

Nome: João Emanuel - Matricula: 162080263 - Data
13/agosto/2020

Dataframes ¶

In []:

```
#!/pip install pandas  
#!/conda install pandas
```

In [1]:

```
import pandas as pd
```

In [2]:

```
df = pd.DataFrame()  
df
```

Out[2]:

—

In [3]:

```
type(df)
```

Out[3]:

pandas.core.frame.DataFrame

In [4]:

```
df['Nome'] = [ 'Jessica', 'Aline']  
df
```

Out[4]:

	Nome
0	Jessica
1	Aline

In [5]:

```
df['cre'] = [ 7.8, 8.3 ]
df
```

Out[5]:

	Nome	cre
0	Jessica	7.8
1	Aline	8.3

Dataframe de casos COVID-19 - em 13/agosto/2020

In [7]:

```
atributos = [ 'Local', 'Confirmados', 'Novos casos (60 dias)', 'Casos (milhão)',  
'recuperados', 'Mortes' ]
```

In [8]:

```
atributos
```

Out[8]:

```
['Local',  
'Confirmados',  
'Novos casos (60 dias)',  
'Casos (milhão)',  
'recuperados',  
'Mortes']
```

In [10]:

```
data = {  
    'Estado': ['São Paulo', 'Bahia', 'Ceará', 'Rio de Janeiro', 'Paraíba'],  
    'Confirmados': [655181, 203020, 192422, 185610, 92897 ],  
    'Casos (milhão)': [14879, 13421, 21760, 11276, 23555 ],  
    'Mortes': [25869, 4135, 8052, 14295, 2071 ]  
}  
data
```

Out[10]:

```
{'Estado': ['São Paulo', 'Bahia', 'Ceará', 'Rio de Janeiro', 'Paraíba'],  
'Confirmados': [655181, 203020, 192422, 185610, 92897],  
'Casos (milhão)': [14879, 13421, 21760, 11276, 23555],  
'Mortes': [25869, 4135, 8052, 14295, 2071]}
```

In [11]:

```
df = pd.DataFrame(data)
df
```

Out[11]:

	Estado	Confirmados	Casos (milhão)	Mortes
0	São Paulo	655181	14879	25869
1	Bahia	203020	13421	4135
2	Ceará	192422	21760	8052
3	Rio de Janeiro	185610	11276	14295
4	Paraíba	92897	23555	2071

In [12]:

```
type(df)
```

Out[12]:

pandas.core.frame.DataFrame

In [13]:

```
df['Estado']
```

Out[13]:

```
0      São Paulo
1        Bahia
2        Ceará
3  Rio de Janeiro
4        Paraíba
Name: Estado, dtype: object
```

In [14]:

```
df[['Estado', 'Confirmados']]
```

Out[14]:

	Estado	Confirmados
0	São Paulo	655181
1	Bahia	203020
2	Ceará	192422
3	Rio de Janeiro	185610
4	Paraíba	92897

In [15]:

```
df
```

Out[15]:

	Estado	Confirmados	Casos (milhão)	Mortes
0	São Paulo	655181	14879	25869
1	Bahia	203020	13421	4135
2	Ceará	192422	21760	8052
3	Rio de Janeiro	185610	11276	14295
4	Paraíba	92897	23555	2071

In [16]:

```
df.head()
```

Out[16]:

	Estado	Confirmados	Casos (milhão)	Mortes
0	São Paulo	655181	14879	25869
1	Bahia	203020	13421	4135
2	Ceará	192422	21760	8052
3	Rio de Janeiro	185610	11276	14295
4	Paraíba	92897	23555	2071

In [23]:

```
df.tail()
```

Out[23]:

	Estado	Confirmados	Casos (milhão)	Mortes
0	São Paulo	655181	14879	25869
1	Bahia	203020	13421	4135
2	Ceará	192422	21760	8052
3	Rio de Janeiro	185610	11276	14295
4	Paraíba	92897	23555	2071

In [18]:

```
df.index
```

Out[18]:

```
RangeIndex(start=0, stop=5, step=1)
```

In [21]:

```
atributo = df.columns.values
atributo
```

Out[21]:

```
array(['Estado', 'Confirmados', 'Casos (milhão)', 'Mortes'], dtype=object)
```

In [22]:

```
atributo[0]
```

Out[22]:

```
'Estado'
```

In [24]:

```
df.values
```

Out[24]:

```
array([[ 'São Paulo', 655181, 14879, 25869],
       [ 'Bahia', 203020, 13421, 4135],
       [ 'Ceará', 192422, 21760, 8052],
       [ 'Rio de Janeiro', 185610, 11276, 14295],
       [ 'Paraíba', 92897, 23555, 2071]], dtype=object)
```

In [25]:

```
df2 = df
del df2['Estado']
df2
```

Out[25]:

	Confirmados	Casos (milhão)	Mortes
0	655181	14879	25869
1	203020	13421	4135
2	192422	21760	8052
3	185610	11276	14295
4	92897	23555	2071

In [29]:

```
k = df2.values
print(k)
```

```
[[655181 14879 25869]
 [203020 13421  4135]
 [192422 21760  8052]
 [185610 11276 14295]
 [ 92897 23555  2071]]
```

In [30]:

```
x = df[ ['Confirmados', 'Casos (milhão)']]
x
```

Out[30]:

	Confirmados	Casos (milhão)
0	655181	14879
1	203020	13421
2	192422	21760
3	185610	11276
4	92897	23555

In [31]:

```
y = df['Mortes']
y
```

Out[31]:

```
0    25869
1     4135
2     8052
3    14295
4     2071
Name: Mortes, dtype: int64
```

In [32]:

```
x = x.values
x
```

Out[32]:

```
array([[655181, 14879],
       [203020, 13421],
       [192422, 21760],
       [185610, 11276],
       [ 92897, 23555]])
```

In [33]:

```
y = y.values
y
```

Out[33]:

```
array([25869,  4135,  8052, 14295,  2071])
```

In [34]:

```
from sklearn import linear_model
from sklearn.metrics import r2_score
```

In [35]:

```
# Criando e treinando um modelo
modelo = linear_model.LinearRegression()
X = x
modelo.fit(X,y)
```

Out[35]:

LinearRegression()

In [36]:

```
def r2_est(X,y):
    modelo = linear_model.LinearRegression(normalize = False, fit_intercept = True)
    return r2_score(y, modelo.fit(X,y).predict(X))
```

In [37]:

```
print ('R2: %0.3f' % r2_est(X,y))
```

R2: 0.854

In [38]:

df

Out[38]:

	Confirmados	Casos (milhão)	Mortes
0	655181	14879	25869
1	203020	13421	4135
2	192422	21760	8052
3	185610	11276	14295
4	92897	23555	2071

In [39]:

```
import numpy as np

z = np.array([100000, 15000])

modelo.predict(z.reshape(1, -1))
```

Out[39]:

array([5536.56757517])

In [40]:

```
df
```

Out[40]:

	Confirmados	Casos (milhão)	Mortes
0	655181	14879	25869
1	203020	13421	4135
2	192422	21760	8052
3	185610	11276	14295
4	92897	23555	2071

Slicing

In [41]:

```
df[1:]
```

Out[41]:

	Confirmados	Casos (milhão)	Mortes
1	203020	13421	4135
2	192422	21760	8052
3	185610	11276	14295
4	92897	23555	2071

In [43]:

```
df[:3]
```

Out[43]:

	Confirmados	Casos (milhão)	Mortes
0	655181	14879	25869
1	203020	13421	4135
2	192422	21760	8052

In [45]:

```
df.describe().T
```

Out[45]:

	count	mean	std	min	25%	50%	75%	m
Confirmados	5.0	265826.0	222074.932497	92897.0	185610.0	192422.0	203020.0	655181
Casos (milhão)	5.0	16978.2	5378.087643	11276.0	13421.0	14879.0	21760.0	23555
Mortes	5.0	10884.4	9584.144970	2071.0	4135.0	8052.0	14295.0	25869



In [46]:

```
df.min()
```

Out[46]:

Confirmados 92897
Casos (milhão) 11276
Mortes 2071
dtype: int64

In [47]:

```
df.max()
```

Out[47]:

Confirmados 655181
Casos (milhão) 23555
Mortes 25869
dtype: int64

In [48]:

```
df['Casos (milhão)'].min()
```

Out[48]:

11276

In [50]:

```
df.Confirmados.min()
```

Out[50]:

92897

In []: