

Resumo biblioteca Pandas

Luís Fernandes Saucedo Souza

VISÃO GERAL

Pandas Data frame é uma biblioteca (extensão com funcionalidades a mais) do Python utilizada para a manipulação e análise de dados. Sua facilidade de utilização e aprendizado faz com o que a biblioteca seja muito famosa.

Além disso, a Pandas Data frame é excelente para quem está começando no mundo da análise de dados.

Tecnicamente, a Pandas Data Frame é uma estrutura de dados tabular bidimensional potencialmente heterogênea e de tamanho variável com eixos rotulados (linhas e colunas).

Por sua vez, um quadro de dados é uma estrutura de dados bidimensional, o que significa que os dados são organizados em uma tabela em linhas e colunas. O Pandas Data Frame consiste em três componentes principais: dados, linhas e colunas ([COUTINHO DE OLIVEIRA, 2021](#)).

Exemplo:

```
In [1]: import pandas as pd

In [2]: pd.DataFrame({'A': [1, 2, 3]})
Out[2]:
   A
0  1
1  2
2  3
```

Figura 1. Exemplo dataframe.

Fonte: [User guide](#)

Principais comandos

Criar:

Há várias maneiras de criar um DataFrame de pandas. Na maioria dos casos, você usará o construtor DataFrame e fornecerá os dados, rótulos e outras informações. Você pode passar os dados como uma lista bidimensional, tupla ou matriz NumPy. Você também pode passá-lo como um dicionário ou instância da série pandas, ou como um dos vários outros tipos de dados ([STOJILJKOVIĆ, 2020](#)).


Para este exemplo, suponha que você esteja usando um dicionário para passar os dados, como na (Figura 2):

```
data = {  
    'name': ['Xavier', 'Ann', 'Jana', 'Yi', 'Robin', 'Amal', 'Nori'],  
    'city': ['Mexico City', 'Toronto', 'Prague', 'Shanghai',  
            'Manchester', 'Cairo', 'Osaka'],  
    'age': [41, 28, 33, 34, 38, 31, 37],  
    'py-score': [88.0, 79.0, 81.0, 80.0, 68.0, 61.0, 84.0]  
}
```

```
row_labels = [101, 102, 103, 104, 105, 106, 107]
```

```
df = pd.DataFrame(data=data, index=row_labels)
```

```
df
```



| | name | city | age | py-score |
|-----|--------|-------------|-----|----------|
| 101 | Xavier | Mexico City | 41 | 88.0 |
| 102 | Ann | Toronto | 28 | 79.0 |
| 103 | Jana | Prague | 33 | 81.0 |
| 104 | Yi | Shanghai | 34 | 80.0 |
| 105 | Robin | Manchester | 38 | 68.0 |
| 106 | Amal | Cairo | 31 | 61.0 |
| 107 | Nori | Osaka | 37 | 84.0 |

Figura 2: Dicionário entrada de dado.

Fonte: ([STOJILJKOVIĆ, 2020](#))

Localizar:

Se acessa uma coluna com o código: `dataframe['coluna']` ou `dataframe.coluna` e como output retorna um `pandas.Series` de uma coluna (Figura 3).

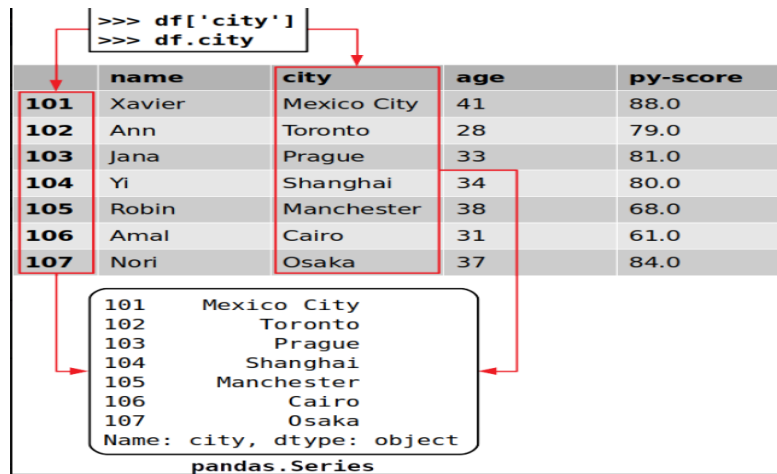


Figura 3: Localizando coluna.

Fonte: [\(STOJILJKOVIĆ, 2020\)](#)

Comandos principais

- `df.to_csv('data.csv')` (Transforma um Data Frame em um arquivo .csv)
- `pd.read_csv('data.csv')` (Lê um arquivo .csv e transforma em Data Frame)
- `df.index` (Retorna os valores dos índices das linhas)
- `df.columns` (Retorna os valores dos índices das colunas)
- `df.to_numpy()` (Transforma um Data Frame em uma array do NumPy)
- `df.ndim` (Retorna a dimensão do Data Frame)
- `df.shape` (Retorna o número de linhas e colunas do Data Frame)
- `df.size` (Retorna o número de dados do Data Frame)
- `df.loc[i]` (Localiza a linha com índice 'i')
- `df.iloc[i]` (Localiza a linha na posição 'i')

- `df.loc[11:15, ['name', 'city']]` (Como parâmetro pode filtrar a linha e coluna)
- `df.at[i, 'coluna']` (Retorna o dado na linha 'i' da coluna)
- `df.insert(loc=x, column='nome coluna', value=[a,b, ...])` (Insere uma coluna na localização 'x')
- `del df['coluna']` (Deleta a coluna)

Operações

Pode-se fazer operações com as colunas filtrá-las

Exemplos:

- `df['total'] = 0.4 * df['py-score'] + 0.3 * df['django-score'] + 0.3 * df['js-score']`
- `filter = df['django-score'] >= 80`
- `df[(df['py-score'] >= 80) & (df['js-score'] >= 80)]`
- `df['django-score'].where(cond=df['django-score'] >= 80, other=0.0)`
- `df_.mean()`

Gráficos

Com a biblioteca `matplotlib`, pode-se plotar gráficos selecionando quais dados quer se comparar (Figura 4).

```
import matplotlib.pyplot as plt
```

```
df.loc[:, ['py-score', 'total']].plot.hist(bins=5, alpha=0.4)  
plt.show()
```

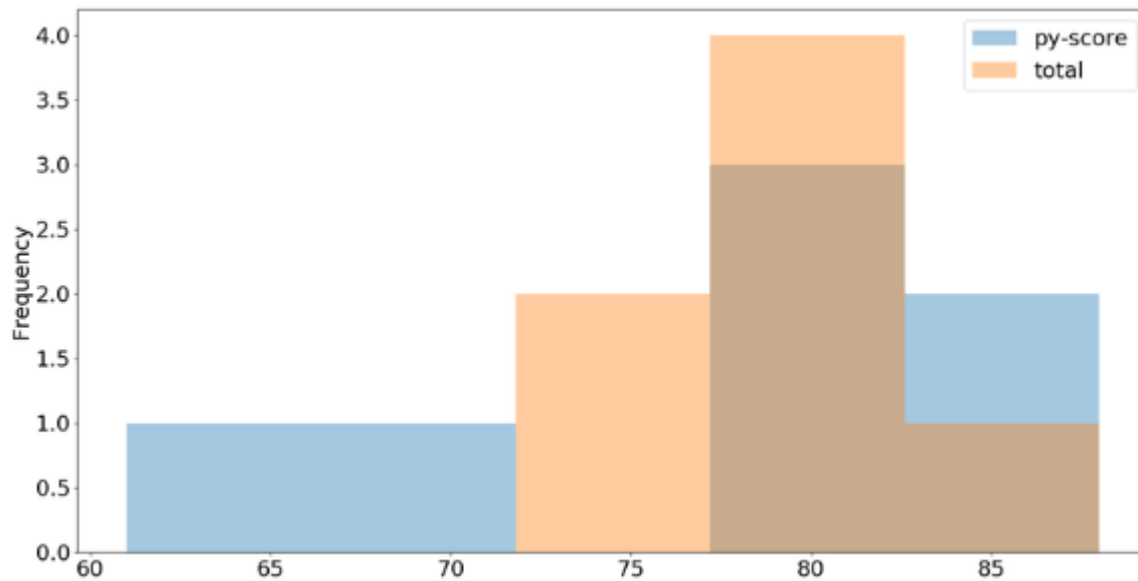


Figura 4. Gráfico dos dados do Data Frame

Fonte: [\(STOJILJKOVIĆ, 2020\)](#)

Aplicações

Saúde :

É usado na saúde para a análise de dados como os da pandemia do COVID-19, para estudar as relações de comorbidades como em [\(ROSA, 2022\)](#) ou para analisar a respostas dos governos, como em [\(IMTYAZ, 2020\)](#).

Biologia:

Análise de dados de florestas nativas, seu estado de preservação, seus recursos e sua composição como em [\(VAGIZOV, 2021\)](#).

Engenharia:

Sendo usado para machine learning na engenharia agrícola para analisar os dados de produção, através de um método de tomada de decisão, a qualidade da soja produzida [\(OLIVEIRA, 2022\)](#).

Química:

Análise de dados de espectroscopia de concentração de partículas de ouro na água [\(HUGHES, 2015\)](#).

Economia:

Obter projeções e estimativas do mercado de ação através da análise de big data [\(ARAÚJO, 2016\)](#).

Referências

ARAÚJO, Alcides Carlos De ; MONTINI, Alessandra De Ávila . Técnicas de Big Data e Projeção de Risco de Mercado utilizando Dados em Alta Frequência. 3. ed. São Paulo: **FUTURE STUDIES RESEARCH JOURNAL**, 2016. 83 – 108 p. v. 8. ISBN: [2175-5825](#).

- COUTINHO DE OLIVEIRA, Thiago. Quais são as vantagens e funcionalidades da biblioteca Pandas data frame?. **Voitto**, 2021. Disponível em: <https://www.voitto.com.br/blog/artigo/dataframe>. Acesso em: 18 maio 2023.
- HUGHES, Adam ; LIU, Zhaowen ; REEVES, M. E. . Scikit-spectra: Explorative Spectroscopy in Python. 6. ed. Washington: **Journal of Open Research Software**, 2015. v. 3. DOI: <http://dx.doi.org/10.5334/jors.bs>.
- IMTYAZ, Ayman ; HALEEM, Abid ; JAVAID, Mohd . Analysing governmental response to the COVID-19 pandemic. 10. ed. New Delhi, India: **Journal of Oral Biology and Craniofacial Research**, 2020. 504–513 p. DOI: [10.1016/j.jobcr.2020.08.005](https://doi.org/10.1016/j.jobcr.2020.08.005).
- OLIVEIRA, Daniela C. De ; BARBOSA, Uender C. ; BERGLAND, Alcídia C. R. O. Bergland ; RESENDE, Osvaldo; OLIVEIRA, Daniel E. C. De. G-SOJA - WEBSITE WITH PREDICTION ON SOYBEAN CLASSIFICATION USING MACHINE LEARNING. Goiás: **Journal of the Brazilian Association of Agricultural Engineering**, 2022. v. 42. ISBN [1809-4430](https://doi.org/10.1809/1809-4430).
- ROSA, Ruy Roberto Porto Ascenso; LAVAREDA FILHO, Ronem Matos ; LINHARES, José Elislande Breno De Souza . Influência das comorbidades para a ocorrência de óbitos por COVID-19 em 2020: razão de chances no estado do Amazonas.48. ed. Amazonas: HU Revista, 2022. 1-8 p. DOI: [10.34019/1982-8047.2022.v48.37689](https://doi.org/10.34019/1982-8047.2022.v48.37689).
- STOJILJKOVIĆ, Mirko. O DataFrame dos pandas: torne o trabalho com dados delicioso. **Real Python**, 2020. Disponível em: <https://realpython.com/pandas-dataframe>. Acesso em: 18 maio 2023.
- User guide. **Pandas**. Disponível em: [User Guide — pandas 2.0.1 documentation \(pydata.org\)](https://pandas.pydata.org/docs/user_guide/). Acesso em: 18 maio 2023.
- VAGIZOV, M ; POTAPOV, A; KONZHGOLADZE, K; STEPANOV, S; MARTYN, I. Prepare and analyze taxation data using the Python Pandas library. 876. ed. St. Petersburg: **IOP Conf. Series: Earth and Environmental Science**, 2021. DOI: [10.1088/1755-1315/876/1/012078](https://doi.org/10.1088/1755-1315/876/1/012078).