Educação Profissional Paulista

Técnico em **Ciência de Dados**

Secretaria da Educação

SÃO PAULO
GOVERNO DO ESTADO

Bibliotecas: Pandas, NumPy, SciPy, Matplotlib e Seaborn

Pandas – Manipulação de datas

Aula 1

Código da aula: [DADOS]ANO1C2B4S27A1

Mapa da Unidade 5 Componente 3

Bibliotecas: Pandas, NumPy, SciPy, Matplotlib e Seaborn

semana 23

Pandas: Acesso e seleção

semana 25

Pandas: Transformando *data frame*

Você está aqui!

semana 27

Pandas: Manipulação de datas

Bibliotecas: Pandas, NumPy, SciPy, Matplotlib e Seaborn

Mapa da Unidade 5 Componente 3

Você está aqui!

Pandas: Manipulação de datas

27

Aula 1

Código da aula: [DADOS]ANO1C2B4S27A1

Objetivos da Aula

- Conhecer o conceito de agrupar e resumir por uma coluna usando o método value_counts e *groupby* da biblioteca Pandas do Python.

Recursos Didáticos

- Recurso audiovisual para exibição de vídeos e imagens;
- Acesso ao laboratório de informática e/ou internet;
- Software Anaconda/Jupyter Notebook instalado ou similar.

Duração da Aula

50 minutos.

Competências Técnicas

- Ser proficiente em linguagens de programação para manipular e analisar grandes conjuntos de dados;
- Usar técnicas para explorar e analisar dados, aplicar modelos estatísticos, identificar padrões, realizar inferências e tomar decisões baseadas em evidências.

Competências Socioemocionais

- Colaborar efetivamente com outros profissionais, como cientistas de dados e engenheiros de dados;
- Trabalhar em equipes multifuncionais, colaborando com colegas, gestores e clientes.

Elaborado especialmente para o curso com apoio da ferramenta Microsoft Copilot e imagens © Getty Images.

Primeiras **ideias**

- Todas as datas da imagem são iguais?
- É possível transformar o formato das datas?
- Qual seria o formato de data correto?

Ponto de
partida

Datas

As diferentes convenções de formatação de datas surgiram devido a diversas razões históricas, culturais e práticas.

Aqui estão alguns fatores que contribuíram para essas diferenças:

Tradição cultural: No Brasil e em muitos países de língua portuguesa e espanhola, o formato dia/mês/ano é comum, enquanto nos EUA e países de língua inglesa, o formato mês/dia/ano predomina.

Influência histórica: As convenções de formatação de datas podem ser influenciadas por fatores históricos, como antigos sistemas de calendário ou de colonizadores.

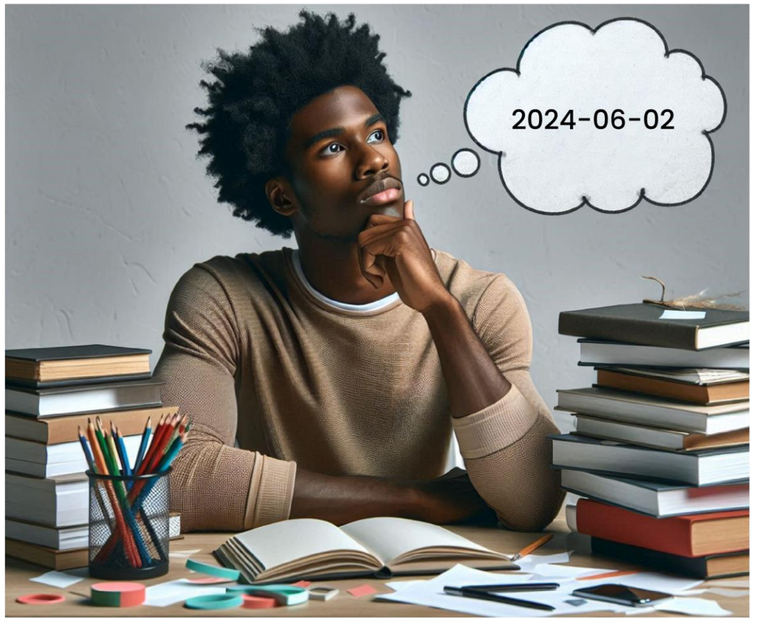
Os sistemas de datação anglo-saxônicos e latinos tiveram papéis importantes na formação dessas convenções.

Ponto de **partida**

Datas

Padrões internacionais: Alguns países adotaram o formato ISO 8601 (ano/mês/dia) para facilitar a comunicação global e evitar ambiguidades, especialmente em contextos que exigem precisão, como sistemas de informação.

Fonte: ISO/OBP, [s.d.].



Elaborado especialmente para o curso com apoio da ferramenta Microsoft Copilot.

Na sua opinião, qual é o melhor formato de data para trabalhar em ciência de dados e por quê? Compare os seguintes formatos: 02/06/2024, 06/02/2024 e 2024-06-02.

Construindo o **conceito**

Funções do Pandas

O Pandas é uma biblioteca de análise de dados em Python amplamente utilizada devido à sua capacidade de manipular e analisar dados de forma eficiente.

Duas funções importantes dentro do Pandas são **value_counts() e groupby(),** que são utilizadas para resumir e agrupar dados.

Vamos explorar cada uma delas em detalhes?

Construindo o **conceito**

Resumir: value_counts()

value_counts: É um método que retorna uma série com a contagem de valores únicos em uma coluna. Ele é útil para entender a distribuição dos valores em uma variável categórica.

```python
import pandas as pd

# Criando um DataFrame de exemplo
data = {'Categoria': ['A', 'B', 'A', 'C', 'B', 'A']}
df = pd.DataFrame(data)
df
```

	Categoria
0	A
1	B
2	A
3	C
4	B
5	A

```python
# Contagem de valores únicos na coluna 'Categoria'
df['Categoria'].value_counts()
```

```
Categoria
A    3
B    2
C    1
Name: count, dtype: int64
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Construindo o **conceito**

Resumir: value_counts()

Exemplo:

```python
import pandas as pd

data = {'Fruta': ['Maça', 'Banana', 'Maça', 'Laranja', 'Banana']}
df = pd.DataFrame(data)
df
```

	Fruta
0	Maça
1	Banana
2	Maça
3	Laranja
4	Banana

```python
# Contar o número de ocorrências de cada valor único na coluna 'Fruta'
df['Fruta'].value_counts()
```

```
Fruta
Maça       2
Banana     2
Laranja    1
Name: count, dtype: int64
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Construindo o **conceito**

Agrupar: groupby()

groupby: o método groupby() é usado para agrupar dados com base em uma ou mais colunas, e realizar operações agregadas em cada grupo. É muito útil quando se quer calcular estatísticas (como soma, média, contagem, etc.) para diferentes categorias em um conjunto de dados.

```python
import pandas as pd

# Criando um DataFrame de exemplo
data = {'Categoria': ['A', 'B', 'A', 'C', 'B', 'A']}
df = pd.DataFrame(data)
df
```

	Categoria
0	A
1	B
2	A
3	C
4	B
5	A

```python
# Agrupando por 'Categoria' e contar o número de ocorrências em cada grupo
df.groupby('Categoria').size()
```

```
Categoria
A    3
B    2
C    1
dtype: int64
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Construindo
o **conceito**

Agrupar: groupby()

Vamos continuar com o mesmo exemplo das frutas:

```
import pandas as pd

data = {'Fruta': ['Maça', 'Banana', 'Maça', 'Laranja', 'Banana']}
df = pd.DataFrame(data)
df
```

	Fruta
0	Maça
1	Banana
2	Maça
3	Laranja
4	Banana

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Construindo
o **conceito**

Agrupar: groupby()

```
# Agrupar o DataFrame por valores únicos na coluna 'Fruta' e contar o número de ocorrências em cada grupo
df.groupby('Fruta').size()
```

```
Fruta
Banana     2
Laranja    1
Maça       2
dtype: int64
```

Observe que o resultado foi o mesmo de **value_counts** e **groupby**. Entretanto, o value_counts apenas funciona para contagem, enquanto o groupby agrupa de várias formas.

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Construindo o **conceito**

Exemplos

1. Com o DataFrame abaixo, será que é possível usar os dois métodos para agrupar e resumir a coluna indicada?

a) Coluna Cores

```
import pandas as pd

cores = ["vermelho", "azul", "verde", "amarelo", "azul", "laranja", "preto",
         "verde", "cinza", "preto", "vermelho", "azul"]

df_cores = pd.DataFrame({"Cores": cores})
```

b) Coluna Frutas

```
import pandas as pd

frutas = ["maçã", "banana", "laranja", "uva", "morango", "abacaxi", "pera", "maçã", "banana", "laranja", "uva",
          "morango", "abacaxi", "pera", "maçã", "banana", "laranja", "uva", "morango", "abacaxi"]

df_frutas = pd.DataFrame({"Frutas": frutas})
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

Construindo o **conceito**

Exemplos

2. Com o DataFrame abaixo, analise:

```python
import pandas as pd

data = {
    "Produto": ["A", "B", "A", "C", "B", "A"],
    "Quantidade": [10, 5, 8, 12, 3, 6],
    "Preço": [100, 50, 80, 120, 30, 60],
}

df = pd.DataFrame(data)
```

Elaborado especialmente para o curso com a ferramenta Jupyter Notebook.

- Qual é o resultado de agrupar/resumir pela coluna “Produto”?
- Qual é o significado de agrupar/resumir pela coluna “Produto”, no contexto das vendas de produtos?
- Tente agrupar/resumir pelas outras colunas. O que acontece?
- O que acontece se trocar o método size() por sum()? df.groupby("Produto").sum()

Registro

Vamos fazer um **quiz**

Qual é a função de *groupby* no Pandas?

- Criar gráficos de grupo.
- Agrupar o DataFrame por chaves e operar em cada grupo.
- Ordenar o DataFrame por grupos de valores.
- Remover duplicatas do DataFrame.

Vamos fazer um **quiz**

Como é possível obter o tamanho de cada grupo após utilizar o *groupby*?

- Usando o método .size().
- Usando o método .count().
- Usando o método .length().
- Usando o método .group_size().

Vamos fazer um **quiz**

O que acontece quando é utilizado value_counts em uma coluna de um DataFrame com valores categóricos?

- Retorna apenas os valores numéricos.
- Converte todos os valores para *string*.
- Conta a frequência de cada categoria.
- Agrupa os valores por tipo de dado.

Diálogo

© Getty Images

O que nós **aprendemos hoje?**

Nuvem de palavras

© Getty Images

O que nós **aprendemos hoje?**

Então ficamos assim...

1. A função value_counts() é utilizada para contar o número de ocorrências de cada valor único em uma série de dados.

2. A função groupby() divide dados em grupos com base em um critério e aplica funções como soma, média ou contagem a cada grupo, útil para análise segmentada.

3. Ambas as funções/métodos são usados para agrupar e/ou resumir os dados de um DataFrame do Pandas.

Saiba mais

Uma outra forma de resumir os dados de uma tabela usando o Pandas, é o Describe.

Esse é o momento de conhecer melhor!
Basta acessar o artigo:

NEVES, D. *Ampliando a análise com o Describe*. Alura, 3 maio 2021. Disponível em: https://www.alura.com.br/artigos/ampliando-a-analise-com-describe/. Acesso em: 30 jul. 2024.

Referências da aula

ISO - ONLINE BROWSING PLATFORM (OBP). *ISO 8601-1: 2019*: date and time — Representations for information interchange — Part 1: Basic rules, [s.d.]. Disponível em: https://www.iso.org/obp/ui/en/#iso:std:iso:8601:-1:ed-1:v1:en/. Acesso em: 30 jul. 2024.

MCKINNEY, W. *Python para análise de dados*: tratamento de dados com Pandas, NumPy & Jupyter. São Paulo: Novatec, 2023.

PANDAS. *User Guide*: merge, join, concatenate and compare, [s.d.]. Disponível em: https://pandas.pydata.org/docs/user_guide/merging.html. Acesso em: 13 ago. 2024.

Identidade visual: imagens © Getty Images.

Educação Profissional Paulista

Técnico em **Ciência de Dados**

Secretaria da Educação

SÃO PAULO
GOVERNO DO ESTADO