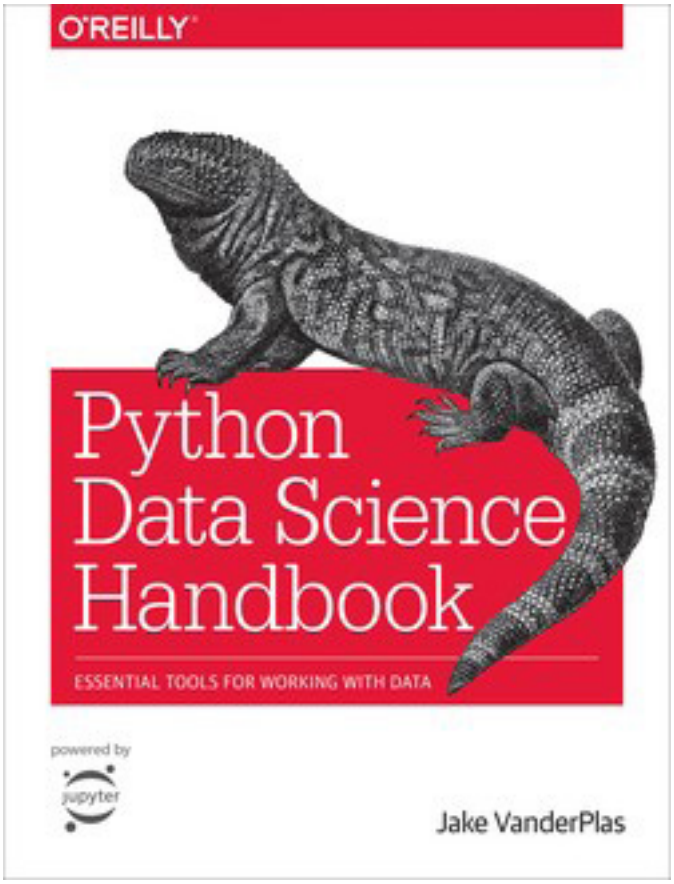
Python Data Science Handbook

by Jake VANDERPLAS (2016)

Alexis EIDELMAN[1]
Statisticien et data scientist public, administrateur hors classe de l'Insee



Livre (https://jakevdp.github.io/PythonDataScienceHandbook/)
Auteur : Jake VANDERPLAS
Édition : O'Reilly Media, Inc. – 2016
ISBN : 9781491912058

1. alexis.eidelman@travail.gouv.fr

DO YOU SPEAK PYTHON ?

La plupart du temps, le statisticien n'utilise qu'un seul langage de programmation. Même s'il connait parfois les rudiments de plusieurs d'entre eux, il s'est souvent spécialisé et n'en pratique qu'un seul. Pourtant, de la même manière que de plus en plus de personnes pratiquent plusieurs langues selon le contexte (familial, professionnel, voyages, etc.), le statisticien sera peut-être amené à utiliser plusieurs langages informatiques selon ce qu'il souhaite faire (collecte de données, traitement statistique, data visualisation, modélisation, etc.).

Un des langages intéressants aujourd'hui, à utiliser seul ou en complément d'autres, est le langage Python dont le livre *Python Data Science Handbook* constitue une référence pour initier le statisticien. Si la documentation des librairies et la communauté sur Python est conséquente et peut suffire, ce livre écrit par Jake VanderPlas, accessible librement[2], peut faciliter la prise en main et permettre de se « lancer » dans la découverte de Python.

Il est décomposé en quatre parties qui déroulent différents aspects du travail du statisticien/*datascientist*, plus une partie préliminaire consacrée à l'environnement Python utilisée dans le livre qui est essentielle. On pourra regretter toutefois que cette partie ne mentionne pas l'environnement Spyder qui est à mon sens l'environnement le plus pratique pour un usage « data » de Python.

Outre la partie sur les graphiques sur laquelle je ne m'étendrai pas, le livre présente les avantages de Python via trois entrées qui correspondent à trois niveaux de l'utilisation de Python et qui, sans peut-être le vouloir, correspondent à une chronologie du développement croissant de ce langage pour le traitement des données : calcul matriciel, manipulation de tableaux de données, *machine learning*.

Pour commencer, présentons en quelques mots le langage Python. Nommé en hommage aux Monty Python (et non pas en référence à un dangereux serpent), il se veut un langage généraliste mettant la lisibilité et la simplicité de sa syntaxe comme principe premier. Ce point, additionné au caractère *open source* du langage, lui a sans doute permis de devenir une référence pour le travail collaboratif. De ce fait, il a vu de nombreux *packages*/librairies se développer et est devenu grâce à eux un véritable couteau-suisse utilisé pour le traitement des données mais aussi pour les protocoles réseaux, pour des moteurs de recherches, pour les sites web et pour pratiquement tout ce que l'on peut faire avec un ordinateur. Le parti pris du *Python Data Science Handbook* est de ne pas s'attarder sur les bases du langage mais de renvoyer dans la préface à un ouvrage du même auteur qui permet d'apprendre pas à pas ses rudiments (fonctions, boucles, listes, etc.).

Langage de haut-niveau au sens informatique du terme, ce que Python gagne en simplicité/lisibilité il pourrait le perdre en performance par rapport à des langages comme C/C++, Fortran ou Perl. C'est sans compter sur différentes options permettant de retrouver cette puissance. A ce titre, un virage dans l'utilisation de Python pour le traitement de données a été le développement de Numpy, une librairie qui permet dans une syntaxe Python de faire du calcul matriciel avec la puissance de C. Cette librairie a permis à Python de se diffuser dans bon nombre de communautés scientifiques, en particulier dans l'univers de la physique. La partie de l'ouvrage qui présente Numpy décrit les bases de son utilisation et permet de comprendre comment accéder aux valeurs d'un tableau et la machinerie à l'œuvre lors des calculs, mais cette partie ne nécessite pas une lecture minutieuse car il n'est que très rarement nécessaire d'interagir directement avec cette librairie.

1. https://jakevdp.github.io/PythonDataScienceHandbook/

En effet, on se contente le plus souvent d'utiliser une autre librairie de Python, Pandas, qui utilise Numpy mais qui ajoute aux tableaux des noms de colonnes et prévoit des fonctions pour calculer des moyennes, pour regrouper les données selon une catégorie, etc. Cette librairie Pandas est « la » librairie de traitement de données statistiques. Le livre présente une introduction progressive assez proche de celle que l'on peut trouver dans la documentation de Pandas. On appréciera en particulier le passage sur les valeurs manquantes, les fusions de table ou encore sur les groupes. Cette partie comme le reste de l'ouvrage est très clair et mêle des textes d'explication et du code. Ce dernier peut d'ailleurs être exécuté depuis son ordinateur (le préliminaire l'explique), ce qui n'est pas sans vertu pédagogique.

Enfin, et c'est l'objet de la dernière partie, le langage Python est un langage de référence pour l'apprentissage automatique (*machine learning*). Développé par des communautés d'informaticiens, de statisticiens et d'experts en traitement d'image, l'apprentissage automatique s'est développé en Python et non dans des langages orientés vers les statistiques. Les librairies les plus utilisées du domaine sont écrites en Python et lorsque de nouveaux langages spécifiques se développent (Lua, Torch, TensorFlow, par exemple), ils sont très vite accompagnés d'une interface Python qui permet de garder la syntaxe et l'environnement Python tout en bénéficiant de la performance de ces langages dans leur domaine d'application. L'utilisation de Tensorflow est ainsi évoquée dans le livre peut-être trop rapidement pour les lecteurs qui souhaitent se pencher sur ce sujet. À sa décharge, l'ouvrage *Python Data Science Handbook* date de 2016, et s'il n'est pas obsolète et reste une excellente introduction, il ne couvre pas les progrès importants de ces dernières années dans le domaine du *machine learning*.

Le livre est un manuel de qualité mais s'adresse à des personnes qui ont déjà décidé d'apprendre le langage. Il ne répond pas – et ce n'est pas son rôle – à la question : quelles sont les qualités de Python qui pourraient conduire le statisticien à l'utiliser ? Une citation circule sur Python le présentant comme n'étant que le deuxième meilleur langage, mais dans tous les domaines. Je la trouve particulièrement pertinente. Certes, le statisticien spécialiste de son domaine préférera pour ses études utiliser le meilleur logiciel de statistique et il aura raison. Pourtant, dans bien des cas, le travail ne consiste pas seulement à ouvrir une base de données déjà nettoyée sur laquelle il n'y a plus que les calculs à réaliser et à présenter. Il est parfois – et même souvent – nécessaire de réaliser des opérations à la frontière et qui ne sont pas nécessairement statistiques. Un langage comme Python qui peut aisément gérer une chaine de traitement avec une gestion des messages d'erreur efficace, manipuler, convertir, mettre en ordre des données efficacement avec différents systèmes de stockage, nettoyer et gérer, par exemple, les chaines de caractères, réaliser des appels à des API ou à des sites web, faire de la représentation graphique, etc., devient alors un atout en particulier pour toute opération récurrente.

Python s'est développé loin du monde de la statistique et n'est devenu intéressant pour les statisticiens que récemment (l'indispensable librairie Pandas à moins de dix ans !). Puisque je tire le parallèle avec les langues parlées, je me risquerais à dire que Python est un peu l'anglais de la programmation. Parlé plus ou moins bien, il permet de faire le pont avec un écosystème très riche (informaticiens, physiciens, etc.) et de bénéficier de leurs apports via les librairies, mais aussi via les tutoriels ou forums d'échanges techniques. Ces éléments produits par des personnes du monde entier sont d'ailleurs pour beaucoup, comme le livre *Python Data Science Handbook*, en anglais.