

# En route pour Jupyter

---

*Loïc Gouarin, Pierre Navaro*

<https://github.com/GroupeCalcul>

Vers un enseignement vivant

---

Interactivité

Partage facilité

# Un format unique

mais conversion dans de multiples formats  
possible

Vers une recherche reproductible

---

# Processus complet d'une simulation

Théorie

Code

Données

Visualisation



Faciliter les interactions

Mieux comprendre les méthodes

# Project Jupyter

## Les outils



Notebook

JupyterLab

Ipywidgets

Nbgrader

Nbconvert

JupyterHub

Binder

...

# Jupyter

## Les blocks



Navigateur de fichiers

Notebooks

Terminal

Éditeur de texte

Kernels

Output

# Jupyter

## Les kernels



Python

Julia

R

C++

# Jupyter

## Les kernels



Matlab

Scilab

Octave

Gnuplot

...

# Notebook

# Un environnement de calcul dans son navigateur

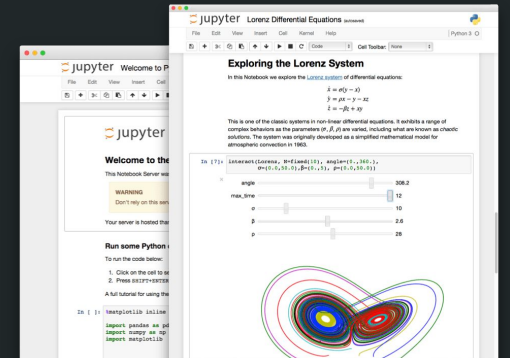
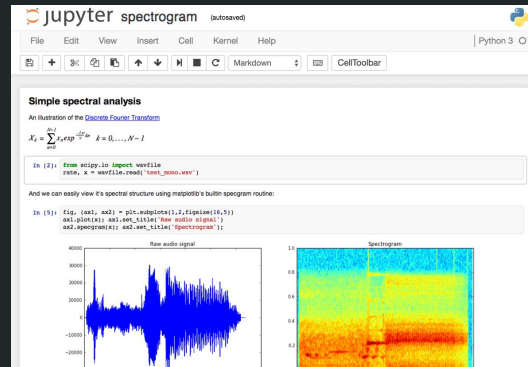


# Texte au format *markdown*

# Équations mathématiques

## Code

## Images et vidéos

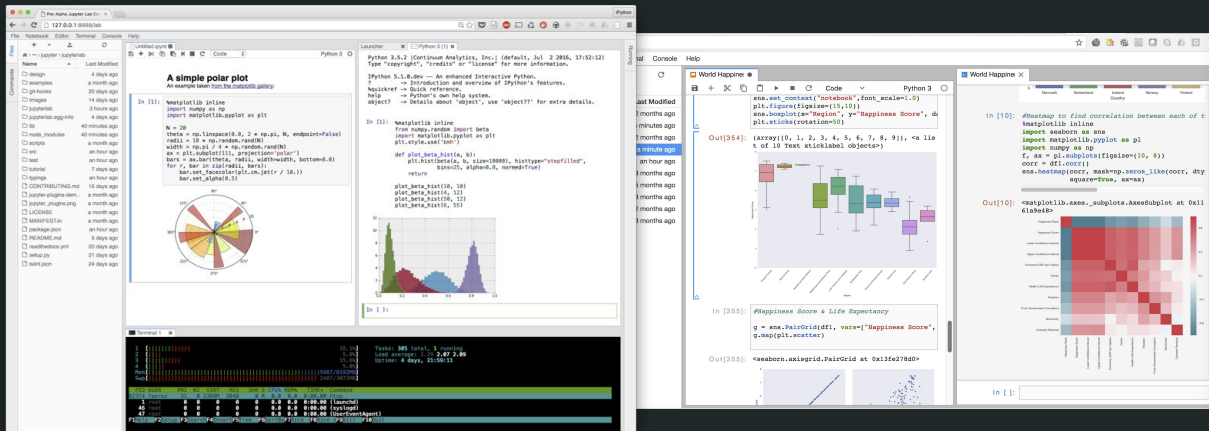


# JupyterLab

Environnement de  
travail enrichi



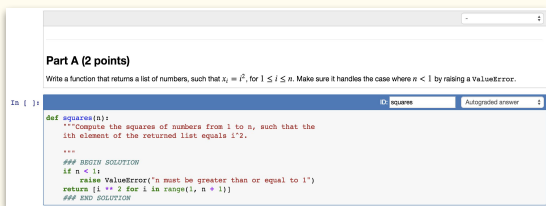
Interface plus conviviale  
Offre beaucoup plus de flexibilité  
Facilement modifiable





# Nbgrader

## Partage et évaluation pour son enseignement

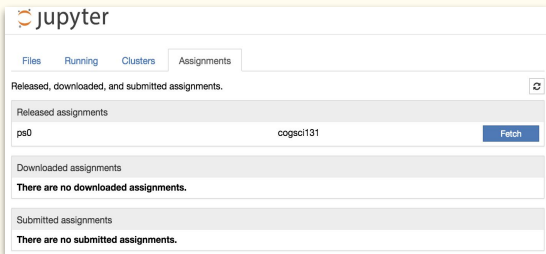


Part A (2 points)

Write a function that returns a list of numbers, such that  $x_i = i^2$ , for  $1 \leq i \leq n$ . Make sure it handles the case where  $n < 1$  by raising a `ValueError`.

```
In [ ]:
```

```
def squares(n):  
    """Compute the squares of numbers from 1 to n, such that the  
    1th element of the returned list equals 1^2.  
    """  
    ## BEGIN SOLUTION  
    if n < 1:  
        raise ValueError("n must be greater than or equal to 1")  
    return [i ** 2 for i in range(1, n + 1)]  
    ## END SOLUTION
```



Permet de définir des exercices

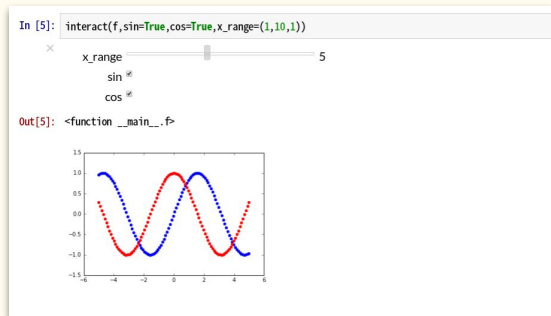
Correction automatique (avec notation)

Distribution des notebooks aux étudiants

Suivi des travaux des étudiants

# Ipywidgets

**Jouer avec ses  
paramètres**



Permet de voir rapidement l'influence des paramètres sur une simulation

Toute une collection de widgets prêts à l'emploi  
(slider, barre de progression, boîte à texte, bouton, ...)

Facilement extensible  
(bqplot, pythreejs, ...)

# Nbconvert

## Convertir ses notebooks

68

CHAPTER 2. DISCRETE BAYES FILTER

$x_t$  is the current position or state. If the dog is at 17 m then  $x_t = 17$ .

$f_t(\cdot)$  is the state propagation function for  $x_t$ . It describes how much the  $x_t$  changes over one time step. For our example it performs the computation  $15 \cdot 2$  so we would define it as

$$f_t(v_k, t) = v_k t$$

### 2.6 Adding Uncertainty to the Prediction

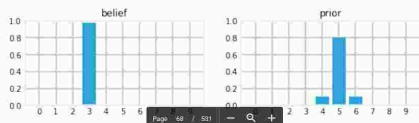
`perfect_predict()` assumes perfect measurements, but all sensors have noise. What if the sensor reported that our dog moved one space, but he actually moved two spaces, or zero? This may sound like an insurmountable problem, but let's model it and see what happens.

Assume that the sensor's movement measurement is 80% likely to be correct, 10% likely to overshoot one position to the right, and 10% likely to undershoot to the left. That is, if the movement measurement is 4 (meaning 4 spaces to the right), the dog is 80% likely to have moved 4 spaces to the right, 10% to have moved 3 spaces, and 10% to have moved 5 spaces.

Each result in the array now needs to incorporate probabilities for 3 different situations. For example, consider the reported movement of 2. If we are 100% certain the dog started from position 3, then there is an 80% chance he is at 5, and a 10% chance for either 4 or 6. Let's try coding that:

```
In [16]: def predict_move(belief, move, p_under, p_correct, p_over):
        n = len(belief)
        prior = np.zeros(n)
        for i in range(n):
            prior[i] = (
                belief[(i-move) % n] * p_correct +
                belief[(i-move-1) % n] * p_under +
                belief[(i-move+1) % n] * p_over)
        return prior

belief = [0., 0., 0., 1., 0., 0., 0., 0., 0., 0.]
prior = predict_move(belief, 2, .1, .8, .1)
book_plots.plot_belief_vs_prior(belief, prior)
```



Différents formats disponibles

(pdf, latex, html, ...)

Basé sur des templates

Possibilité d'exécuter les cellules du notebook lors de l'export

# JupyterHub

**Un serveur  
multi-utilisateurs pour  
les notebooks**

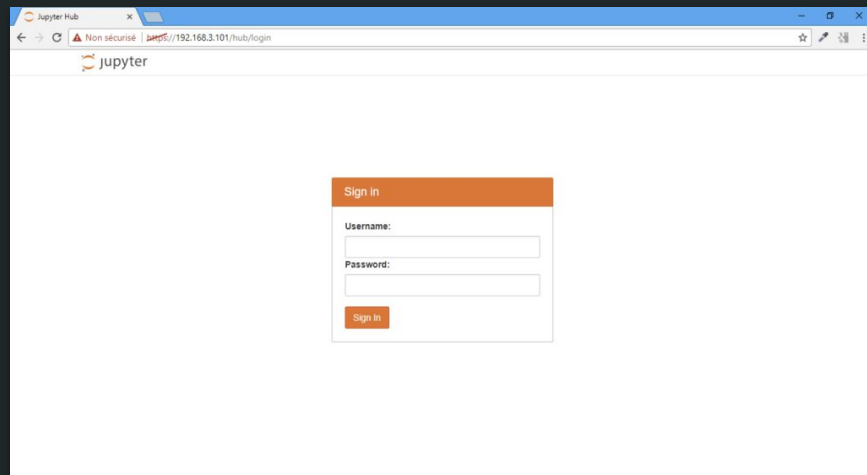


Serveur avec des images de base

Plus besoin d'avoir Jupyter sur son ordinateur

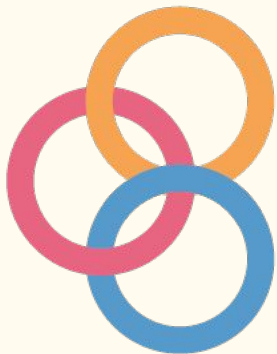
Fournit un environnement de travail homogène

<https://jupyter.math.cnrs.fr/hub/>



# BinderHub

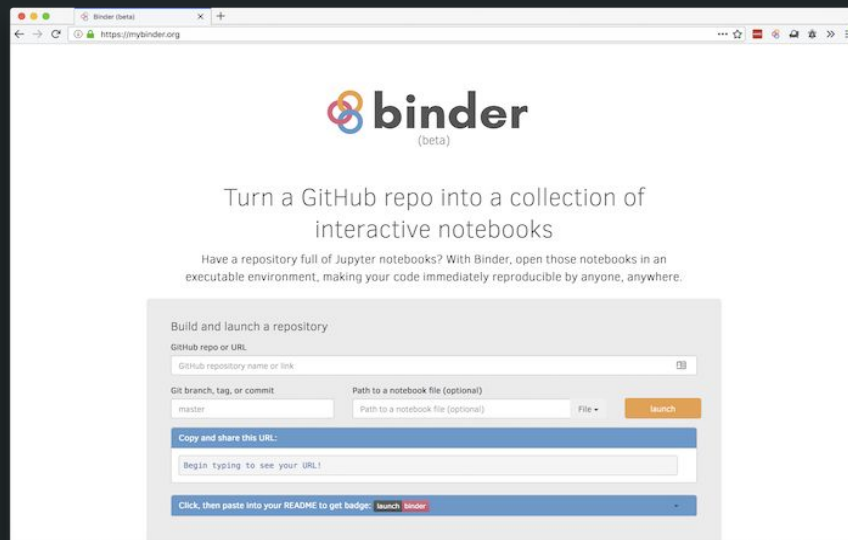
Créer ses propres  
environnements de  
calcul



Repo2docker + jupyterhub

Grande flexibilité dans la création d'environnement de recherche

Plus de liberté



Vers un BinderHub académique

---

# GT depuis quelques mois

Plusieurs groupes et universités intéressés  
(groupe Calcul, Mathrice, Polytechnique, Gricad, Romeo, ...)

# Objectifs

Mise en place d'une fédération de binder

Authentification nationale pour les chercheurs et les étudiants

Construction d'environnements de recherche à partir de gitlab privé

Pérennisation des données

Utilisation des centres de calcul régionaux



# Merci pour votre attention

Questions ?