   21
2  34  19  31  25    6
3  30   2   6  22   12
4  15   1  27  12   29
5  29  28   5  20    9
6  33  32  16   8    4
```

3-

```
[2]: dados = pd.read_csv('https://raw.githubusercontent.com/selva86/datasets/master/
↳BostonHousing.csv')
dataframe= pd.DataFrame(dados)
print(dataframe[['crim', 'medv']])
```

```

      crim  medv
0    0.00632  24.0
1    0.02731  21.6
2    0.02729  34.7
3    0.03237  33.4
4    0.06905  36.2
..      ...   ...
501  0.06263  22.4
502  0.04527  20.6
503  0.06076  23.9
504  0.10959  22.0
505  0.04741  11.9

```

[506 rows x 2 columns]

4-

```

[3]: d = pd.read_csv('https://raw.githubusercontent.com/selva86/datasets/master/
↳Cars93_miss.csv')
df = pd.DataFrame(d)
df.rename(columns={'Type': 'CarType'}, inplace=True)
df.columns=df.columns.str.replace('.', '_')
df

```

<ipython-input-3-73b16c9e1659>:4: FutureWarning: The default value of regex will change from True to False in a future version. In addition, single character regular expressions will *not* be treated as literal strings when regex=True.

```
df.columns=df.columns.str.replace('.', '_')
```

```

[3]:  Manufacturer      Model  CarType  Min_Price  Price  Max_Price  MPG_city  \
0      Acura    Integra    Small      12.9    15.9      18.8      25.0
1      NaN     Legend  Midsize      29.2    33.9      38.7      18.0
2      Audi       90    Compact      25.9    29.1      32.3      20.0
3      Audi      100  Midsize      NaN    37.7      44.6      19.0
4      BMW      535i  Midsize      NaN    30.0      NaN      22.0
..      ...      ...      ...      ...      ...      ...      ...
88  Volkswagen  Eurovan    Van      16.6    19.7      22.7      17.0
89  Volkswagen  Passat    Compact      17.6    20.0      22.4      21.0
90  Volkswagen  Corrado  Sporty      22.9    23.3      23.7      18.0
91      Volvo     240    Compact      21.8    22.7      23.5      21.0
92      NaN      850  Midsize      24.8    26.7      28.5      20.0

```

```

      MPG_highway      AirBags  DriveTrain  ...  Passengers  Length  \
0      31.0      None      Front  ...      5.0    177.0
1      25.0  Driver & Passenger      Front  ...      5.0    195.0
2      26.0      Driver only      Front  ...      5.0    180.0
3      26.0  Driver & Passenger      NaN  ...      6.0    193.0
4      30.0      NaN      Rear  ...      4.0    186.0

```

```

..      ...      ...      ...      ...      ...
88      21.0      None      Front      ...      7.0      187.0
89      30.0      None      Front      ...      5.0      180.0
90      25.0      None      Front      ...      4.0      159.0
91      28.0      Driver only      Rear      ...      5.0      190.0
92      28.0      Driver & Passenger      Front      ...      5.0      184.0

Wheelbase  Width  Turn_circle  Rear_seat_room  Luggage_room  Weight  \
0      102.0  68.0      37.0      26.5      NaN      2705.0
1      115.0  71.0      38.0      30.0      15.0      3560.0
2      102.0  67.0      37.0      28.0      14.0      3375.0
3      106.0  NaN      37.0      31.0      17.0      3405.0
4      109.0  69.0      39.0      27.0      13.0      3640.0
..      ...      ...      ...      ...      ...
88      115.0  72.0      38.0      34.0      NaN      3960.0
89      103.0  67.0      35.0      31.5      14.0      2985.0
90      97.0  66.0      36.0      26.0      15.0      2810.0
91      104.0  67.0      37.0      29.5      14.0      2985.0
92      105.0  69.0      38.0      30.0      15.0      3245.0

Origin      Make
0  non-USA      Acura Integra
1  non-USA      Acura Legend
2  non-USA      Audi 90
3  non-USA      Audi 100
4  non-USA      BMW 535i
..      ...
88      NaN      Volkswagen Eurovan
89  non-USA      Volkswagen Passat
90  non-USA      Volkswagen Corrado
91  non-USA      Volvo 240
92  non-USA      Volvo 850

```

[93 rows x 27 columns]

5-

```
[4]: pd.isnull(df)
```

```

[4]:  Manufacturer  Model  CarType  Min_Price  Price  Max_Price  MPG_city  \
0      False  False  False      False  False      False      False
1      True  False  False      False  False      False      False
2      False  False  False      False  False      False      False
3      False  False  False      True  False      False      False
4      False  False  False      True  False      True      False
..      ...      ...      ...      ...      ...      ...
88      False  False  False      False  False      False      False

```

89	False	False	False	False	False	False	False
90	False	False	False	False	False	False	False
91	False	False	False	False	False	False	False
92	True	False	False	False	False	False	False

	MPG_highway	AirBags	DriveTrain	...	Passengers	Length	Wheelbase	\
0	False	False	False	...	False	False	False	
1	False	False	False	...	False	False	False	
2	False	False	False	...	False	False	False	
3	False	False	True	...	False	False	False	
4	False	True	False	...	False	False	False	
..	
88	False	False	False	...	False	False	False	
89	False	False	False	...	False	False	False	
90	False	False	False	...	False	False	False	
91	False	False	False	...	False	False	False	
92	False	False	False	...	False	False	False	

	Width	Turn_circle	Rear_seat_room	Luggage_room	Weight	Origin	Make
0	False	False	False	True	False	False	False
1	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False
3	True	False	False	False	False	False	False
4	False	False	False	False	False	False	False
..
88	False	False	False	True	False	True	False
89	False	False	False	False	False	False	False
90	False	False	False	False	False	False	False
91	False	False	False	False	False	False	False
92	False	False	False	False	False	False	False

[93 rows x 27 columns]

6-

```
[5]: print(df.isnull().sum())
```

Manufacturer	4
Model	1
CarType	3
Min_Price	7
Price	2
Max_Price	5
MPG_city	9
MPG_highway	2
AirBags	6
DriveTrain	7
Cylinders	5

```

EngineSize      2
Horsepower      7
RPM             3
Rev_per_mile    6
Man_trans_avail 5
Fuel_tank_capacity 8
Passengers      2
Length         4
Wheelbase       1
Width           6
Turn_circle     5
Rear_seat_room  4
Luggage_room    19
Weight          7
Origin          5
Make            3
dtype: int64

```

7-

```

[6]: dataf = pd.DataFrame(np.arange(20).reshape(-1,5), columns=list('abcde'))
      datafr = pd.DataFrame(dataf['a'])
      datafr

```

```

[6]:      a
0     0
1     5
2    10
3    15

```

8-A

```

[7]: dataf = pd.DataFrame(np.arange(20).reshape(-1,5), columns=list('abcde'))
      dataf[list('cbade')]

```

```

[7]:      c  b  a  d  e
0     2  1  0  3  4
1     7  6  5  8  9
2    12 11 10 13 14
3    17 16 15 18 19

```

8-B

```

[8]: def troca(dataf,coluna1,coluna2):
      c = dataf.columns.tolist()
      a = c[coluna2]
      c[coluna2] = c[coluna1]
      c[coluna1] = a
      dataf = dataf.reindex(columns=c)

```

```

return dataf
dataf =troca(dataf,2,3)
dataf

```

```

[8]:
   a  b  d  c  e
0  0  1  3  2  4
1  5  6  8  7  9
2 10 11 13 12 14
3 15 16 18 17 19

```

8-C

```

[9]: dataf = pd.DataFrame(np.arange(20).reshape(-1,5), columns=list('abcde'))
dataf[list('edcba')]

```

```

[9]:
   e  d  c  b  a
0  4  3  2  1  0
1  9  8  7  6  5
2 14 13 12 11 10
3 19 18 17 16 15

```

9-

```

[10]: dados=pd.read_csv('https://raw.githubusercontent.com/selva86/datasets/master/
↳Cars93_miss.csv')
dataframe= pd.DataFrame(dados)
dataframe = dataframe.loc[0:10]
dataframe[dataframe.columns[0:10]]

```

```

[10]:
Manufacturer      Model      Type  Min.Price  Price  Max.Price  MPG.city  \
0          Acura      Integra   Small      12.9   15.9      18.8      25.0
1           NaN      Legend  Midsize      29.2   33.9      38.7      18.0
2          Audi         90  Compact      25.9   29.1      32.3      20.0
3          Audi        100  Midsize       NaN   37.7      44.6      19.0
4          BMW        535i  Midsize       NaN   30.0       NaN      22.0
5          Buick    Century  Midsize      14.2   15.7      17.3      22.0
6          Buick   LeSabre   Large      19.9   20.8       NaN      19.0
7          Buick Roadmaster   Large      22.6   23.7      24.9      16.0
8          Buick   Riviera   Midsize      26.3   26.3      26.3      19.0
9        Cadillac   DeVille   Large      33.0   34.7      36.3      16.0
10         Cadillac   Seville  Midsize      37.5   40.1      42.7      16.0

MPG.highway      AirBags  DriveTrain
0          31.0          None      Front
1          25.0  Driver & Passenger      Front
2          26.0      Driver only      Front
3          26.0  Driver & Passenger       NaN
4          30.0           NaN      Rear

```

5	31.0	Driver only	NaN
6	28.0	Driver only	Front
7	25.0	Driver only	Rear
8	27.0	Driver only	Front
9	25.0	Driver only	Front
10	25.0	Driver & Passenger	Front

10-

```
[11]: dados=pd.read_csv('https://raw.githubusercontent.com/selva86/datasets/master/
↳Cars93_miss.csv')
dataframe= pd.DataFrame(dados)
dataframe =dataframe.loc[0:20]
dataframe[['Manufacturer','Model','Type']]
```

```
[11]:   Manufacturer      Model      Type
0      Acura      Integra      Small
1        NaN      Legend  Midsize
2      Audi         90   Compact
3      Audi        100  Midsize
4      BMW        535i  Midsize
5      Buick      Century  Midsize
6      Buick     LeSabre   Large
7      Buick  Roadmaster   Large
8      Buick     Riviera  Midsize
9  Cadillac     DeVille   Large
10  Cadillac     Seville  Midsize
11  Chevrolet  Cavalier   Compact
12  Chevrolet     Corsica  Compact
13  Chevrolet     Camaro   Sporty
14  Chevrolet     Lumina  Midsize
15  Chevrolet  Lumina_APV     Van
16  Chevrolet     Astro     Van
17  Chevrolet     Caprice   Large
18  Chevrolet  Corvette   Sporty
19        NaN  Concorde   Large
20  Chrysler     LeBaron  Compact
```

11-

```
[12]: df = pd.DataFrame(np.random.randint(10, 40, 60).reshape(-1, 4))
dfcopia = df.copy()
dfcopia = dfcopia.assign(Total=0)
dfcopia['Total']=dfcopia.sum(axis=1)
total = dfcopia.loc[dfcopia['Total']>100]
total[-2:]
```

```
[12]:
```

	0	1	2	3	Total
7	34	11	26	37	108
12	11	31	36	24	102

12-

```
[13]: d = pd.DataFrame(np.arange(25).reshape(5,-1))
      c=d.iloc[1].copy()
      b= d.iloc[2]
      d.at[1]=b
      d.at[2]=c
      d
```

```
[13]:
```

	0	1	2	3	4
0	0	1	2	3	4
1	10	11	12	13	14
2	5	6	7	8	9
3	15	16	17	18	19
4	20	21	22	23	24

```
[ ]:
```