

# Données multimédia : Python pour le traitement d'images et de données audio

## Master Humanités Numériques du CESR

Clément Plancq (MSH VDL / CITERES)

- Pas un cours de traitement d'image ou de traitement du signal
  - [Supports de cours de Frédéric Legrand](#)
  - [Supports de \*Digital Image Processing\* \(Stanford\)](#)
- Pas un cours de python
  - vous avez déjà un cours de Python
  - [Mooc sur la plateforme FUN](#)
  - [Google's Python Class](#)
- Python appliqué au traitement d'images et de données audio

## Python appliqué au traitement d'images et de données audio

Logiciels avec interface graphique (GUI) :

Photoshop (Adobe)



Gimp



XnView



Audacity



|||≡ Ableton

Ableton

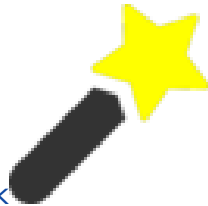


Garage Band (Apple)

## Python appliqué au traitement d'images et de données audio

Logiciels avec interface en ligne de commande (CLI) :

ffmpeg  FFmpeg



ImageMagick

## Python appliqué au traitement d'images et de données audio

Quels sont les avantages par rapport à un logiciel (GUI ou CLI) ?

1. Automatisation : en écrivant des scripts vous pouvez répéter des opérations (filtre, découpe, conversion) sur un grand nombre de fichiers sans intervention manuelle (les logiciels peuvent aussi faire du traitement par lot)
2. Personnalisation : vous avez accès à plus de paramètres pour manipuler les algorithmes (les filtres par exemple) et vous pouvez les combiner à votre guise
3. Intégration : vous pouvez intégrer des traitements dans une application, un site ou une chaîne de traitement

## Python appliqué au traitement d'images et de données audio

Quels sont les avantages par rapport à un logiciel (GUI ou CLI) ?

4. Performance : Python n'est pas réputé pour sa rapidité. Néanmoins un script bien conçu pourra être moins gourmand en ressource que l'utilisation d'un logiciel
5. Ressources communautaires : vous trouverez beaucoup de documentation, de tutoriels et d'aide auprès de la communauté d'utilisateurs Python
6. Portabilité : vos scripts Python fonctionneront sur les OS les plus courants (Linux, Mac OS, Windows)
7. Savoir : programmer vous aidera à appréhender les concepts de traitement du signal et vous vous améliorerez en Python 🐍

## Python appliqué au traitement d'images et de données audio

Et les inconvénients ?

1. C'est certainement plus compliqué et plus long d'apprendre à écrire un script que d'apprendre à utiliser un logiciel avec interface graphique
2. Le manque de visualisation est un inconvénient. Sans représentation graphique ou sonore c'est plus difficile de comprendre et d'améliorer ce qu'on fait
3. Vous pourrez partager vos scripts mais ils n'auront pas d'interface graphique. Et les seules visualisations seront celles que vous aurez écrites

Un exemple simple de traitement d'image qu'on peut faire avec Python : flouter une partie d'image

```
In [1]: from IPython.display import Image, display

image_url = 'https://cesr.univ-tours.fr/medias/photo/cpr-1130x400_1521210968'
# une image de https://cesr.univ-tours.fr/centre-detudes-superieures-de-la-r
display(Image(url=image_url))
```



Pour flouter on va utiliser le module [Pillow](#)

Pour simplifier on va travailler avec un fichier image et pas une url

```
In [2]: # import Pillow library
        from PIL import Image, ImageFilter

        image = Image.open('../img/CPR_1130x400.jpg') # 'image' est un objet de la classe Image
        display(image)
```

Une fois en mémoire on peut appliquer le filtre *blur* sur une partie de l'image  
(voir la [doc](#), toujours voir la doc)

```
In [3]: box = (185, 25, 335, 175) # on définit la partie à flouter avec ses coordonnées
        region = image.crop(box) # hop on découpe la partie
        blurred_region = region.filter(ImageFilter.BoxBlur(10)) # Blur pas 0
        image.paste(blurred_region, box) # on copie la partie floutée

        display(image)
```



## Python

Le langage Python a été créé par Guido Van Rossum en 1991

C'est un langage interprété, de haut niveau, multiplateformes, sous licence libre  
(voir [le code source](#))



Python	C
1991	1972
interprété	compilé
multiplateforme	non
haut niveau	bas niveau
typage dynamique	typage statique
syntaxe lisible et simple à utiliser	syntaxe moins lisible et plus ardue
pas le plus rapide	parmi les plus rapides

Le noyau de Python est écrit en C

```
In [4]: import this
```



The Zen of Python, by Tim Peters

Beautiful is better than ugly.  
Explicit is better than implicit.  
Simple is better than complex.  
Complex is better than complicated.  
Flat is better than nested.  
Sparse is better than dense.  
Readability counts.  
Special cases aren't special enough to break the rules.  
Although practicality beats purity.  
Errors should never pass silently.  
Unless explicitly silenced.  
In the face of ambiguity, refuse the temptation to guess.  
There should be one-- and preferably only one --obvious way to do it.  
Although that way may not be obvious at first unless you're Dutch.  
Now is better than never.  
Although never is often better than *\*right\** now.  
If the implementation is hard to explain, it's a bad idea.  
If the implementation is easy to explain, it may be a good idea.  
Namespaces are one honking great idea -- let's do more of those!

## Python ?

En 2023 Python est indéniablement devenu un langage très populaire, très utilisé et beaucoup enseigné

<https://spectrum.ieee.org/the-top-programming-languages-2023>

 top IEEE 2023

Ce qui a fait le succès de Python, amha, c'est la qualité et la diversité des bibliothèques (*libraries*) et des packages de la communauté Open Source

En particulier dans les domaines suivants :

- Calcul scientifique
- Manipulations et visualisation de données / *data science*
- *Machine learning*
- Traitement Automatique des Langues (TAL) / *NLP*
- Développement web
- Traitement d'images et de données audio 😊

## Calcul scientifique

- **NumPy** (Numerical Python)  
Manipulation de tableaux à n dimensions (matrices)  
Algèbre linéaire, calcul matriciel

- [SciPy](#)  
Implémentation d'algorithmes pour le calcul scientifique (dont le traitement du signal)  
Repose sur NumPy

Ces deux bibliothèques sont écrites en Python, C/C++ et Fortran

## Manipulations et visualisations de données

- [Pandas](#)  
Manipulation et analyse de données tabulaires
- [Matplotlib](#)  
Visualisations de données  
Combinée avec NumPy et SciPy, cette suite concurrence Matlab
- [Seaborn](#)  
Visualisations de données

## *Machine learning*

- [scikit-learn](#)  
Propose l'implémentation de beaucoup de méthodes d'apprentissage automatique (classification, clustering, ...)
- [TensorFlow](#)
- [PyTorch](#)
- [Keras](#) Deep Learning pour ces 3 dernières
- [Hugging Face](#)

## TAL / NLP

- [NLTK](#)
- [Stanza](#)
- [Spacy](#)

Traitement de données langagières, des textes surtout, avec reconnaissance des parties du discours, des entités nommées, analyse syntaxe, analyse de sentiment, ...

```
In [5]: !python3 -m spacy download fr_core_news_md
```

Defaulting to user installation because normal site-packages is not writeable

Collecting fr-core-news-md==3.6.0

Downloading [https://github.com/explosion/spacy-models/releases/download/fr\\_core\\_news\\_md-3.6.0/fr\\_core\\_news\\_md-3.6.0-py3-none-any.whl](https://github.com/explosion/spacy-models/releases/download/fr_core_news_md-3.6.0/fr_core_news_md-3.6.0-py3-none-any.whl) (45.8 MB)

45.8/45.8 MB 17.1 MB/s eta 0:0

0:00m eta 0:00:01[36m0:00:01

Requirement already satisfied: spacy<3.7.0,>=3.6.0 in /home/plancq/.local/lib/python3.10/site-packages (from fr-core-news-md==3.6.0) (3.6.1)

Requirement already satisfied: spacy-legacy<3.1.0,>=3.0.11 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (3.0.12)

Requirement already satisfied: pathy>=0.10.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (0.10.2)

Requirement already satisfied: tqdm<5.0.0,>=4.38.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (4.65.0)

Requirement already satisfied: pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.3.0)

Requirement already satisfied: jinja2 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (3.1.2)

Requirement already satisfied: catalogue<2.1.0,>=2.0.6 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.0.9)

Requirement already satisfied: cymem<2.1.0,>=2.0.2 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.0.7)

Requirement already satisfied: packaging>=20.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (21.3)

Requirement already satisfied: typer<0.10.0,>=0.3.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (0.9.0)

Requirement already satisfied: preshed<3.1.0,>=3.0.2 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (3.0.8)

Requirement already satisfied: setuptools in /usr/lib/python3/dist-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (59.6.0)

Requirement already satisfied: langcodes<4.0.0,>=3.2.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (3.3.0)

Requirement already satisfied: requests<3.0.0,>=2.13.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.31.0)

Requirement already satisfied: srsly<3.0.0,>=2.4.3 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.4.7)

Requirement already satisfied: numpy>=1.15.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (1.23.5)

Requirement already satisfied: smart-open<7.0.0,>=5.2.1 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (6.4.0)

Requirement already satisfied: wasabi<1.2.0,>=0.9.1 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (0.10.1)



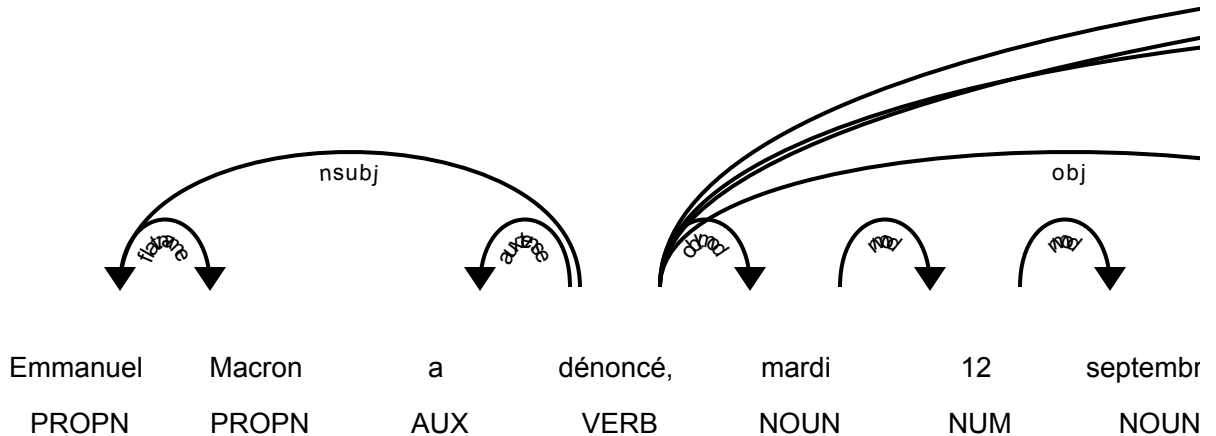
ib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (1.1.1)  
Requirement already satisfied: thinc<8.2.0,>=8.1.8 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (8.1.12)  
Requirement already satisfied: murmurhash<1.1.0,>=0.28.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (1.0.9)  
Requirement already satisfied: spacy-loggers<2.0.0,>=1.0.0 in /home/plancq/.local/lib/python3.10/site-packages (from spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (1.0.5)  
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /usr/lib/python3/dist-packages (from packaging>=20.0->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.4.7)  
Requirement already satisfied: pydantic-core==2.6.3 in /home/plancq/.local/lib/python3.10/site-packages (from pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.6.3)  
Requirement already satisfied: typing-extensions>=4.6.1 in /home/plancq/.local/lib/python3.10/site-packages (from pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (4.7.1)  
Requirement already satisfied: annotated-types>=0.4.0 in /home/plancq/.local/lib/python3.10/site-packages (from pydantic!=1.8,!1.8.1,<3.0.0,>=1.7.4->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (0.5.0)  
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/lib/python3/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (1.26.5)  
Requirement already satisfied: idna<4,>=2.5 in /usr/lib/python3/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (3.3)  
Requirement already satisfied: certifi>=2017.4.17 in /usr/lib/python3/dist-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2020.6.20)  
Requirement already satisfied: charset-normalizer<4,>=2 in /home/plancq/.local/lib/python3.10/site-packages (from requests<3.0.0,>=2.13.0->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (3.1.0)  
Requirement already satisfied: confection<1.0.0,>=0.0.1 in /home/plancq/.local/lib/python3.10/site-packages (from thinc<8.2.0,>=8.1.8->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (0.1.3)  
Requirement already satisfied: blis<0.8.0,>=0.7.8 in /home/plancq/.local/lib/python3.10/site-packages (from thinc<8.2.0,>=8.1.8->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (0.7.10)  
Requirement already satisfied: click<9.0.0,>=7.1.1 in /home/plancq/.local/lib/python3.10/site-packages (from typer<0.10.0,>=0.3.0->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (8.1.4)  
Requirement already satisfied: MarkupSafe>=2.0 in /home/plancq/.local/lib/python3.10/site-packages (from jinja2->spacy<3.7.0,>=3.6.0->fr-core-news-md==3.6.0) (2.1.2)  
✓ Download and installation successful  
You can now load the package via spacy.load('fr\_core\_news\_md')

```
In [2]: import spacy
        from spacy import displacy

        nlp = spacy.load('fr_core_news_md')
        doc = nlp("Emmanuel Macron a dénoncé, mardi 12 septembre, des « polémiques c
        displacy.render(doc, style="ent")
```

Emmanuel Macron **PER** a dénoncé, mardi 12 septembre, des « polémiques qui n'ont pas lieu d'être » sur la relation bilatérale entre la France **LOC** et le Maroc **LOC**, alors que Rabat **LOC** n'a pour l'heure pas retenu l'aide proposée par Paris **LOC**, après le séisme dévastateur dans la région de Marrakech **LOC**.

```
In [3]: displacy.render(doc, style="dep", jupyter=True, options={"distance":90})
```



## Développement web

- [Django](#)
- [Flask](#)
- [FastAPI](#)
- [BeautifulSoup](#)

Permet d'extraire des données depuis des pages HTML ou des fichiers XML

## Traitement d'images et de données audio

- Traitement d'images
- [Pillow](#)
- [OpenCV](#)

- Traitement de données audio
- [pydub](#)
- [librosa](#)