

Introdução a Python para análise de dados

Parte 2: NumPy e Pandas

Thiago Cardoso

thiago.figueredo@cesar.org.br



Conteúdo

NumPy

Pandas

Explorando um data set

Índice de exercícios

Exercício 1

Exercício 2

NumPy

Arrays uni/multidimensionais e funções para operações rápidas:

- Operações matemáticas e lógicas
- Manipulação de forma
- Seleção e ordenação
- ...

NumPy

Instalação

```
pip install numpy
```

Carregamento

```
import numpy as np
```

NumPy

Gerar dados

`array()`

`arange()`

`zeros()`

`ones()`

`eye()`

`linspace()`

`random.random()`

```
>>> np.array([2,3,4])
```

```
>>> np.arange(4)
```

```
array([0,1,2,3])
```

```
>>> np.linspace(1,10,20)
```

NumPy

Manipular forma

`shape()`

`reshape()`

```
>>> np.arange(10).shape  
(10,)
```

```
>>> np.arange(10).reshape(2,5)  
array([[0, 1, 2, 3, 4],  
       [5, 6, 7, 8, 9]])
```

NumPy

Operar sobre dados

`max()`

`min()`

`var()`

`std()`

`mean()`

`cumsum()`

`cumprod()`

```
>>> a = np.linspace(1,10,20)
```

```
>>> np.mean(a)
```

```
5.5
```

```
>>>
```


NumPy

Seleção de elementos

arr[indice]

arr[start:stop:step]

arr[condicional]

```
>>> a = np.linspace(1,10,20)
```

```
>>> a[1]
```

```
1.4736842105263157
```

```
>>> a[0:21:10]
```

```
array([1.          , 5.73684211])
```

```
>>> a[a > 8]
```

```
array([ 8.10526316,  8.57894737,  
       9.05263158,  9.52631579, 10.  
       ])
```

```
>>>
```

NumPy

Operações com arrays

arr1 +-/ arr2*

arr +-/ inteiro*

```
>>> a1 = np.ones(3)
```

```
>>> a2 = np.ones(3)
```

```
>>> a3 = a1 + a2
```

```
>>> a3 * 10
```

```
array([20., 20., 20.])
```

```
>>>
```

NumPy

Operações com arrays

arr1 +-/ arr2*

arr +-/ inteiro*

O que acontece quando
arr1 e arr2 tem tamanhos
diferentes?

```
>>> a1 = np.ones(3)
>>> a2 = np.ones(3)
>>> a3 = a1 + a2
>>> a3 * 10
array([20., 20., 20.])
>>>
```


Exercício 1

1. Usando NumPy, calcule os 10 primeiros elementos e a soma da progressão **artimética** cujo primeiro elemento é 10 e a razão é 3.
2. Usando NumPy, calcule os 10 primeiros elementos e a soma da progressão **geométrica** cujo primeiro elemento é 10 e a razão é 3.
3. Calcule o rendimento do CDI como nos exercícios anteriores do primeiro ao décimo ano usando NumPy.

Pandas

Python Data Analysis Library

Instalação

```
pip install pandas
```

Carregamento

```
import pandas as pd
```

Pandas

```
pd.Series(data)
```

```
pd.Series(data, index)
```

Lista indexada

```
>>> pd.Series([1, 3, np.NaN, 9])
```

```
0    1.0
```

```
1    3.0
```

```
2    NaN
```

```
3    9.0
```

```
>>> pd.Series([1, 3, np.NaN, 9],  
               [2019, 2020, 2021, 2022])
```

```
2019    1.0
```

```
2020    3.0
```

```
2021    NaN
```

```
2022    9.0
```


Pandas

Índices podem ter tipo diversos (tipos hasheáveis)

```
>>> pd.Series([1, 3, np.NaN, 9], ['a',  
    'b', 'c', 'd'])
```

```
a    1.0
```

```
b    3.0
```

```
c    NaN
```

```
d    9.0
```

```
>>> pd.Series([1, 3, np.NaN, 9],  
    [(0,0), (0,1), (1,0), (1,1)])
```

```
(0,0)    1.0
```

```
(0,1)    3.0
```

```
(1,0)    NaN
```

```
(1,1)    9.0
```

Pandas

Dados são arrays NumPy,
todas as operações
anteriores são aplicáveis

```
>>> s = pd.Series([1, 3, 7, 9],  
[2019, 2020, 2021, 2022])
```

```
>>> s.values
```

```
array([1., 3., 7., 9.])
```

```
>>> s.sum( )
```

```
20
```

```
>>> s.cumsum( )
```

```
2019      1
```

```
2020      4
```

```
2021     11
```

```
2022     20
```

Pandas

`pd.DataFrame`

Estrutura bidimensional

Similar a um dicionário de
Series

```
>>> d = {  
    'attempts': [1, 3, ... ],  
    'name': ['Anastasia', 'Dima', ... ],  
    'qual': ['yes', 'no', ... ],  
    'score': [12.5, 9, ... ],  
}  
  
>>> df = pd.DataFrame(d)  
  
>>>
```


Pandas

Visualizar dados

dataframe.shape

dataframe.head() / tail()

dataframe.describe()

dataframe.sort_values(by=column)

dataframe.groupby(column)

dataframe.T

```
>>> df.shape
```

```
(11, 4)
```

```
>>> df.head(2)
```

```
...
```

```
>>>
```

Pandas

Selecionar dados

dataframe.loc[inicio:fim, inicio:fim]

dataframe.iloc[inicio:fim, inicio:fim]

dataframe[condição]

```
>>> df.loc[:, ['name', 'score']]
```

```
0  Anastasia  12.5
```

```
1  Dima       9.0
```

```
... .....
```

```
>>> df.iloc[:, [1, 3]]
```

```
0  Anastasia  12.5
```

```
1  Dima       9.0
```

```
... .....
```

```
>>> df[df.score > 10]
```

```
...
```

```
>>>
```

Pandas

Boa parte das operações de arrays NumPy e Series estão disponíveis para o DataFrame

São executadas para todas as colunas (ou uma seleção delas)

```
>>> df.mean()  
attempts      1.818182  
score         13.777778  
  
>>> df.max()  
attempts      3  
name          Suresh  
qualify       yes  
score         20  
  
>>> df.max(numeric_only=True)  
attempts      3.0  
score         20.0
```


Pandas

Arquivos

`pd.read_csv()`

`dataframe.to_csv()`

Há suporte a outros
formatos

```
>>> df = pd.read_csv('data.csv')
```

```
>>> df.sort_values(by='score')
```

```
>>> df.to_csv('sorted_data.csv')
```

Pandas + matplotlib

Histogramas

```
dataframe.hist()
```

Matriz de dispersão

```
from pandas.plotting import  
scatter_matrix
```

```
scatter_matrix(dataframe)
```

```
>>> df.hist()
```

```
>>> plt.show()
```

```
>>> scatter_matrix(df)
```

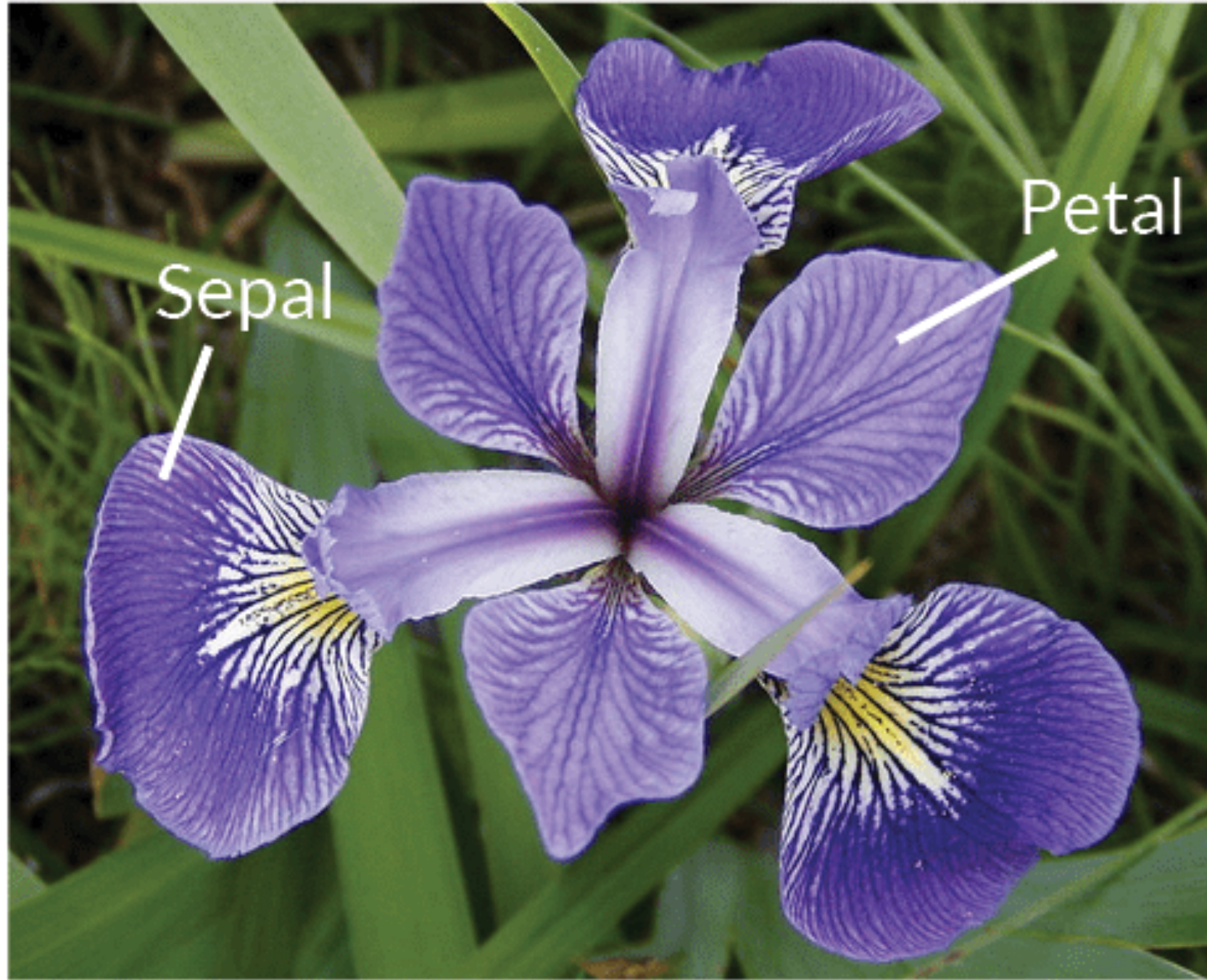
```
>>> plt.show()
```

Exercício 2

Considerando o arquivo data.csv:

- A. Leia os dados em um DataFrame e explore-o
- B. Adicione uma nova coluna *ratio*, sendo a razão entre pontuação e tentativas para cada pessoa.
- C. Imprima as informações das pessoas com o maior e menor *ratio*.
- D. Ordene decrescentemente pelo *ratio*.
- E. Salve o novo arquivo CSV ignorando a coluna de índices do DataFrame.

Explorando um data set



Iris Versicolor

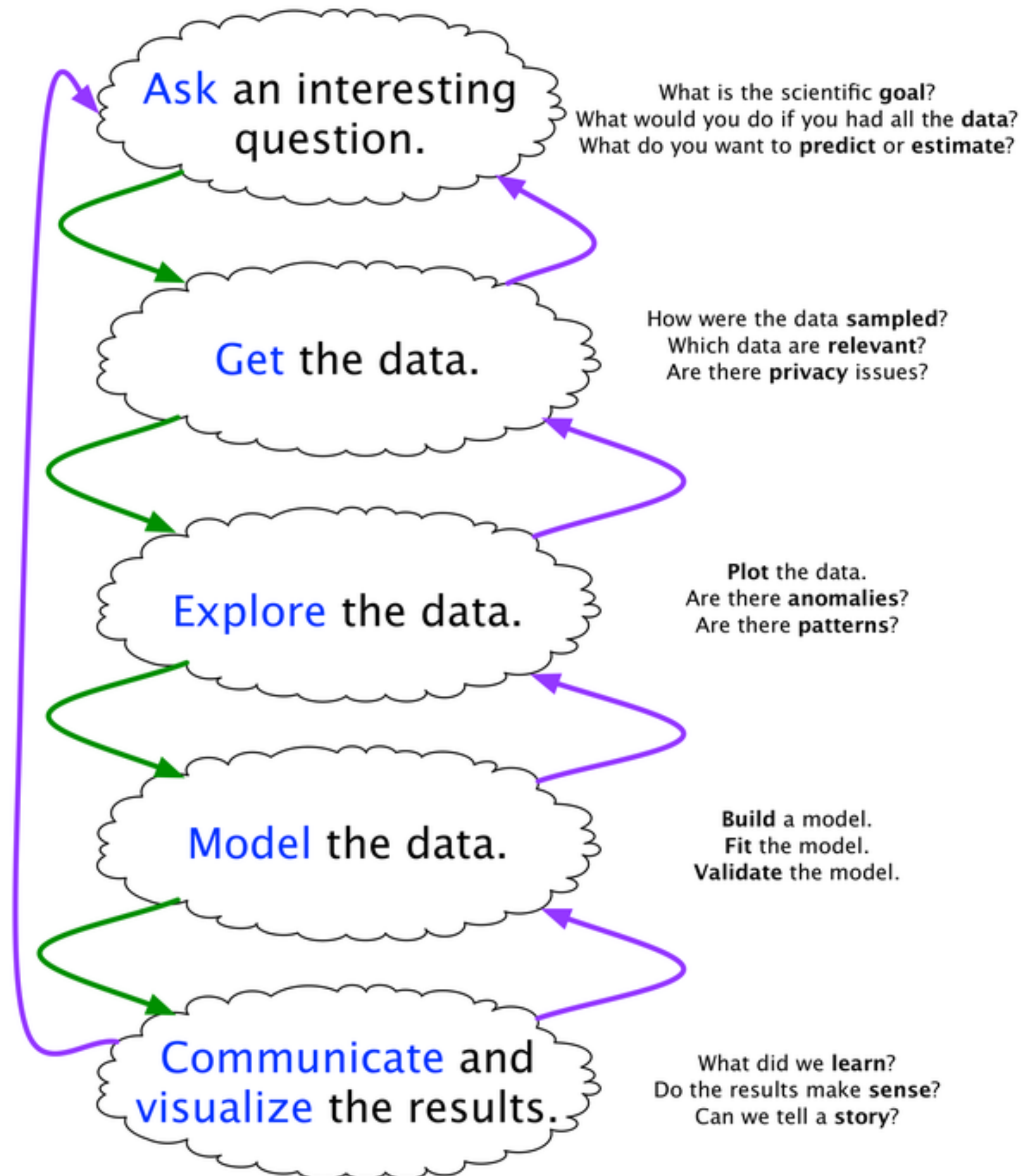


Iris Setosa



Iris Virginica


The Data Science Process



Pegar os dados

- Carregue com Pandas

UCI



Machine Learning Repository

Center for Machine Learning and Intelligent Systems

[About](#) [Citation Policy](#) [Donate a Data Set](#) [Contact](#)

Search

☒ Repository ☐ Web

Google™

[View ALL Data Sets](#)

Iris Data Set

Download: [Data Folder](#), [Data Set Description](#)

Abstract: Famous database; from Fisher, 1936



Data Set Characteristics:	Multivariate	Number of Instances:	150	Area:	Life
Attribute Characteristics:	Real	Number of Attributes:	4	Date Donated	1988-07-01
Associated Tasks:	Classification	Missing Values?	No	Number of Web Hits:	2567326

Source:

Creator:

R.A. Fisher

Donor:

Michael Marshall (MARSHALL%PLU%20io.arc.nasa.gov)

Explorar os dados

- Verifique o formato (shape)
- Veja algumas linhas dos dados
- Veja o sumário estatístico dos dados
- Veja como é a distribuição de classes entre os dados
- Plote histograma e a matriz de dispersão para fazer uma exploração visual

**Mais nas próximas etapas do
curso :)**

Referências e material complementar

[Introduction to Numpy for Data Analysis](#)

[A Gentle Introduction to Pandas](#)

[Your First Machine Learning Project in Python Step-By-Step](#)

[Principal Component Analysis in 3 Simple Steps](#)

[Kaggle Datasets](#)

[UCI Machine Learning Repository](#)