Pandas: Tipo de dados e Valores faltantes

Professor: **Nator Junior Carvalho da Costa**

October 15, 2022

Sumário

GroupBy

- sum();
- max();
- min();
- count();
- mean();
- describe();

mean()

```
1 import pandas as pd
2 df = pd.DataFrame({'ano': [2010, 2010, 2011,2012,2012],
3                    'idade': [10,16,20,25,31]})
4 df.groupby('ano').mean()
```

Out[78]:

ano	idade
2010	13
2011	20
2012	28

max()

```
1 import pandas as pd
2 df = pd.DataFrame({'ano': [2010, 2010, 2011,2012,2012],
3                    'idade': [10,16,20,25,31]})
4 df.groupby('ano').max()
```

Out[79]:

ano	idade
2010	16
2011	20
2012	31

min()

```
1 import pandas as pd
2 df = pd.DataFrame({'ano': [2010, 2010, 2011,2012,2012],
3                    'idade': [10,16,20,25,31]})
4 df.groupby('ano').min()
```

Out[88]:

	idade
ano	
2010	10
2011	20
2012	25

describe()

```
1 import pandas as pd
2 df = pd.DataFrame({'ano': [2010, 2010, 2011,2012,2012],
3                    'idade': [10,16,20,25,31]})
4 df.groupby('ano').describe()
```

Out[80]:

	idade							
	count	mean	std	min	25%	50%	75%	max
ano								
2010	2.0	13.0	4.242641	10.0	11.5	13.0	14.5	16.0
2011	1.0	20.0	NaN	20.0	20.0	20.0	20.0	20.0
2012	2.0	28.0	4.242641	25.0	26.5	28.0	29.5	31.0

sort_values


```
1 import pandas as pd
2 df = pd.DataFrame({
3     'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4     'col2': [5, 2, 5, 1, 3, 6],
5     'col3': [0, 7, 2, 8,1, 2],
6 })
7 print(df.sort_values(by=['col1']))
```

Out[85]:

	col1	col2	col3
0	g	5	0
5	g	6	2
2	n	5	2
4	n	3	1
1	t	2	7

sort_values


```
1 import pandas as pd
2 df = pd.DataFrame({
3     'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4     'col2': [5, 2, 5, 1, 3, 6],
5     'col3': [0, 7, 2, 8,1, 2],
6 })
7 print(df.sort_values(by=['col1','col2']))
```

Out[92]:

	col1	col2	col3
0	g	5	0
5	g	6	2
4	n	3	1
2	n	5	2
1	t	2	7

sort_values


```
1 import pandas as pd
2 df = pd.DataFrame({
3     'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4     'col2': [5, 2, 5, 1, 3, 6],
5     'col3': [0, 7, 2, 8,1, 2],
6 })
7 df.sort_values(by=['col1','col2'], ascending=False)
8
```

Out[93]:

	col1	col2	col3
3	w	1	8
1	t	2	7
2	n	5	2
4	n	3	1

Atividade

Para essa atividade considere a base de dados a seguir: https://drive.google.com/file/d/1YYduh61skvexLXRyxigCLYhC-QdJX7XnH/view?usp=sharing

1. Separa apenas os dados com **circunstancia_obito** igual a 2;
2. Quantificar os óbitos por sexo;
3. Quantificar os óbitos por idade;
4. Plotar os óbitos por sexo em uma gráfico do plotly;
5. Plotar os óbitos por idade em uma gráfico do plotly.

Tipos de dados

1. dtypes;
2. info();
3. astype();
4. to_numeric();

dtypes()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  print(df.dtypes)
8  col1      object
9  col2     float64
10 col3     float64
11 dtype: object
```

dtypes()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  print(df.dtypes.value_counts())
8  float64    2
9  object     1
10 dtype: int64
```

info()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  print(df.info())
8  <class 'pandas.core.frame.DataFrame'>
9  RangeIndex: 6 entries, 0 to 5
10 Data columns (total 3 columns):
11  #   Column  Non-Null Count  Dtype
12 ---  ------  --------------  -----
13  0   col1    6 non-null      object
14  1   col2    4 non-null      float64
15  2   col3    4 non-null      float64
```

astype()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [5, 10.1, 5, 1, 3, 15.2],
6      'col4': [0, 7, np.nan, 8,1, np.nan]
7  })
8  print(df.col3.astype(int))
9  0     5
10 1    10
11 2     5
12 3     1
13 4     3
14 5    15
15 Name: col3, dtype: int64
```

to_numeric()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, '-'],
5      'col3': [5, 10.1, 5, 1, 3, 15.2],
6      'col4': [0, 7, np.nan, 8,1, np.nan]
7  })
8  print(pd.to_numeric(df['col2'], errors='coerce'))
9  0    5.0
10 1    NaN
11 2    5.0
12 3    1.0
13 4    3.0
14 5    NaN
15 Name: col2, dtype: float64
```

Valores faltantes

1. isnull();
2. notnull();
3. fillna();
4. replace();
5. dropna();

isnull()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  df.isnull()
8      col1   col2   col3
9  0  False  False  False
10 1  False   True  False
11 2  False  False   True
12 3  False  False  False
13 4  False  False  False
14 5  False   True   True
```

notnull()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  df.isnull()
8     col1   col2   col3
9  0  True   True   True
10 1  True  False   True
11 2  True   True  False
12 3  True   True   True
13 4  True   True   True
14 5  True  False  False
```

isnull e notnull() onde usar?


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  df.loc[df.col2.isnull()]
8    col1  col2  col3
9  1    t   NaN   7.0
10 5    g   NaN   NaN
```

isnull e notnull() onde usar?


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  df.loc[df.col2.notnull()]
8    col1  col2  col3
9  0    g   5.0   0.0
10 2    n   5.0   NaN
11 3    w   1.0   8.0
12 4    n   3.0   1.0
```

fillna()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  print(df.fillna('Valor NULO'))
8    col1         col2         col3
9  0    g          5.0          0.0
10 1    t  Valor NULO          7.0
11 2    n          5.0  Valor NULO
12 3    w          1.0          8.0
13 4    n          3.0          1.0
14 5    g  Valor NULO  Valor NULO
```

fillna()


```
1  import pandas as pd
2  import numpy as np
3  df = pd.DataFrame({
4      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
5      'col2': [5, np.nan, 5, 1, 3, np.nan],
6      'col3': [0, 7, np.nan, 8,1, np.nan]
7  })
8  print(df.fillna(method ='pad'))
9    col1  col2  col3
10 0    g   5.0   0.0
11 1    t   5.0   7.0
12 2    n   5.0   7.0
13 3    w   1.0   8.0
14 4    n   3.0   1.0
15 5    g   3.0   1.0
```

replace()


```
1  import pandas as pd
2  df = pd.DataFrame({
3      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
4      'col2': [5, np.nan, 5, 1, 3, np.nan],
5      'col3': [0, 7, np.nan, 8,1, np.nan]
6  })
7  print(df.replace(to_replace = np.nan, value = -200))
8    col1   col2   col3
9  0    g    5.0    0.0
10 1    t -200.0    7.0
11 2    n    5.0 -200.0
12 3    w    1.0    8.0
13 4    n    3.0    1.0
14 5    g -200.0 -200.0
```

dropna()


```
1  import pandas as pd
2  import numpy as np
3  df = pd.DataFrame({
4      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
5      'col2': [5, np.nan, 5, 1, 3, np.nan],
6      'col3': [0, 7, np.nan, 8,1, np.nan]
7  })
8  print(df.dropna())
9    col1  col2  col3
10 0    g   5.0   0.0
11 3    w   1.0   8.0
12 4    n   3.0   1.0
```

dropna()


```
1  import pandas as pd
2  import numpy as np
3  df = pd.DataFrame({
4      'col1': ['g', 't', 'n', 'w', 'n', 'g'],
5      'col2': [5, np.nan, 5, 1, 3, np.nan],
6      'col3': [0, 7, np.nan, 8,1, np.nan]
7  })
8  print(df.dropna(subset=['col2']))
9    col1  col2  col3
10 0    g   5.0   0.0
11 2    n   5.0   NaN
12 3    w   1.0   8.0
13 4    n   3.0   1.0
```

Obrigado!

Pandas: Tipo de dados e Valores faltantes

Professor: **Nator Junior Carvalho da Costa**

October 15, 2022