

**Enseignants :**

- Joseph Salmon : [joseph.salmon@umontpellier.fr](mailto:joseph.salmon@umontpellier.fr),
- Pascal Azerad : [pascal.azerad@umontpellier.fr](mailto:pascal.azerad@umontpellier.fr)
- Florent Bascou : [florent.bascou@etu.umontpellier.fr](mailto:florent.bascou@etu.umontpellier.fr)

**Horaire de consultation (*office hours*) :**

- Jeudi 18h-19h, avec prise de rendez-vous par mail uniquement.

**Prérequis**

Les étudiants sont supposés connaître les bases de la théorie des probabilités et de l'algèbre linéaire.

**Probabilités :** Pour le démarrage en probabilité les ouvrages suivant sont en français [CGDM01], [Ouv08], [Ouv07], [GS01].

**Algèbre Linéaire :** Il faut être suffisamment à l'aise avec le calcul matriciel pour bien commencer avec les modèles linéaires et aller jusqu'aux modèles économétriques : [Sch05] et [Gv13] peuvent être un bon début.

Les classiques du genre sont [Gv13, HJ94] et [TB97] est aussi un bon point d'entrée. Pour des questions plus avancées, par exemple de majorization [MOA11] (utile pour les notions de répartitions de richesses, etc.) et les matrices bistochastiques [Bha97].

**Description du cours**

Ce cours traite du cadre de l'utilisation de Python pour le traitement et la visualisation de données. Nous nous intéresserons principalement aux méthodes basiques de programmation en Python ainsi qu'aux bibliothèques standards de méthodes numériques (`numpy`, `scipy`) de traitement de bases de données (`pandas`) et de visualisation (`matplotlib`, `seaborn`). Au delà des aspects de programmation, veillera aussi à introduire de bonnes pratiques numériques.

**Notation**

**TP Noté : 40% de la note finale ;** Date de mise en ligne : lundi 21 octobre, et à rendre sur le Moodle du cours pour le vendredi 25 octobre, 23h59. Rendu sous forme de `jupyter notebook` (un unique fichier au format `.ipynb`).

**Détails de la notation du TP (noté sur 20) :**

- Qualité des réponses aux questions : **14** pts
- Qualité de rédaction et d'orthographe : **1** pt
- Qualité des graphiques (légendes, couleurs) : **1** pt
- Style PEP8 valide : **2** pts
- Qualité d'écriture du code (nom de variable clair, commentaires utile, code synthétique, etc.) : **1** pt
- Notebook reproductible ( *i.e.*, *Restart & Run all* marche correctement) et absence de bug : **1** pt

**Contrôle continu : 60% note finale**; sur table; effectué le mardi 8 octobre (la séance du 15 octobre est supprimée)

**Bonus : 1 pt** supplémentaire sur **la note finale** (de la partie 1) peut être obtenu pour toute contribution à l'amélioration des cours (présentations, codes, etc.)

Contraintes :

- seule la première amélioration proposée pour un thème particulier est “rémunérée”
- déposer un fichier **.txt** (taille <10 ko) en créant une fiche dans la partie du Moodle intitulée “Bonus - Proposition d'améliorations”
- détailler précisément (ligne de code, page des présentations, etc.) l'amélioration proposée, ce qu'elle corrige et/ou améliore
- pour les fautes d'orthographe : proposer au moins 5 corrections par contribution
- chaque élève ne peut gagner que 2 points maximum de cette manière

**Thèmes des séances :**

Séance 1 : Introduction à Python : panorama, environnement, packages, premières commandes.

Séance 2 : Base de la programmation : aide, boucles, **enumerate**,

Séance 3 : Base de la programmation (suite) : fonctions, matrices (**numpy**) vectorisation

Séance 4 : Import/Export des données, notamment les images (Colormap, gestion des couleurs, format vectoriel et bitmap)

Séance 5 : Statistiques descriptives : types de données

Séance 6 : Histogrammes, boîtes à moustache, méthodes à noyaux, violon, vidéos (matplotlib.animate), données manquantes

Séance 7 : Contrôle continu

Il est aussi recommandé de suivre les vidéos de Jake VanderPlas :

<http://jakevdp.github.io/blog/2017/03/03/reproducible-data-analysis-in-jupyter/> au rythme d'une part semaine. Les vidéos 1; 2; 7; 8; 9 et 10 les plus pertinentes, les autres pouvant être laissées de côté dans un premier temps.

**Livres et Ressources en ligne complémentaires**

**Général** : Skiena, The algorithm design manual, 1998 (en anglais) [[Ski98](#)]

**Général** : Courant *et al.*, Informatique pour tous en classes préparatoires aux grandes écoles : Manuel d'algorithmique et programmation structurée avec Python, 2013, [[CdG+13](#)]

**Général / Science des données** : Guttag, Introduction to Computation and Programming, 2016 [[Gut16](#)] (en anglais)

**Science des données** : J. Van DerPlas, With Application to Understanding Data [[Van16](#)] (en anglais)

**Code et style** : Boswell et Foucher, The Art of Readable Code [[BF11](#)] (en anglais)

**Python** : <http://www.scipy-lectures.org/>

Visualisation (sous R) : <https://serialmentor.com/dataviz/>

Page web du cours :

Supports : [http://josephsalmon.eu/index.php?page=teaching\\_18\\_19&lang=fr](http://josephsalmon.eu/index.php?page=teaching_18_19&lang=fr)

Dépôts (Moodle) : <https://moodle.umontpellier.fr/enrol/index.php?id=5558>

## Références

- [BF11] D. Boswell and T. Foucher. *The Art of Readable Code*. O'Reilly Media, 2011.
- [Bha97] R. Bhatia. *Matrix analysis*, volume 169 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1997.
- [CdG<sup>+</sup>13] J. Courant, M. de Falco, S. Gonnord, J.-C. Filliâtre, S. Conchon, G. Dowek, and B. Wack. *Informatique pour tous en classes préparatoires aux grandes écoles : Manuel d'algorithmique et programmation structurée avec Python*. Eyrolles, 2013.
- [CGDM01] M. Cottrell, V. Genon-Catalot, C. Duhamel, and T. Meyre. *Exercices de probabilités, Licence - Master - Écoles d'ingénieur*. Cassini, 3<sup>e</sup> edition, 2001.
- [GS01] G. R. Grimmett and D. R. Stirzaker. *Probability and random processes*. Oxford University Press, New York, third edition, 2001.
- [Gut16] J. V. Guttag. *Introduction to Computation and Programming Using Python : With Application to Understanding Data*. MIT Press, 2016.
- [Gv13] G. H. Golub and C. F. van Loan. *Matrix computations*. Johns Hopkins University Press, Baltimore, MD, fourth edition, 2013.
- [HJ94] R. A. Horn and C. R. Johnson. *Topics in matrix analysis*. Cambridge University Press, Cambridge, 1994. Corrected reprint of the 1991 original.
- [MOA11] A. W. Marshall, I. Olkin, and B. C. Arnold. *Inequalities : theory of majorization and its applications*. Springer Series in Statistics. Springer, second edition, 2011.
- [Ouv07] J-Y. Ouvrard. *Probabilités : Tome 2, Licence - CAPES*. Enseignement des mathématiques. Cassini, 2 edition, 2007.
- [Ouv08] J-Y. Ouvrard. *Probabilités : Tome 1, Licence - CAPES*. Enseignement des mathématiques. Cassini, 2 edition, 2008.
- [Sch05] J. R. Schott. *Matrix analysis for statistics*. Wiley Series in Probability and Statistics. Wiley-Interscience [John Wiley & Sons], second edition, 2005.
- [Ski98] S. S. Skiena. *The algorithm design manual*, volume 1. Springer Science & Business Media, 1998.
- [TB97] L. N. Trefethen and D. III Bau. *Numerical linear algebra*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.
- [Van16] J. VanderPlas. *Python Data Science Handbook*. O'Reilly Media, 2016.