

+

• • • • • .
• • • • • .
• + .
+ •

|

+

FIAP

• • + .
■ □ *.
• • .
• + .
• • .



Statistics for DATA SCIENCE & **MACHINE LEARNING**

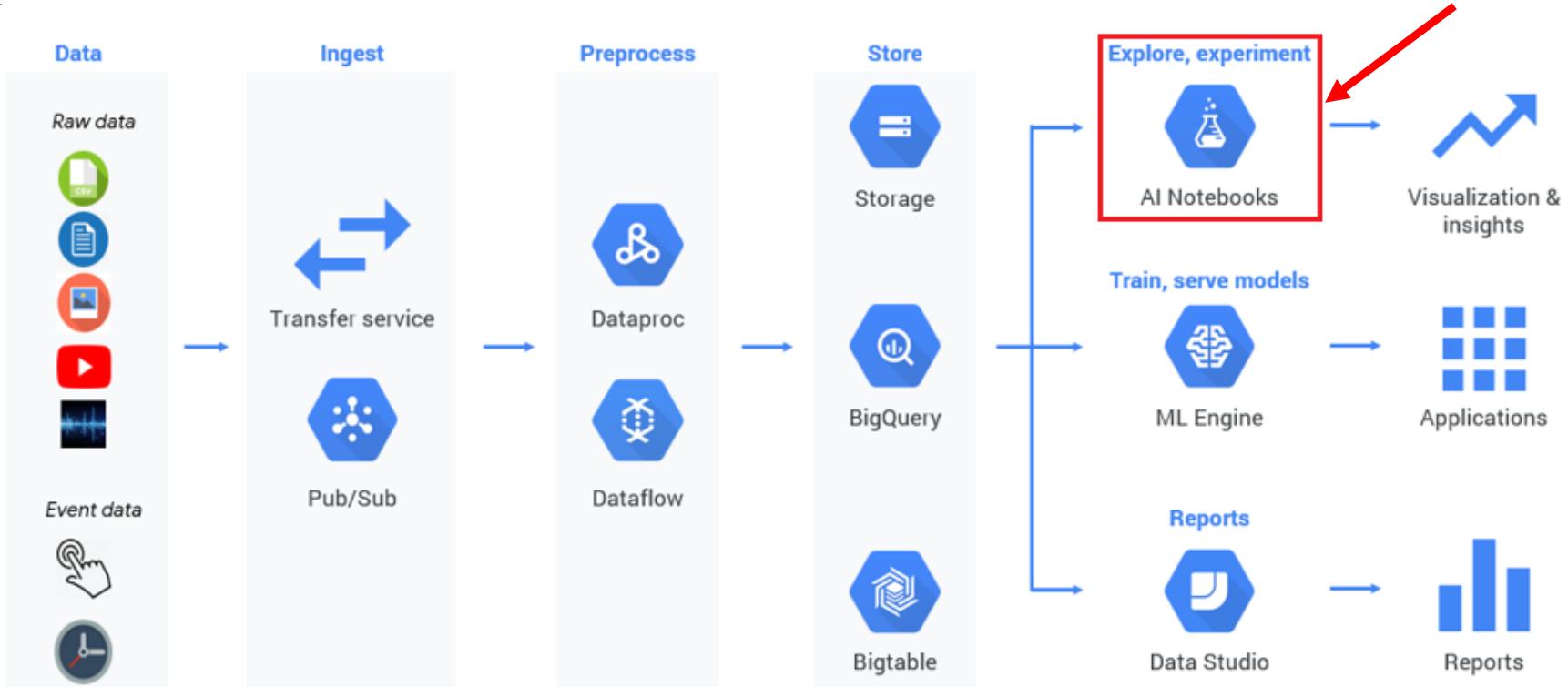
AULA 2

“Capacitar o aluno no entendimento de conceitos básicos
de estatística e análise de dados.”

“Preparar os alunos para entender e desempenhar
conceitos futuros relacionados à Análise Exploratória de
Dados e Machine Learning.”

Alcides C. Araújo

Fluxo geral projetos de *Data Science* e *Machine Learning*



NOTEBOOKS de Desenvolvimento



File Edit View Insert Cell

Welcome to the Jupyter Notebook Server

WARNING
Don't rely on this server

Your server is hosted there

Run some Python code:

To run the code below:

1. Click on the cell to select it
2. Press SHIFT+ENTER

A full tutorial for using the Notebook is available at [http://jupyter.org](#).

```
In [7]: interact(Lorenz, N=fixed(10), angle=(0.,360.),  
sigma=(0.0,50.0),beta=(0.,5.),rho=(0.0,50.0))
```

angle: 308.2

max_time: 12

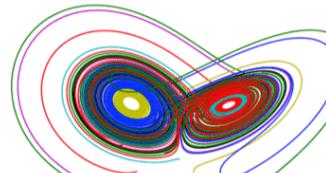
σ : 10

β : 2.6

ρ : 28

```
In [ ]: %matplotlib inline
```

```
import pandas as pd  
import numpy as np  
import matplotlib
```



This figure shows the Lorenz attractor, a complex, chaotic attractor in three-dimensional space. It is formed by three interlocking, butterfly-shaped trajectories. The colors of the trajectories transition through various hues, creating a vibrant, swirling pattern.

colab

História

- Os primeiros notebooks foram utilizados em ferramentas analíticas voltadas a academia, como o MATHEMATICA e o MatLab.
- O uso expandiu com os iPython`s notebooks, muito utilizados pela comunidade de **Python**.
- Atualmente existem diversos tipo de notebooks: **Jupyter** (sucessor do iPython), R Markdown, Apache Zeppelin e **Google Colaboratory**.
- Estes notebooks possibilitam o uso de vários kernels que possibilitam a programação em várias linguagens como: **Python**, R, Scala, Julia...

O que são

- Os primeiros notebooks foram utilizados para documentação de pesquisa. Qualquer pesquisador poderia replicar os resultados de uma pesquisa simplesmente aplicando o código nos mesmos dados fonte.
- Desta forma, os notebooks fornecem um ambiente para exploração, colaboração e visualização de resultados.
- Podem ser utilizados em ambientes de computação distribuída.
- Além de ser fácil o compartilhamento ([Google Drive](#), [Github](#), [Gitlab](#))

O que são

Data science

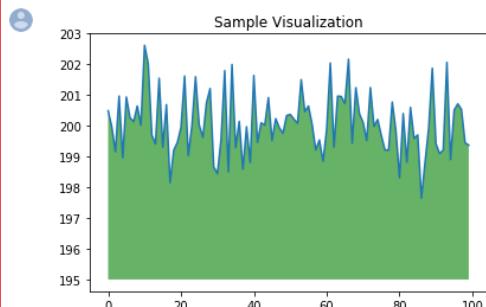
With Colab you can harness the full power of popular Python libraries to analyze and visualize data. The code cell below uses `numpy` to generate some random data, and uses `matplotlib` to visualize it. To edit the code, just click the cell and start editing.

```
[ ] import numpy as np
from matplotlib import pyplot as plt

ys = 200 + np.random.randn(100)
x = [x for x in range(len(ys))]

plt.plot(x, ys, '-')
plt.fill_between(x, ys, 195, where=(ys > 195), facecolor='g', alpha=0.6)

plt.title("Sample Visualization")
plt.show()
```



Texto explicativo

Código e Visualização

Boas práticas

- Um notebook um foco. O notebook ao ser compartilhado precisa ser voltado para um único objetivo específico. Por exemplo, voltado para exploração de dados ou voltado para um algoritmo de classificação.
- O código precisa estar explícito. Evitar criar linhas de códigos muito grandes.
- Códigos apresentados em módulos. Por exemplo, um bloco para importação dos pacotes, outro para carregar os dados, outro para alguma análise.
- Manter o código limpo e as variáveis criadas de forma explícita (fácil de serem interpretadas)

R: <https://style.tidyverse.org/>

Python: <https://www.python.org/dev/peps/pep-0008/>

O Google Colaboratory (Colab)

- É a ferramenta notebook do Google.
- Possibilita o uso de GPU`s e TPU`s.
- Facilidade na integração com Google Drive e BigQuery.

The screenshot shows the Google Colaboratory interface. At the top, there's a navigation bar with the title "Welcome To Colaboratory" and icons for File, Edit, View, Insert, Runtime, Tools, and Help. Below the navigation bar is a toolbar with "Table of contents" (highlighted), "Getting started", "+ Code", "+ Text", and "Copy to Drive". On the left, a sidebar titled "Table of contents" lists sections: "Getting started", "Data science" (which is currently selected and highlighted in orange), "Machine learning", "More Resources", "Machine Learning Examples", and "Section". The main content area has a large orange "CO" logo and the heading "What is Colaboratory?". The text explains that Colaboratory allows writing and executing Python in a browser with zero configuration, free access to GPUs, and easy sharing. There are also small control icons on the right side of the main content area.

Colab: <https://colab.research.google.com/notebooks/intro.ipynb>



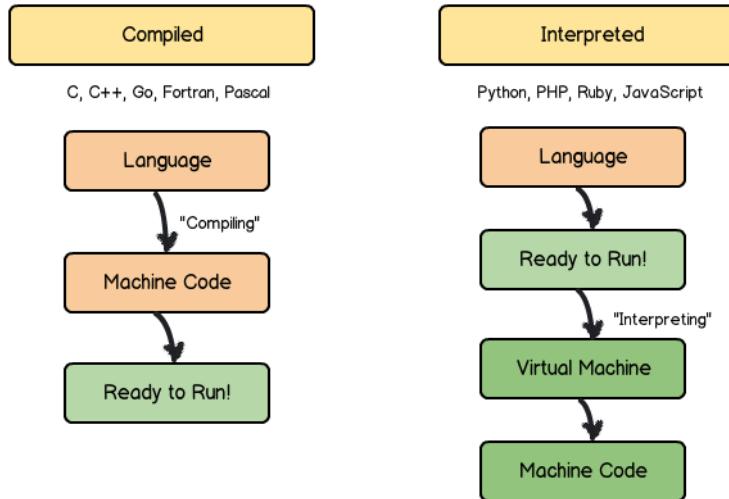
Python: introdução

- Surgiu em **1991** como uma linguagem interpretada.
- É bastante adotada devido sua facilidade de **leitura, simplicidade e explicitude**.
- Acabou sendo utilizada não somente para **pesquisa**, mas também **prototipação** e construir **sistemas em produção**.
- Números, textos, funções, classes, módulos são todos **OBJETOS**.



Python

- Linguagem compilada v.s. interpretada:
- Interpretadores irão rodar um programa linha a linha e executarão cada comando.



<https://medium.com/@astermanuelg/blurred-lines-is-ruby-an-interpreted-language-2d3d6bca3d37>

NumPy

- NumPy é um acrônimo para *Numerical Python*.
- É um **pacote** fundamental para análise de dados e computação científica (base do Pandas).
- Possibilita a criação e operação de matrizes multidimensionais (*ndarray*) de forma rápida e eficiente.
- Possui diversas **ferramentas para leitura e escrita de dados**, além de possibilitar aplicações para álgebra linear e geração de números aleatórios.



NumPy

<https://numpy.org/>

NumPy

- *Ndarray* (matriz NumPy)
- É um recipiente **multidimensional** genérico para dados homogêneos.
- Todos os elementos precisam ser do **mesmo tipo**
- Todo *Ndarray* possui seus elementos, dimensão e tipo (**int, float, str**)

In [8]: data

Out[8]:

```
array([[ 0.9526, -0.246 , -0.8856],
       [ 0.5639,  0.2379,  0.9104]])
```

Pandas



- Pandas é um acrônimo para Python Data Analysis Library.
- É um pacote poderoso e flexível para utilização em análise de dados
- Possibilita o uso de dados advindos de bases de dados (bancos SQL, .csv, .tsv) para criação de objetos com linhas e colunas (Serie ou Dataframe)

Pandas



- Series:
- Uma série no pandas é um **dicionário ordenado de tamanho fixo** mapeado por índice que está relacionado aos valores.
- Pode ser criado por uma matriz NumPy associada com **rótulos de dados (índice)**

```
In [4]: obj = Series([4, 7, -5, 3])
```

```
In [5]: obj  
Out[5]:  
0    4  
1    7  
2   -5  
3    3
```

Índices e dados

Pandas



Pandas

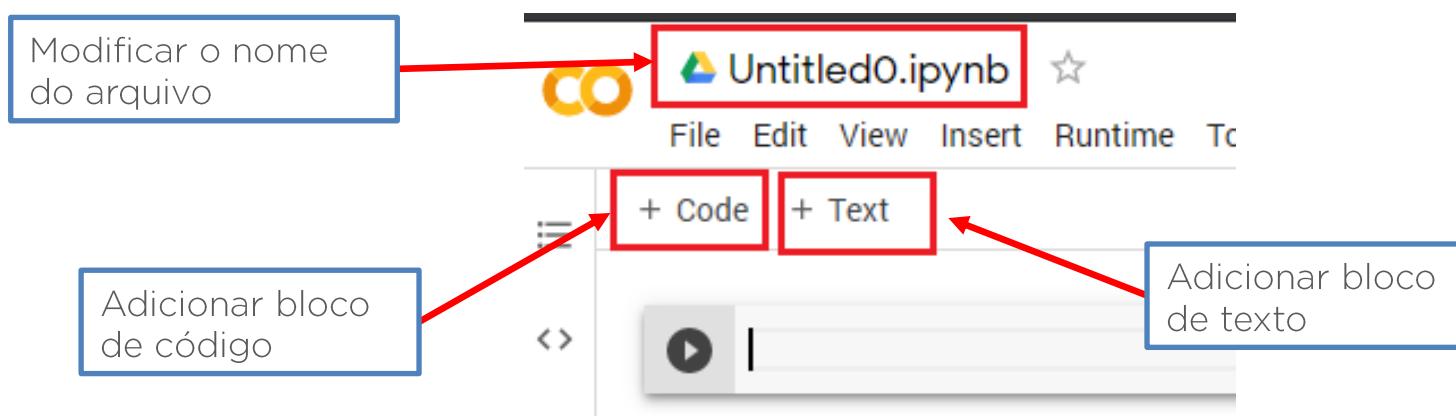
<https://pandas.pydata.org/>

- Dataframes:
- Os dataframes são representados por uma **estrutura tabular** contendo linhas e colunas.
- Cada coluna pode conter algum tipo específico de valor (**numérico, texto ou booleano**)

```
In [38]: frame
Out[38]:
      pop    state   year
0    1.5    Ohio  2000
1    1.7    Ohio  2001
2    3.6    Ohio  2002
3    2.4  Nevada  2001
4    2.9  Nevada  2002
```

Uso do Colab

- Para acessar o colab: <https://colab.research.google.com/>
- Criar um notebook novo: file - new notebook



Comandos básicos

- Adicionar bloco: ctrl + M + B
- Deletar bloco: ctrl + M + D
- Mudar de código para texto: ctrl + M + M
- Mudar de texto para código: ctrl + M + Y
- Rodar bloco de código: ctrl + Shift + Enter

OPERAÇÕES BÁSICAS

- Soma:

```
2+2
```

4

- Subtração:

```
4-3
```

1

- Multiplicação:

```
5*32
```

160

- Divisão:

```
76/21
```

3.619047619047619

```
4**4
```

256

- Criação de funções:

```
[7] def minha_fun_potencia(x, potencia):
```

```
    return x**potencia
```

```
[8] minha_fun_potencia(2,4)
```

OPERAÇÕES BÁSICAS

- Criação de vetores/lista:

```
[10] vetor1 = [1, 5, 7, 3, 8, 10]
     vetor1
```

```
[1, 5, 7, 3, 8, 10]
```

```
[13] vetor2 = list(range(5))
     vetor2
```

```
[0, 1, 2, 3, 4]
```

- Criação de vetores com números aleatórios:

```
rnd.seed(123)
vetor_aleatorio_0_1 = [round(rnd.uniform(0,1),4) for i in range(5)]
vetor_aleatorio_0_1
```

```
[0.0524, 0.0872, 0.4072, 0.1077, 0.9012]
```

```
vetor_aleatorio_inteiros = rnd.sample(range(10,100), 5)
vetor_aleatorio_inteiros
```

```
[53, 16, 30, 27, 81]
```

OPERAÇÕES BÁSICAS

- Operações com vetores:

```
np.array(vetor1)+np.array(vetor2)  
array([ 1,  6,  9,  6, 12, 15])
```

```
np.array(vetor1)*np.array(vetor2)  
array([ 0,  5, 14,  9, 32, 50])
```

OPERAÇÕES BÁSICAS

- Criação de matrizes:

```
matrizA = np.array([[2,3], [1,4]])
matrizA

array([[2, 3],
       [1, 4]])
```

- Criação de matrizes de números aleatórios:

```
matrizB = np.random.rand(2,2)
matrizB

array([[0.41148688,  0.44283629],
       [0.35156835,  0.65167443]])
```

OPERAÇÕES BÁSICAS

- Operações com matrizes:

```
matrizA+matrizB
```

```
array([[2.41148688, 3.44283629],  
       [1.35156835, 4.65167443]])
```

```
matrizA*matrizB
```

```
array([[0.82297377, 1.32850886],  
       [0.35156835, 2.60669771]])
```

```
matrizA.dot(matrizB)
```

```
array([[1.87767881, 2.84069586],  
       [1.81776028, 3.049534 ]])
```

EXERCÍCIOS

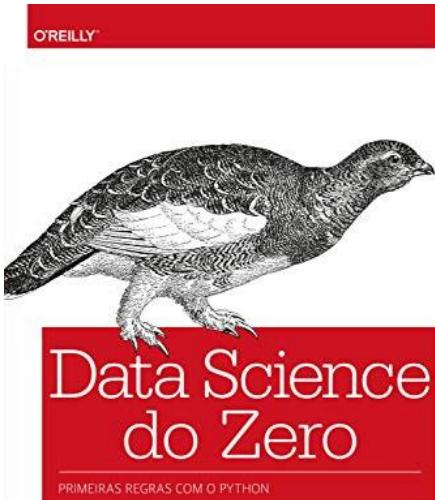
- Criar a seguinte função:

$$znorm = \frac{x - \min(x)}{\max(x) - \min(x)}$$

- Criar 2 vetores: vetorA (números de 1 a 20), vetorB (números aleatórios entre 1 e 20).
- Aplique a função criada nos vetores. O que ocorreu? Salve os resultados em novos vetores.
- Crie as matrizes matrixAB(20,2) e matrixAB_norm(20,2) utilizando os vetores criados anteriormente.
- Aplique as operações “*” e “dot” nas matrizes. Quais os resultados obtidos?

Onde estudar mais

- Livro:



- Vídeos:
- Introdução ao Colab:
<https://www.youtube.com/watch?v=inN8seMm7UI>
- Colab GPU, CPU:
<https://youtu.be/tCYSce6I8gA?list=PLQY2H8rRoyvwLbzbnKJ59NkZvQAW9wLbx>
- Colab GPU, CPU e Google Drive:
<https://youtu.be/vVe648dJOdl>



Joel Grus

OBRIGADO

FIAP

Copyright © 2020 | Professor Alcides C. Araújo

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.

+

• • • • • .
• • • • • .
• + .
+ •

|

+

FIAP

• • + .
■ □ *.
• • .
• + .
• • .