

Conférence "Python Scientifique" du 14 juin 2017

Conférence "Python Scientifique" Le 14 juin 2017, Amphi 200 PCMR En savoir plus
Conférence "Python Scientifique" Université de La Rochelle - 14 Juin 2017

Le 14 juin prochain, le laboratoire MIA et l'Université de la Rochelle organisent une journée d'échanges ouverte au public et à la communauté scientifique de la Rochelle (enseignants-chercheur(e)s, doctorant(e)s, ingénieur(e)s, ...). L'objectif de cette journée pluri-disciplinaire est de faire partager des pratiques (plus ou moins scientifiques) concernant la manipulation, la visualisation, le calcul et le traitement de données avec un focus autour du langage Python. Elle se tiendra dans l'amphithéâtre du Pôle Communication Multimédia Réseaux, 44 Avenue Albert Einstein, sur le campus des Minimes à la Rochelle. La journée débutera à 9h et se terminera aux alentours de 17h30. Nous accueillerons un certain nombre d'invités:

- Nicolas Rougier (INRIA, Bordeaux) sur la visualisation scientifique en Python
- Virginie Steiner (Ville de La Rochelle) sur l'ouverture et exploitation des données hétérogènes.
- Hassan Bouia (EDF R&D) sur le développement de codes pour l'énergétique du bâtiment.
- Vincent Dejouty (Quantmetry) Un exposé sur la reproductibilité des résultats et la gestion des tâches d'exécution dans un programme.
- Bruno Sciolla (Creatis, Lyon) sur l'analyse d'images bio-médicales 3D via le deep-learning.

D'autres thématiques seront abordées dans la journée :

- Python et le Web des données
- Python et les données géospatiales
- Python et le calcul scientifique (Méthodes d'optimisation évolutionnaire, analyses de sensibilité (Sobol, Morris,...), plans d'expérience
- Python et le calcul intensif (parallélisation, GPU, ...)
- Python et l'analyse d'images 2D/3D. Vous pouvez nous proposer une communication (20 min) sur le thème de votre choix avant le 15 mai si possible. Afin de recenser les personnes intéressés par cette journée, veuillez librement vous inscrire en remplissant le formulaire.

<http://obs-lienss.prod.lamp.cnrs.fr/limesurvey/index.php/726916/lang-fr> Nous espérons vous voir nombreux pour échanger autour de nos pratiques et s'enrichir mutuellement. {mospagebreak title=Programme} Lien vers les présentations disponibles Programme

- 9h : Accueil et ouverture
- 9h15-10h Nicolas Rougier (INRIA, Bordeaux) : Visualisation Scientifique avec Python
- 10h-10h45 Vincent Dejouty (Quantmetry) : Reproductibilité des résultats et optimisation des performances dans Numpy/Pandas

Pause (15 mins)

- 11h-11h30 Virginie Steiner (Ville de La Rochelle) : Ouvrir ses données, pourquoi, pour qui et comment ? Le projet de la Ville de La Rochelle : portail, collaborations, expérimentations
- 11h30-11h50 Christine Plumejeaud (LIENSs): Python et le web des données
- 11h50-12h10 Pierre Marige (BU): Python et le bibliothécaire standard

Puis discussions avec le public.

Pause déjeuner

- 14h00-14h30 Mathias Bouquerel (EDF R&D) sur le développement de codes pour l'énergétique du bâtiment.
- 14h30-14h50 Emmanuel Bozonnet (LaSIE): Optimisation multicritère des bâtiments
- 14h50-15h15 Erwan Liberge (LaSIE) : Un couteau suisse pour la modélisation en mécanique

Pause (15 mins)

- 15h30-16h15 Bruno Sciolla (Creatis, Lyon) : Analyse d'images médicales 3D basée sur le deep-learning (et tensorflow)
- 16h15-16h35 Jérôme Lux (LaSIE): Python pour l'analyse, le traitement et la visualisation d'images 2D/3D
- 16h35-17h00 Christophe Saint-Jean (MIA): Une expérimentation autour de l'exploitation de données images annotées dans une architecture client-serveur.
- 17h00-17h20 Arnaud Revel (L3i): Programmation du robot Nao : de la programmation graphique à la programmation Python

puis discussions avec le public.

17h30: Fermeture Lien vers les présentations disponibles

{mospagebreak title=Résumés} Résumé des exposés Nicolas Rougier : Visualisation scientifique avec Python Python offre une large gamme d'outils et de libraries permettant de réaliser des visualisations scientifiques statiques ou dynamiques de très bonnes qualités. Parmi ces outils, il y a la librairie matplotlib qui est devenue en quelques années le standard de la visualisation avec Python car on peut (presque) tout faire avec. Encore faut-il la maîtriser. Au travers de

quelques exemples, nous verrons comment exploiter au mieux les capacités de cette librairie et vers quoi se tourner lorsqu'elle ne suffit pas. Enfin, nous passerons en revue quelques règles élémentaires pour réaliser de bonnes figures à partir de l'article "Ten simple rules for better figures". Vincent Dejoux : Reproductibilité des calculs et optimisations de performances dans Numpy/Pandas Dans un projet de Data Science nous sommes souvent amenés à traiter de multiples sources de données sur lesquelles nous effectuons de nombreux traitements. Il est alors parfois laborieux de retrouver quelles unités de traitement ré-exécuter lorsqu'une ou plusieurs de ces sources changent. Certains outils disponibles permettent de spécifier des graphes d'exécution dans lesquels sont précisés les différentes dépendances entre les unités de traitement et leurs données d'entrée et de sortie. Néanmoins ces outils demandent de décrire manuellement ce graphe d'exécution. Du coup avec des collègues de Quantmetry, nous avons développé une solution permettant de résoudre ce problème à partir de composants existants: Dask et Joblib. Le résultat est un outil permettant de décrire des graphes d'exécution de manière automatique en analysant le code.

Dans une deuxième partie de l'exposé je vous propose un exposé sur deux librairies très largement utilisées dans le monde de la Data Science, Numpy et Pandas. Bien qu'il ne soit pas toujours utile de connaître les détails techniques de l'implémentation pour utiliser ces librairies, afin d'effectuer de manière optimale des calculs sur des volumes de données importants il est important de savoir comment ces librairies organisent les données en mémoire. Dans ce court exposé je vais vous présenter les différentes façons de disposer les données en mémoire et comment cette disposition peut impacter les performances pour des calculs usuels. Virginie Steiner: Ouvrir ses données, pour quoi, pour qui et comment ? Le projet de la Ville de La Rochelle : portail, collaborations, expérimentations à compléter Christine Plumejeaud : Python et le Web de données Dans un monde de données toujours plus interconnectées, le langage de programmation Python offre une API légère et facile à déployer qui nous permet de développer rapidement des services Web pour consommer ou produire des données interconnectées. L'exploitation d'un dictionnaire d'espèces en ligne (le registre mondiale d'espèce marines - WORMS sur <http://www.marinespecies.org/>) aidant à la normalisation de données de pêche constitue notre premier exemple. La mise en place d'un service de cartographie statistique dynamique et interactif sur le suivi de données de biodiversité constitue notre second exemple. Python offre pléthore de frameworks Web, mais il ne sera ici question que de WSGI (Web Server Gateway Interface). Pierre Marige : Python et le bibliothécaire standard Les bibliothécaires sont amenés à effectuer de nombreuses tâches répétitives impliquant des listes de données, manipulations pour lesquelles les outils bureautiques et le temps humain montrent vite leurs limites. Des scripts Python nous aident à exploiter des données (extraites d'un catalogue ou d'autres sources), les manipuler, les enrichir par des ressources web, et les restituer sous une forme exploitable par les bibliothécaires. La polyvalence du langage nous offre de nombreuses fonctionnalités (courriels automatiques, vérification de liens, applications graphiques, …) ; restent à définir les formats de données les plus pertinents pour nos différents usages. Mathias Bouquerel: à compléter Emmanuel Bozonnet: Optimisation multicritère des bâtiments La méthode par algorithme génétique NSGA-II utilisée (Deb et al., 2002) est une méthode qui a largement été utilisée pour la conception des bâtiments. La librairie python DEAP peut être utilisée. Les paramètres de l'optimisation doivent être déterminés par une étude de convergence préalable et les paramètres de conception à optimiser sont préalablement sélectionnés sur la base d'une étude de sensibilité. Les principales méthodes d'analyse de sensibilité sont disponibles dans la librairie python SaLib. Erwan Liberge : Un couteau suisse pour la modélisation en mécanique L'exposé présentera des utilisations variées de python et de certaines de ses librairies dans un contexte de modélisation numérique en ingénierie. Il sera abordé son utilisation permettant de recycler des sous-programmes écrits dans un autre langage de programmation, son rôle de facilitateur lors de campagne d'études paramétriques, et de transcription d'un modèle mathématique vers un modèle numérique. Bruno Sciolla : Analyse d'images médicales 3D basée sur le deep learning (et tensorflow) Le deep learning regroupe un ensemble de méthodes qui ont établi de nouveaux standards de performance sur des problèmes de classification, de prédiction, de segmentation sémantique et de nombreux autres problèmes. Nous présentons brièvement les bases de la segmentation par réseaux de neurones. Puis nous illustrons une implémentation dans Tensorflow, une librairie libre dédiée à l'apprentissage machine et développée par Google. En particulier, Tensorflow a un système de représentation des calculs par graphes, de variables représentant des conteneurs. Ces variables sont ensuite alimentées et les résultats collectés après une évaluation paresseuse. Jérôme Lux : Python pour l'analyse, le traitement et la visualisation d'images 2D/3D L'essor de techniques d'imagerie comme la tomographie aux rayons X en sciences des matériaux ou l'imagerie par résonance magnétique en médecine est accompagné de besoins d'outils informatiques de traitement, d'analyse et de visualisation. Cet exposé vise à dresser un panorama (très incomplet) des bibliothèques python permettant de répondre à ces problématiques de manière la plus efficace possible. Le langage python permet en effet le développement rapide d'algorithmes complexes, grâce notamment aux nombreuses bibliothèques existantes (scipy, scikits, mahotas, itk, vtk/tvtk, opencv). L'inclusion de code inline en c++ peut également être envisagée afin d'améliorer la performance lorsque cela est nécessaire (weave, cython). Enfin des solutions pour la visualisation 2D/3D seront également présentées. Christophe Saint-Jean : Une expérimentation autour de l'exploitation de données images annotées dans une architecture client-serveur. Dans cet exposé, je montrerai la mise en oeuvre du modèle YOLO pour la détection et la reconnaissance d'objets dans des images. Du point de vue Python, on parlera des outils permettant de récolter et agréger des données de sources hétérogènes. Les bibliothèques évoquées sont (a priori) BeautifulSoup, pyzmq, keras, numpy, ... puis les générateurs et la programmation asynchrone. Suivant le temps restant, l'environnement de développement PyCharm sera également présenté et configuré pour une exécution à distance. Arnaud Revel: Programmation du robot Nao : de la programmation graphique à la programmation Python A compléter. Lien vers les présentations disponibles