PYTHON - NÍVEL INICIANTE

POWERED BY MULHERES EM DADOS

PROJETO SATISFAÇÃO DO CONSUMIDOR

AULA 01 - ENTENDIMENTO, CARREGAMENTO E PRÉ-PROCESSAMENTO DOS DADOS

15 DE JUNHO DE 2022

CRONOGRAMA

Acesse **aqui** o notebook da aula 01

JUNHO-JULHO 2022

DOM	SEG	TER	QUA	QUI	SEX	SAB
12	13	14	15 **Entendimento**	16	17	18
19	20	21	22 **Análise Exploratoria**	23	24	25
26	27	28	29 **Criação de modelo**	30	1	2
3	4	5	6 **Avaliação**	7	8	9

Entendimento, carregamento e pré-processamento dos dados

Entendimento do negócio e configurar o colab para pre-processar os dados

Análise exploratória dos dados (EDA)

Analisaremos os dados resultando nos principais insights sobre o negócio

Feature Engineering e criação de modelo de ML

Criaremos o nosso modelo de machine learning a partir das variáveis que mais fazem sentido para o negócio

Avaliação e melhorias no modelo de ML

Entenderemos melhor sobre como avaliar e melhorar a performance do nosso modelo de machine learning

SUMÁRIO

CRESCIMENTO NO USO DE PYTHON

Motivos para o crescimento

- É **grátis**
- Grande comunidade
- Roda em múltiplas IDEs
- Simples - de aprender, escrever e ler
- Vasta **quantidade de biblioteca**s, que fazem tudo
- Mas o que vem acelerando e a consolidando é o uso em **ciência de dados**
- Se espalhou para tudo (incluindo indústria)

BIBLIOTECAS PYTHON

- Bibliotecas são conjuntos de **funções para uma objetivo específico**
- Estão organizadas em: pypi.org/
- A instalação de biblioteca é feita pelo comando: pip install <nome da biblioteca>

Nas próximas páginas, veremos as bibliotecas usadas neste projeto.

BIBLIOTECA - PANDAS

- Pandas = Panel Data
- Biblioteca do Python para **manipulação e análise de dados**. Oferece estruturas de dados e operações para manipular tabelas.

Using a single column's values to select data.

```
In [39]: df[df["A"] > 0]
Out[39]:
                   A         B         C         D
2013-01-01  0.469112 -0.282863 -1.509059 -1.135632
2013-01-02  1.212112 -0.173215  0.119209 -1.044236
2013-01-04  0.721555 -0.706771 -1.039575  0.271860
```

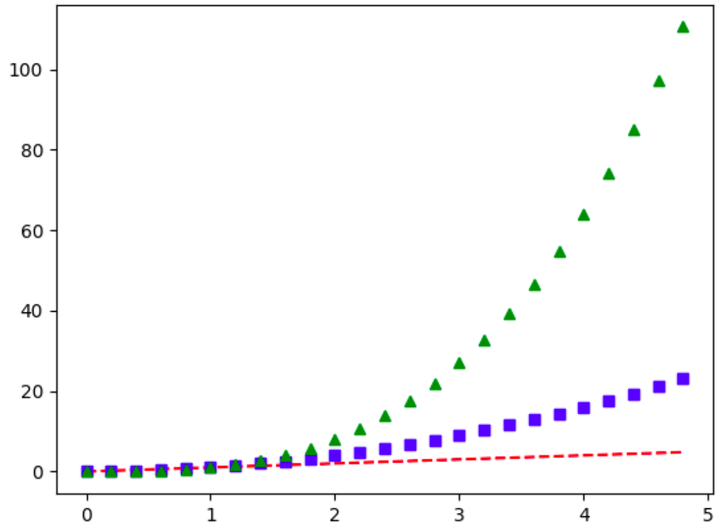
Selecting values from a DataFrame where a boolean condition is met.

```
In [40]: df[df > 0]
Out[40]:
                   A         B         C         D
2013-01-01  0.469112       NaN       NaN       NaN
2013-01-02  1.212112       NaN  0.119209       NaN
2013-01-03       NaN       NaN       NaN  1.071804
2013-01-04  0.721555       NaN       NaN  0.271860
2013-01-05       NaN  0.567020  0.276232       NaN
2013-01-06       NaN  0.113648       NaN  0.524988
```

BIBLIOTECA - MATPLOTLIB

- Biblioteca Python de **plotagem de gráficos** e sua **extensão de matemática numérica NumPy.**

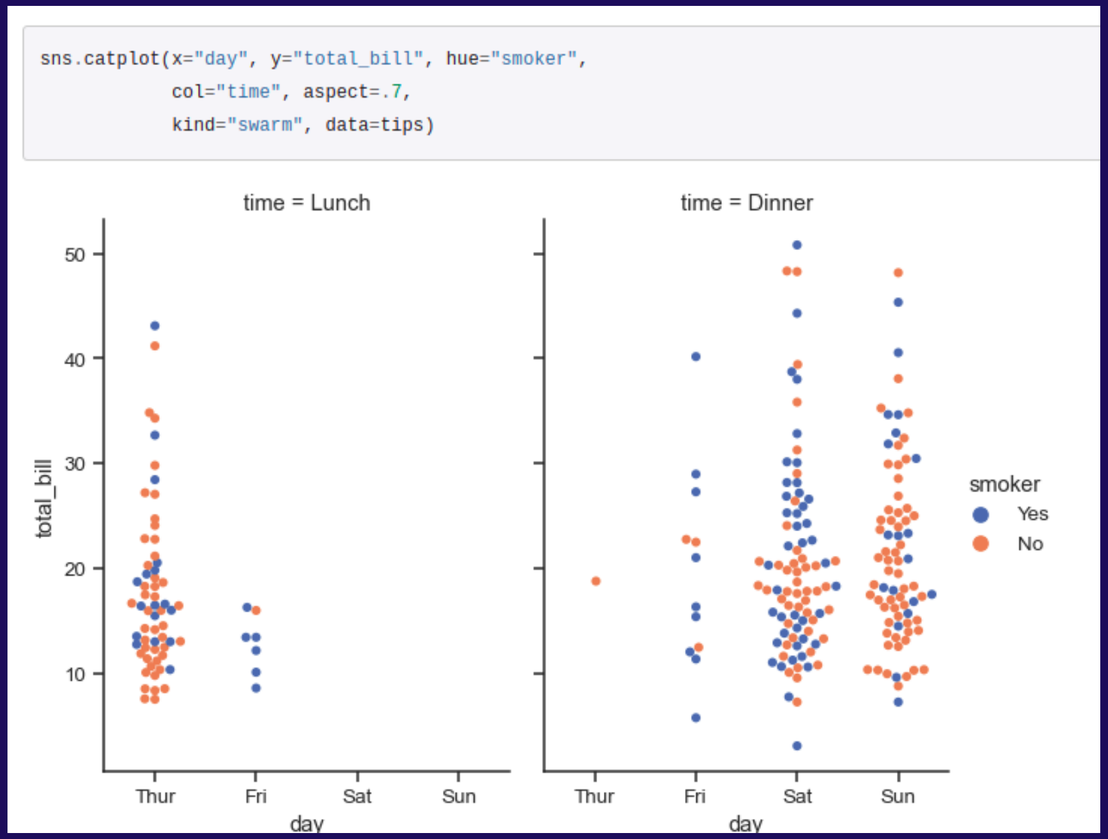
```
import numpy as np

# evenly sampled time at 200ms intervals
t = np.arange(0., 5., 0.2)

# red dashes, blue squares and green triangles
plt.plot(t, t, 'r--', t, t**2, 'bs', t, t**3, 'g^')
plt.show()
```



BIBLIOTECA - SEABORN

- Seaborn: statistical data visualization
- Biblioteca Python de **visualização de dados baseada em matplotlib**. Fornece uma interface de alto nível para desenhar **gráficos estatísticos atraentes e informativos**.

```
sns.catplot(x="day", y="total_bill", hue="smoker",
            col="time", aspect=.7,
            kind="swarm", data=tips)
```



VARIÁVEL

É um nome atribuído a um objeto. O sinal de igual é o operador de atribuição.
Exemplo:

```
x = 10  # variável == valor ou expressão
x       # x é a variável
10      # 10 é o valor atribuído à variável x
```

PALAVRAS RESERVADAS

Não podem ser usadas como nomes de variáveis:

False	break	else	if	not	while
None	class	except	import	or	with
True	continue	finally	in	pass	yield
and	def	for	is	raise	
as	del	from	lambda	return	
assert	elif	global	nonlocal	try	

#comando no Python:

```
import keyword
print(keyword.kwlist)
```

TIPOS DE VARIÁVEIS

INT	FLOAT	COMPLEX	BOOL	STR
Números inteiros	Números racionais (números com casas decimais e notações científicas)	Números complexos (veja **aqui** uma explicação do que são)	Retorna o resultado lógico (se é verdadeiro ou falso)	- Textos ou sequência de caracteres (inclui símbolos, espaços e pontuação - Necessário usar aspas simples ou duplas

COMO TRANSFORMAR UM TIPO DE VARIÁVEL EM OUTRO

- int( )
- float( )
- str( )

Exemplo:

```
a = 10 #número inteiro
b = float(a)
b
10.0 #número real
```

OPERADORES ARITMÉTICOS

** (potenciação)
+
-
/
*
% (módulo = resto de uma divisão)
// (divisão inteira)

OPERADORES RELACIONAIS

<=
<
>
=>
==
!=

OPERADORES LÓGICOS

is
is not
in
not in
not
or
and

PRECEDÊNCIA DOS OPERADORES

1. Parênteses mais internos
2. Operadores aritméticos
3. Operadores relacionais
4. Operadores lógicos

O QUE É UM DATASET?

- É uma base de dados, geralmente dispostas em formato tabular, com linhas e colunas bem definidas e organizadas com informações claras acerca de sua finalidade.

- O formato varia entre **CSV**, **TXT**, **XML** e até **XLS**.

ID_ALUNO	NOME	DISCIPLINA
122900087	Leticia	IHC
122900040	Carlos	Banco de dados
122900011	Pablo	Eng. Software
122900024	Vitória	Física

Exemplo de um dataset

O QUE É UM DATASET?

- O dataset que estamos utilizando no projeto é o **Olist**, um conjunto de dados de ecommerce do Brasil.

	order_id	customer_id	order_status	order_purchase_timestamp	order_approved_at	order_delivered_carrier_date
0	e481f51cbdc54678b7cc49136f2d6af7	9ef432eb6251297304e76186b10a928d	delivered	2017-10-02 10:56:33	2017-10-02 11:07:15	2017-10-04 19:55:00
1	53cdb2fc8bc7dce0b6741e2150273451	b0830fb4747a6c6d20dea0b8c802d7ef	delivered	2018-07-24 20:41:37	2018-07-26 03:24:27	2018-07-26 14:31:00
2	47770eb9100c2d0c44946d9cf07ec65d	41ce2a54c0b03bf3443c3d931a367089	delivered	2018-08-08 08:38:49	2018-08-08 08:55:23	2018-08-08 13:50:00
3	949d5b44dbf5de918fe9c16f97b45f8a	f88197465ea7920adcdbec7375364d82	delivered	2017-11-18 19:28:06	2017-11-18 19:45:59	2017-11-22 13:39:59
4	ad21c59c0840e6cb83a9ceb5573f8159	8ab97904e6daea8866dbdbc4fb7aad2c	delivered	2018-02-13 21:18:39	2018-02-13 22:20:29	2018-02-14 19:46:34

Imagem que mostra parte do dataset

Acesse **aqui** o notebook da aula 01

FUNÇÃO

- É um bloco de código que só é executado quando é chamado. Você pode passar dados (conhecidos como parâmetros) para uma função, retornando dados como resultado.

FUNÇÃO MERGE

- Função do pacote Pandas
- Utilizada para mesclar dois datasets por meio de colunas ou índices comuns

1st DataFrame:

	Name	Working Hours
0	Suraj	1
1	Zeppy	2
2	Alish	3
3	Sarah	5

2nd DataFrame:

	Name	Pay
0	Suraj	5
1	Zack	6
2	Alish	7
3	Raphel	8

Merged DataFrame:

	Name	Working Hours	Pay
0	Suraj	1.0	5
1	Alish	3.0	7
2	Zack	NaN	6
3	Raphel	NaN	8

Exemplo da utilização da função merge

FUNÇÃO JOIN

- Função do pacote Pandas
- join combina dois dataframes com base em seus índices

	key	A
0	K0	A0
1	K1	A1
2	K2	A2
3	K3	A3
4	K4	A4
5	K5	A5

	key	B
0	K0	B0
1	K1	B1
2	K2	B2

	key_caller	A	key_other	B
0	K0	A0	K0	B0
1	K1	A1	K1	B1
2	K2	A2	K2	B2
3	K3	A3	NaN	NaN
4	K4	A4	NaN	NaN
5	K5	A5	NaN	NaN

Exemplo da utilização da função join

FUNÇÃO DROPNA

- Função do pacote Pandas
- Dropna remove as linhas ou colunas com valores ausentes.

	name	toy	born
0	Alfred	NaN	NaT
1	Batman	Batmobile	1940-04-25
2	Catwoman	Bullwhip	NaT

```
>>> df.dropna()
     name        toy        born
1  Batman  Batmobile 1940-04-25
```

Exemplo da utilização da função dropna

FUNÇÃO DROP

- Função do pacote Pandas
- Drop remova linhas ou colunas quando especificamos nomes

	P	Q	R	S
0	0	1	2	3
1	4	5	6	7
2	8	9	10	11

```
df.drop(['Q', 'R'], axis=1)
```

	P	S
0	0	3
1	4	7
2	8	11

#axis = 1 significa a coluna

Exemplo da utilização da função dropna

FUNÇÃO ISNULL

- Função do pacote Pandas
- Isnull é usado para verificar e gerenciar a existência de valores NULL

```
>>> df.isna()
     age   born   name    toy
0  False   True  False   True
1  False  False  False  False
2   True  False  False  False
```

#No Python, isna( ) == isnull( )

Exemplo de saída da função isnull

LISTA VS TUPLA

- As **Listas** são usadas para armazenar vários itens em uma única variável, são criadas usando **colchetes**.

- **Listas** são **ordenadas**, **mutáveis**, permitem **valores duplicados** e são **indexados.** O primeiro item possui índice [0], o segundo item possui índice [1].

Faça esse comando e veja o resultado:

```
fruit = ["apple", "banana", "cherry"]
print(fruit)
```

- Mutável ->podemos alterar, adicionar e remover itens em uma lista após ela ter sido criada.

LISTA VS TUPLA

- As **Tuplas** são usadas para armazenar vários itens em uma única variável, são criadas usando **parênteses.**

- **Tuplas** são **ordenadas**, **imutáveis** e permitem valores **duplicados** e também são **indexados.** O primeiro item possui índice [0], o segundo item possui índice [1].

Faça esse comando e veja o resultado:

```
fruit_tuple = ("apple", "banana", "cherry", "apple", "cherry")
print(fruit_tuple)
```

- Imutáveis -> não podemos alterar, adicionar ou remover itens após a criação da tupla.

ESTRUTURA DE REPETIÇÃO- FOR

- Um loop **for** é usado para iterar sobre uma sequência (que é uma lista, uma tupla, um dicionário, um conjunto ou uma string).

FOR *<VARIÁVEL>* IN *<LISTA>*:
<EXECUTAR INTRUÇÕES>

- Vejamos esse exemplo:

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

temos o **fruits** como uma **lista**
o **for** pode ser traduzido -> para cada x na lista fruits
Imprimindo cada fruta da lista

- Também podemos juntar o for com uma função de intervalo **range( )**

```
for x in range(6):
  print(x)
```

o **for** pode ser traduzido -> para cada x no intervalo de (0, 6)
Imprimido os números de 0 até 5

REFERÊNCIAS

(CLIQUE NOS LINKS)

ACERVO LIMA. **Qual é a diferença entre Join e Merge em Pandas?** Acervo Lima, s/d.

DELFT STACK. **Fundir Pandas DataFrames no Índice**. Delft Stack, 2021.

ESTRELLA, Caio. **Pandas: combinando data frames com merge() e concat()**. Data Hackers, 2020.

MATPLOTLIB. **Matplotlib 3.5.2. Documentation**.

PANDAS. **Pandas Documentation**.

SEABORN. **Seaborn: Statistical Data Visualization**.

PARA SABER MAIS:

(CLIQUE NOS LINKS)

- Aprendendo com Python: Edição interativa @ Projeto Panda IME/USP
- Python Tutorial @ W3Schools
- Trilha de Estudos para Cientista de Dados @ Mulheres em Dados

Sob licença CC BY-NC-SA

OBRIGADA PELA PARTICIPAÇÃO!

POWERED BY MULHERES EM DADOS

- Próxima aula em 22/06, às 19h, no Discord
- Dúvidas e sugestões no canal #python

Equipe Python:
Andressa Apio, Crislane Maria, Érika Santos e Joice Oliveira