

Manipulando dados da Bolsa de Valores com o Python

Prof. Alex Lopes Pereira

E-mail: alexlopespereira@gmail.com

Apresentação Pessoal



2000 a 2004 – Graduação em Engenharia de Computação no ITA



2005 a 2008 – Mestrado em Eng. de Computação e Eletrônica no ITA



2009 a 2015 – Doutorado em Eng. de Computação e Eletrônica no ITA



2004 a 2010 – Empreendedor, sócio em empresa de base tecnológica



2013 a 2017 – Censipam / Ministério da Defesa



2016 – Professor



2017 – Ministério da Economia



2020 – FIDO Fintech de Empréstimo P2P

Metodologia: Associar aspectos emocionais

- Aprender ciência de dados com exemplos práticos
 - E interessantes, curiosos, emotivos.
 - ✓ Meta: convertê-los em amantes da ciência de dados em 1h30
- Exemplos:
 - Saúde / COVID;
 - Investimentos;
 - ✓ Aula de Hoje: Bolsa de Valores



Investimento na Bolsa de Valores: CUIDADO!

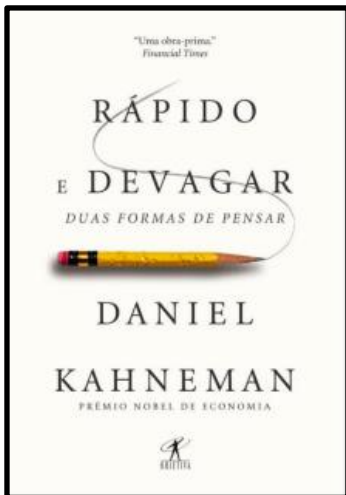
- Investimento de curto prazo na Bolsa
 - Está mais para um jogo de dado,
 - ✓ Do que para um jogo de poker.
- Viéses cognitivos que afetam
 - o desempenho de *traders*:
 - ✓ Loss aversion Bias
 - ✓ Bandwagon Bias
 - ✓ Confirmation Bias
 - ✓ Hindsight Bias



X

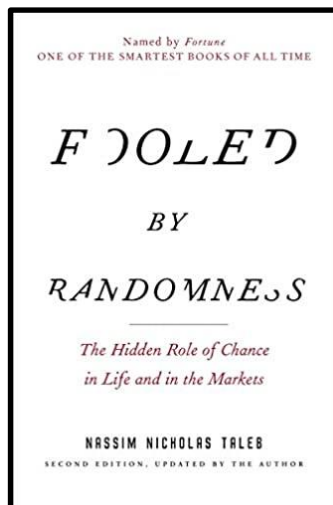
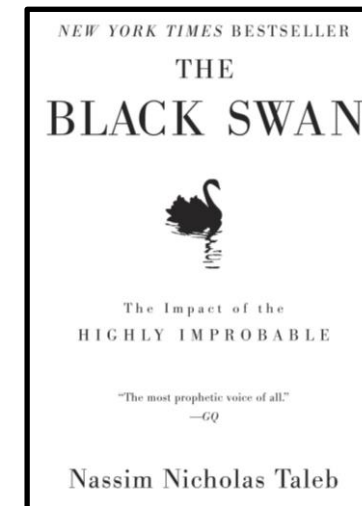


Livros para ficar resiliente aos seus viéses



- Thinking Fast and Slow
 - Daniel Kahneman

- Black Swan
 - Nassim Taleb



- Fooled By Randomness
 - Nassim Taleb

- Misbehaving
 - Richard Thaler



Efficient Market Hypothesis (EMH)

- Tem dois componentes
 - O preço está correto
 - Ninguém bate o mercado
 - ✓ There is no free lunch;
 - ✓ Porque todas as informações estão públicas.
- Até há algum tempo acreditava-se cegamente na EMH
 - Richard Thaler encontrou contra-exemplos
 - ✓ Que mostram que nem sempre a EMH é válida
 - Exemplo: Overreaction ao PreçoPorAção/LucroPorAção (P/L)
 - » e Regressão à média. *

* Thaler, R. H. (2015). *Misbehaving: The making of behavioral economics*. W W Norton & Co; **Capítulo 21**.

Análise Fundamentalista vs Análise Técnica

- Fundamentalista

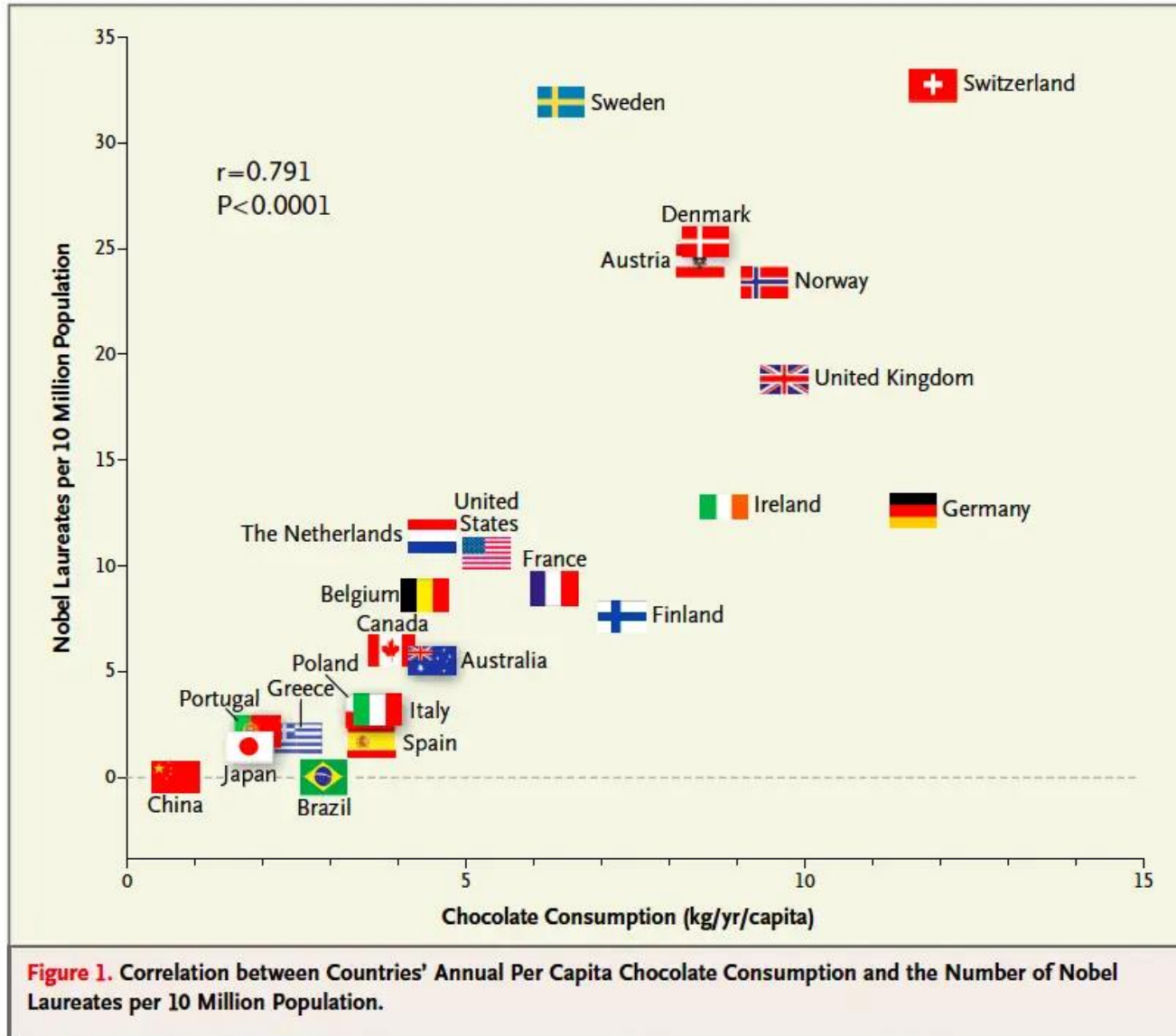
- análise de balanços,
- análise setorial,
- análise econômica,
- estudo da gestão da empresa e
- tendências de consumo.

- Técnica ou Gráfica

- Busca padrões nos gráficos
 - ✓ Assume que eles se repetem
 - com alguma frequência

Exemplos de Análise com Correlação

Exemplo 1:



<https://www.nejm.org/doi/full/10.1056/NEJMon1211064>

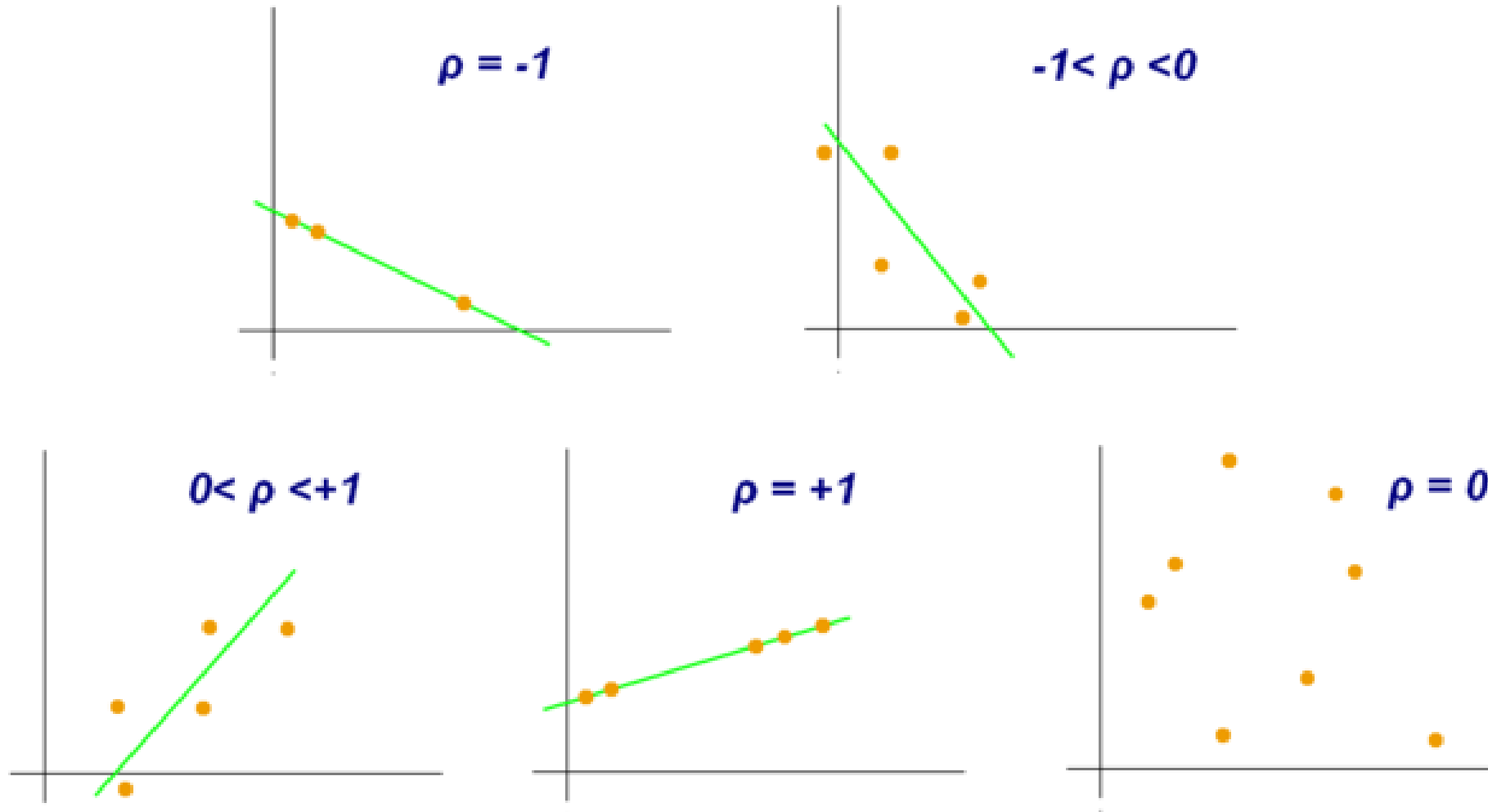
Exemplo 2:

Whether someone likes the taste of beer is the best predictor of if he or she will have sex on the first date.

<https://abcnews.go.com/Technology/dating-site-okcupid-shares-revealing-date-questions/story?id=12874977>

Coeficiente de Correlação

- Mede a direção e a força de uma relação linear



Cálculo do Coeficiente de Correlação

A	B	C	D	E	F
Aluno	Altura (cm)	Peso (kg)	Altura em unidade-padrão	Peso em unidade-padrão	(Peso em unidades-padrão) × (Altura em unidades-padrão)*
Nick	185	88	1,34	1,05	1,41
Elana	165	60	-0,49	-0,74	0,36
Dinah	170	70	-0,03	-0,09	0,01
Rebecca	172	67	0,15	-0,29	-0,04
Ben	183	80	1,16	0,54	0,63
Charu	175	58	0,43	-0,87	-0,37
Sahar	150	45	-1,86	-1,69	3,14
Maggie	158	58	-1,13	-0,87	0,98
Faisal	168	77	-0,21	0,35	-0,07
Ted	175	83	0,43	0,73	0,31
Narciso	175	81	0,43	0,61	0,26
Katrina	175	54	0,43	-1,12	-0,48
CJ	187	103	1,52	2,01	3,05
Sophia	155	53	-1,41	-1,18	1,67
Will	185	96	1,34	1,56	2,09
Média	170,34	71,53			Total = 12,95
Desvio padrão	10,91	15,66	Coeficiente de correlação = Total/n = 12,95/15 = 0,86		

Medição em unidade padrão é a distância à média medida em termos de desvio padrão. Calcula-se assim:

$$\frac{x - \bar{x}}{\sigma}$$

Fórmula:

$$r_{xy} = \frac{1}{n} \sum_{i=1}^n \left(\frac{x_i - \bar{x}}{\sigma_x} \right) \left(\frac{y_i - \bar{y}}{\sigma_y} \right)$$

Interpretação da Correlação

- Há alta correlação quando
 - a distância à média de uma variável oscilar consistentemente com a distância da outra variável.
- Não depende de unidade de medida
 - Pode-se medir a correlação entre
 - ✓ Peso e altura;
 - ✓ Quantidade de televisores em casa e o desempenho no ensino médio.
 - Isso porque o score padronizado (z-score)
 - ✓ é adimensional

$$Z_i = \frac{x_i - \bar{x}}{\sigma}$$

Protegendo seus investimentos com diversificação

- Uma estratégia bem aceita é
 - Investir em ativos não correlacionados.
- Como estimar o nível de correlação entre dois ativos?
 - Por exemplo, na bolsa de valores
 - ✓ Avaliar a correlação da sua cotação diária. (Aula de Hoje)
 - Sujeito a eventos catastróficos, ou seja,
 - ✓ Sisnes Negros (Nassim Taleb, Black Swan)
- Você também pode diversificar investindo em tipos distintos de ativos
 - Como P2P, Fundos Imobiliários e etc.
 - ✓ Não faz parte do escopo desta aula.

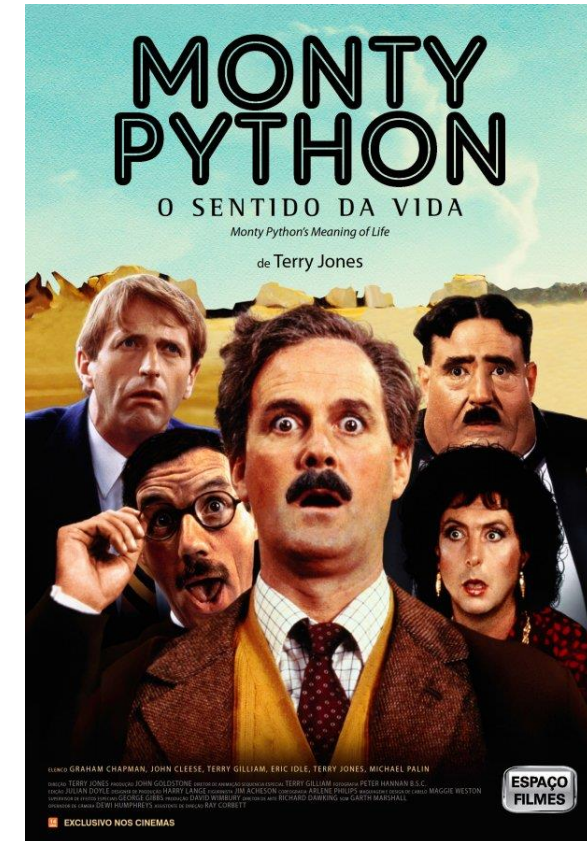
Introdução a linguagem Python



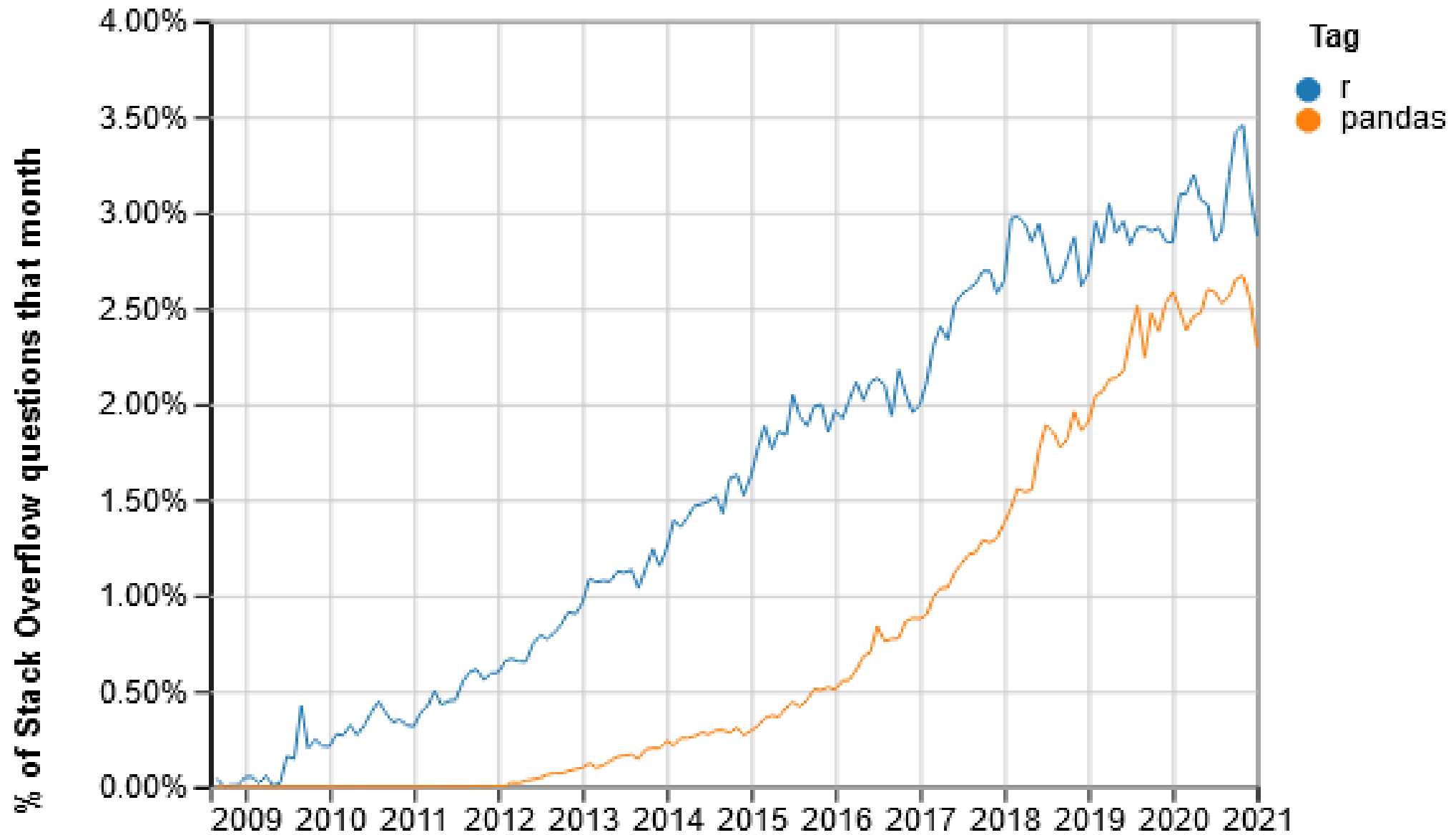
Pythonidae (Python family)



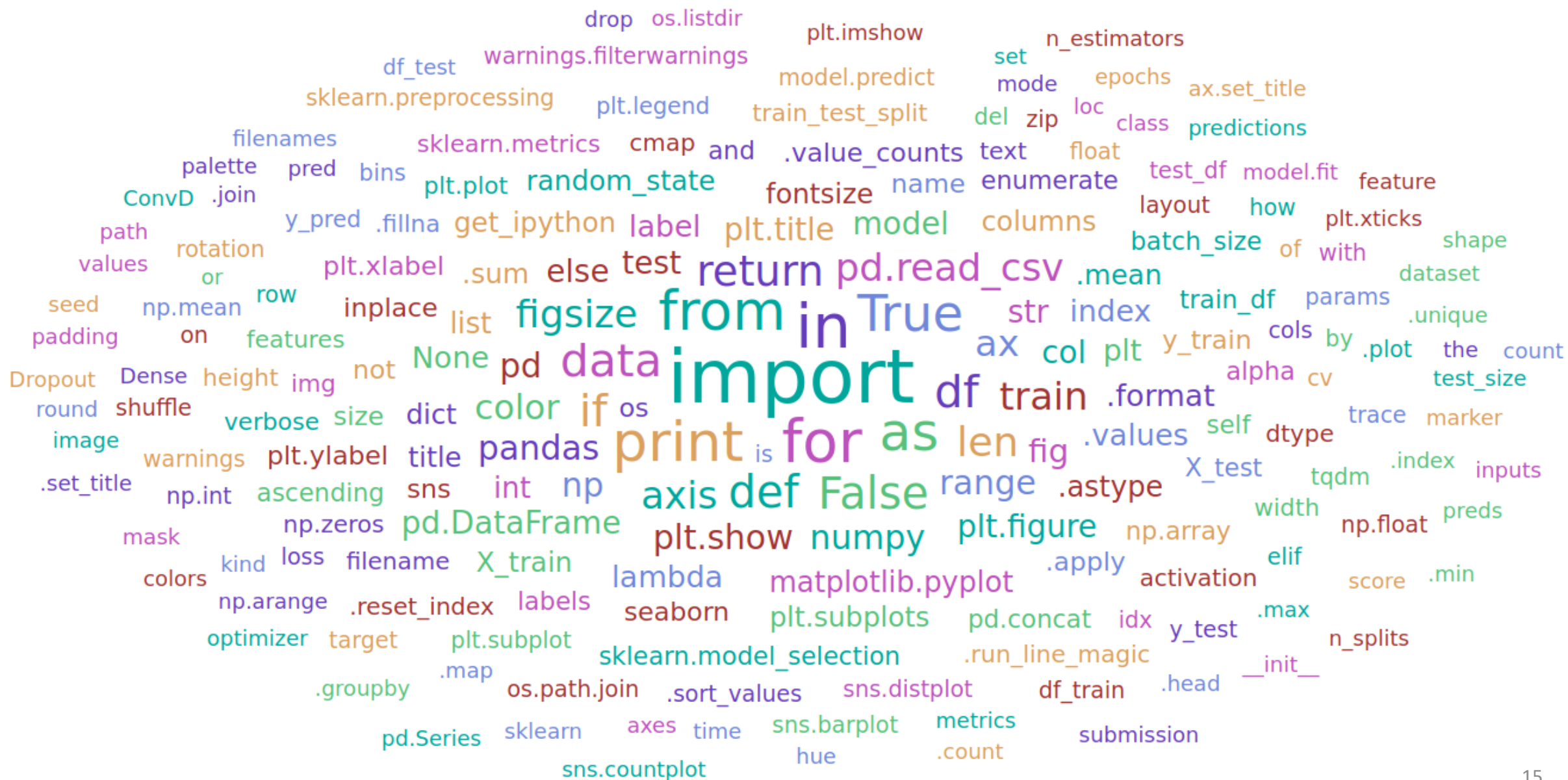
?



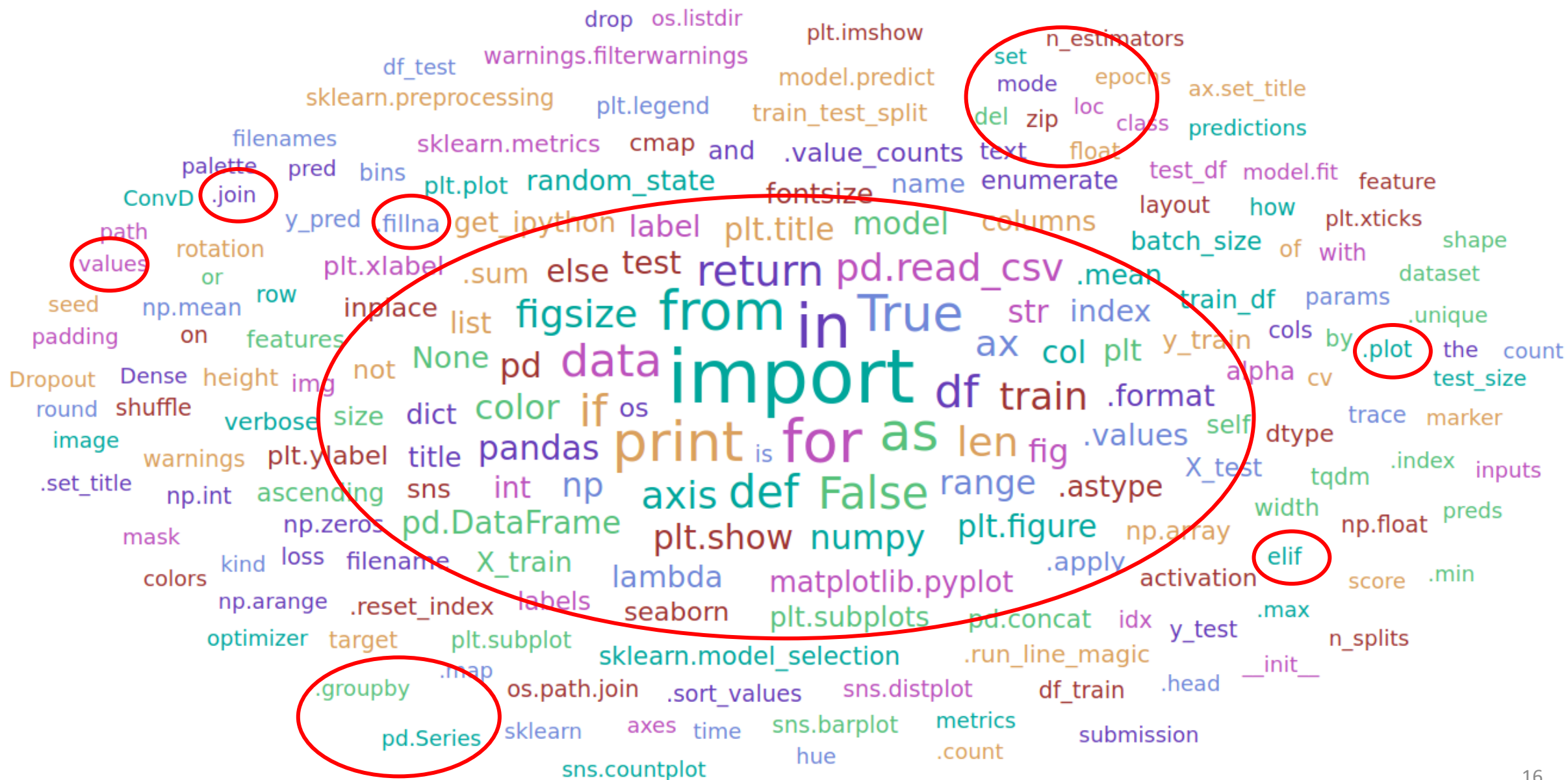
Python/Pandas vs R



Nuvem de Palavras de Arquivos Python do Kaggle

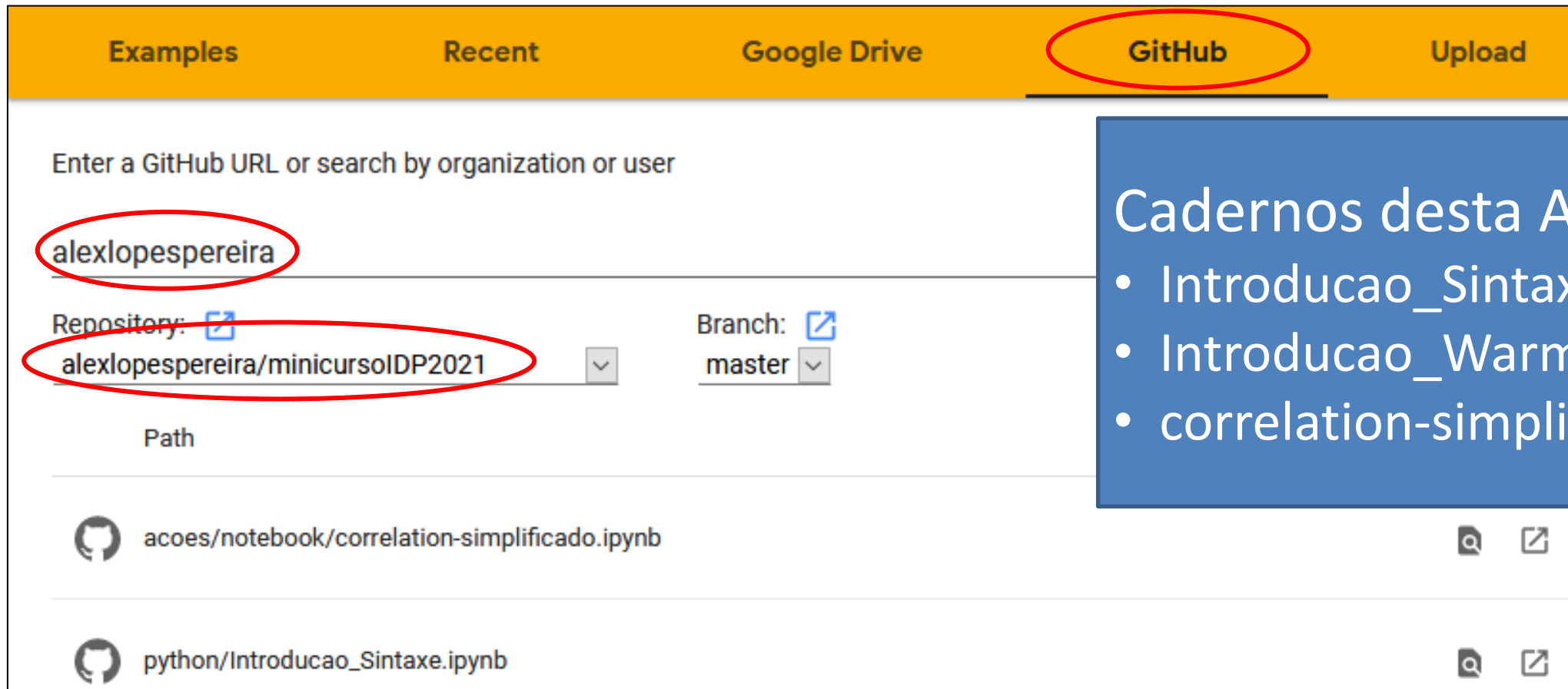


Nuvem de Palavras de Arquivos Python do Kaggle



Acesso aos cadernos Jupyter desta Aula

- <https://github.com/alexlopespereira/minicursoIDP2021>
- Abra no Colab <https://colab.research.google.com/>



The screenshot shows the Colab interface with a navigation bar at the top containing 'Examples', 'Recent', 'Google Drive', 'GitHub' (highlighted with a red circle), and 'Upload'. Below the navigation bar, there is a search bar with the placeholder text 'Enter a GitHub URL or search by organization or user'. The search bar contains the text 'alexlopespereira' (circled in red). Below the search bar, there is a 'Repository:' label and a dropdown menu showing 'alexlopespereira/minicursoIDP2021' (circled in red). To the right of the repository dropdown is a 'Branch:' label and a dropdown menu showing 'master'. Below these fields is a 'Path' label. At the bottom of the interface, there are two search results: 'acoess/notebook/correlation-simplificado.ipynb' and 'python/Introducao_Sintaxe.ipynb', each with a GitHub icon and a search icon.

Cadernos desta Aula

- Introducao_Sintaxe.ipynb
- Introducao_Warmup.ipynb
- correlation-simplificado.ipynb

Sintaxe do Python

- Dois pontos sinaliza o início de uma sentença composta
 - O conteúdo das sentenças compostas é aninhado com espaços/tabs
 - ✓ Sem chaves
- Ponto e vírgula para finalizar uma sentença é opcional

```
x = 2 # Atribuição com o sinal = (igual)
if x > 0:
    # adiciona 1 a x
    x = x + 1;
    print(x)
    print('x maior que zero')
else:
    print('x menor ou igual a zero')
```

```
3
x maior que zero
```

Laços (loops) do tipo for

- Uma maneira de **iterar** sobre uma coleção
- Usa-se o keyword **in** para referenciar a coleção.
 - **val** foi um termo escolhido pelo programador
 - **break** interrompe a iteração

```
sequence = [1, 2, 0, 4, 6, 5, 2, 1]
tot_until_5 = 0
for val in sequence:
    if val == 5:
        break
    tot_until_5 = tot_until_5 + val
print(tot_until_5)
```

Laço do tipo for com a função range

- `range(5)`
 - retorna um iterator da sequência 0, 1, 2, 3, 4
- `range(2, 6)`
 - retorna um iterator da sequência 2, 3, 4, 5

```
for val in range(3):  
    print(val*val)
```

```
0  
1  
4
```

list (Lista)

- Sequência de tamanho variável e conteúdo mutável (alterável)
 - Pode conter objetos de vários tipos
 - Define-se uma lista com colchetes []
 - `append` (inserir elementos), `pop` (remover elementos pelo índice)

```
al = [2, 4, 0, None]
print(al)
al.append(9)
print(al)
al.pop(1) # Remover pelo índice
print(al)
al.remove(0) # Use remove para remover pelo valor
print(al)
```

```
[2, 4, 0, None]
[2, 4, 0, None, 9]
[2, 0, None, 9]
[2, None, 9]
```

Funções Lambda (Anônima)

- São funções definidas em uma única sentença
- O resultado é o retorno de um valor
- São definidas pelo keyword lambda
 - Não há um significado especial para o termo lambda
 - ✓ a não ser "Estamos definindo uma função anônima"
- A função é dita anônima porque não damos um nome a ela

```
def exemplo(x):  
    return x * 2  
  
# Função lambda equivalente  
equiv_anon = lambda x: x * 2
```


Funções Lambda (Anônima)

```
def exemplo(x):  
    return x * 2
```

```
def apply_to_list(some_list, f):  
    return [f(x) for x in some_list]
```

```
ints = [4, 0, 1, 5, 6]  
apply_to_list(ints, lambda x: x * 2)  
apply_to_list(ints, exemplo)
```

```
[8, 0, 2, 10, 12]
```

```
[8, 0, 2, 10, 12]
```

Função Lambda para ordenar uma lista

- Criar uma função lambda para ordenar uma lista de strings
 - pelo tamanho dos seus elementos
 - ✓ do menor para o maior

```
strings = ['fo', 'card', 'bar', 'aaa2222a', 'abab']  
strings.sort(key=lambda x: len(x))
```

```
['fo', 'bar', 'card', 'abab', 'aaa2222a']
```

Biblioteca Datetime do Python

- Útil para gerenciar dados do tipo Data
 - Converter string para data e vice-versa
 - ✓ String para Datetime
 - `datetime.strptime('18/09/19 01:55:19', '%d/%m/%y %H:%M:%S')`
 - ✓ Datetime para string
 - `d.strftime('%Y-%m-%d')`
- Tabela de códigos (diretivas)
 - <https://docs.python.org/3.7/library/datetime.html#strftime-and-strptime-behavior>

Introdução ao Pandas ("panel data")



Fonte: <https://pt.wikipedia.org/wiki/Panda-gigante>

?

DataFrame

- Representa uma tabela rectangular de dados
- Contém uma coleção de colunas ordenadas
 - Onde cada coluna pode conter diferentes tipos de dados
 - ✓ numérico, string, boolean e etc
- Tem uma linha e uma coluna de índices

```
data = {'state': ['DF', 'DF', 'DF', 'SP', 'SP', 'SP'],  
        'year': [2000, 2001, 2002, 2001, 2002, 2003],  
        'pop': [1.5, 1.7, 3.6, 2.4, 2.9, 3.2]}  
df = pd.DataFrame(data)
```

	state	year	pop
0	DF	2000	1.5
1	DF	2001	1.7
2	DF	2002	3.6
3	SP	2001	2.4
4	SP	2002	2.9
5	SP	2003	3.2

Pandas: Escopo de funções para esta Análise

- Seleção de colunas
 - `df['nome_coluna']` ou `df[['coluna1', 'coluna2']]`
- Resetar o índice, transformando-o em colunas
 - `df.reset_index()`
- Mostrar as primeiras linhas de um dataframe
 - `df.head()`
- Aplicar uma operação em todas as células de uma coluna
 - `df['date'].apply(lambda x: OPERACAO SOBRE X)`
- Merge (junção de dois dataframes)
 - `df.merge(right=df2, left_on=['symbol', 'date'], ..)`
- Pivotar uma tabela
 - `df.pivot('date', 'symbol', 'change_pct')`
- Criar uma matriz de correlação
 - `df.corr()`

Mostrar as primeiras linhas de um DataFrame

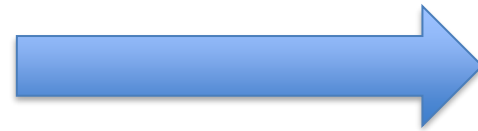
- Use a função head
 - `df.head()`

	b	d	e
SP	0.522095	-1.402182	-1.474172
DF	-0.457297	-1.631634	-0.632099
RJ	0.845294	-0.119810	1.740993
MG	1.051996	-1.109127	-1.876804

Transformar o índice em colunas

- Use a função `reset_index()`
 - `df.reset_index()`

		close
symbol	date	
AMZN	2020-10-08	3190.550049
	2020-10-09	3286.649902
	2020-10-12	3442.929932
	2020-10-13	3443.629883
	2020-10-14	3363.709961



`df.reset_index()`

	symbol	date	close
0	AMZN	2020-10-08	3190.550049
1	AMZN	2020-10-09	3286.649902
2	AMZN	2020-10-12	3442.929932
3	AMZN	2020-10-13	3443.629883
4	AMZN	2020-10-14	3363.709961

Função apply

- Aplica uma função ao longo de um eixo (linhas ou colunas) de um DataFrame

	b	d	e
SP	0.522095	-1.402182	-1.474172
DF	-0.457297	-1.631634	-0.632099
RJ	0.845294	-0.119810	1.740993
MG	1.051996	-1.109127	-1.876804

```
3 f = lambda x: x.max()*2
4 df.apply(f, axis='rows')
```

```
state      SPSP
year       4006
pop         7.2
dtype: object
```

```
2 df.apply(f, axis='columns')
```

```
SP      1.044189
DF     -0.914593
RJ      3.481985
MG      2.103993
dtype: float64
```

Operação Merge (ou Join em Banco de Dados)

Médico

CPF	Nome	Salario
11222731642	Jose Pereira	10
91498733332	Maria da Silva	20
81464221612	Pedro Martins	15

Professor

CPF	Nome	Salario
11222731642	Jose Pereira	6
91498733332	Maria da Silva	8
21564281600	Roberto Afonso	5

Merge

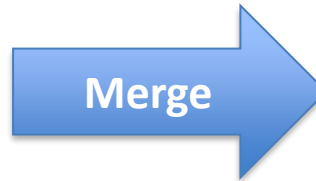
CPF	Nome	Salario_M	Salario_P
11222731642	Jose Pereira	10	6
91498733332	Maria da Silva	20	8

merge (fundir/juntar)

- Você precisa informar a coluna de junção
 - pode ser inferida a partir do contexto da interseção entre as tabelas
 - ✓ Também pode ser especificada com o argumento **on** (Ex.: `on='key'`)

2	df1
	key data1
0	b 0
1	b 1
2	a 2
3	c 3
4	a 4
5	a 5
6	b 6

2	df2
	key data2
0	a 0
1	b 1
2	d 2



6	df1.merge(right=df2, on='key')
	key data1 data2
0	b 0 1
1	b 1 1
2	b 6 1
3	a 2 0
4	a 4 0
5	a 5 0

Reshaping / Pivoting (Pivotar)

- Método pivot

- 3 argumentos: **index**, **columns**, **values**

- ✓ `df.pivot(index='Aluno', columns='Disciplina', values='Objetiva')`

- a função `melt()` faz a operação de despivotar

	Aluno	Disciplina	Objetiva	Discursiva
0	AlunoA	Portugues	8.5	6
1	AlunoA	Matematica	7.5	6.5
2	AlunoB	Geografia	9	7.5
3	AlunoB	História	10	7

	Disciplina	Geografia	História	Matematica	Portugues
Aluno					
AlunoA		NaN	NaN	7.5	8.5
AlunoB		9	10	NaN	NaN

Pivotar

Time shift (Deslocamento no Tempo)

- Criar uma coluna com os valores do dia anterior
- Como: deslocar os valores ao longo das linhas
 - fazendo um merge do dataframe com ele mesmo
 - ✓ deslocado de um dia

	symbol	date	close	ontem	date_desloc	close_desloc	ontem_desloc
0	AMZN	2021-02-04	3331.000000	2021-02-05	2021-02-03	3312.530029	2021-02-04
1	AMZN	2021-02-05	3352.149902	2021-02-06	2021-02-04	3331.000000	2021-02-05
2	AMZN	2021-02-09	3305.000000	2021-02-10	2021-02-08	3322.939941	2021-02-09
3	AMZN	2021-02-10	3286.580078	2021-02-11	2021-02-09	3305.000000	2021-02-10
4	AMZN	2021-02-11	3262.129883	2021-02-12	2021-02-10	3286.580078	2021-02-11

Obrigado

- Tarefinha (básica)
 - para dar mais um passo com a linguagem Python
 - ✓ https://github.com/alexlopespereira/minicursoIDP2021/blob/master/python/Introducao_Warmup.ipynb