

# *Mini-symposium Python*

## *Du calcul symbolique à l'apprentissage, à bord du Python-Express*

*Mini-symposium porté par le GdR Calcul*

### Résumé

L'objectif de ce mini-symposium est de donner un aperçu des fonctionnalités offertes par l'écosystème des logiciels libres autour de Python pour les mathématiques pures et appliquées : modélisation, calcul interactif et haute performance, visualisation, traitement de données, le tout accessible depuis un environnement interactif unifié.

Ce mini-symposium sera suivi dans la soirée par un tutoriel informel, avec aide à l'installation, etc. Les jours suivants les organisateurs seront à la disposition des participants pour une aide plus personnalisée.

### Organisateur(s)

1. **Paul Zimmermann**, Inria Nancy - Grand Est and LORIA.
2. **Nicolas M. Thiéry**, Laboratoire de Recherche en Informatique, Université Paris Sud.

### Liste des orateurs

1. **Maximilian Albert**, Computational Modelling Group, University of Southampton  
*Titre* : De la modélisation à la visualisation avec Jupyter/IPython.
2. **Nicolas M. Thiéry**, Laboratoire de Recherche en Informatique, Université Paris Sud  
*Titre* : Algèbre et mathématique discrètes avec Sage.
3. **Loïc Gouarin**, Laboratoire de Mathématique d'Orsay, Université Paris Sud  
*Titre* : Quelques idées d'optimisation d'un code numérique écrit en Python.
4. **Fabien Pedegrosa**, INRIA Saclay-Île de France  
*Titre* : Apprentissage statistique avec scikit-learn.

**Paul Zimmermann**, [www.loria.fr/~zimmerma/](http://www.loria.fr/~zimmerma/), [Paul.Zimmermann@loria.fr](mailto:Paul.Zimmermann@loria.fr)

**Maximilian Albert**, <http://cmg.soton.ac.uk/people/mha2e09/>, [maximilian.albert@gmail.com](mailto:maximilian.albert@gmail.com)

**Nicolas M. Thiéry**, [Nicolas.Thiery.name](mailto:Nicolas.Thiery.name), [Nicolas.Thiery@u-psud.fr](mailto:Nicolas.Thiery@u-psud.fr)

**Loïc Gouarin**, <http://www.math.u-psud.fr/~gouarin/>, [Loic.Gouarin@math.u-psud.fr](mailto:Loic.Gouarin@math.u-psud.fr)

**Fabien Pedegrosa**, <https://www.mendeley.com/profiles/fabian-pedregosa/>, [f\(at\)bianp.net](mailto:f(at)bianp.net)

# 1 De la modélisation à la visualisation avec Jupyter/IPython

Bla bla.

# 2 Titre de la deuxième section

Bla bla.

# 3 Quelques idées d’optimisation d’un code numérique écrit en Python

Le langage Python est maintenant largement utilisé au sein de la communauté du calcul numérique en s’appuyant fortement sur le module NumPy qui permet de gérer les tableaux. Bien que ce module soit optimisé pour les opérations élémentaires (addition de deux vecteurs, multiplication de matrices, ...), il est souvent nécessaire d’améliorer certaines parties du code avec d’autres outils afin d’obtenir des performances proches d’un langage bas niveau (C ou Fortran). Une large palette est offerte aux développeurs pour optimiser leur code : Cython, Numba, PyPy, Pythran, ... La plupart de ces outils utilise de la compilation "just in time" permettant de générer du code optimisé à l’exécution du programme. Leur syntaxe est très simple et ils peuvent être utilisés sans connaissances particulières. Lors de cette présentation, nous aborderons quelques uns de ces outils ainsi que leurs utilisations sur un cas concret : l’implantation d’une méthode lattice Boltzmann appelée  $D_2Q_9$  pour résoudre les équations de Saint Venant.

# Apprentissage statistique avec scikit-learn

Bla bla.