Ultra R : Comment écrire du comment ultra efficient ?

Mohamed El Fodil Ihaddaden*

Résumé

Au sein de la communauté R, beaucoup pensent que le langage offre une flexibilité non égalée en termes de traitement des données et de visualisation. Par ailleurs, certains ont constaté que le langage n'était pas des plus efficients. A titre d'exemple, une rumeur très répandue dit que les boucles *for* sur R sont très lentes. Ce qui est absolument faux. De plus, la majorité des utilisateurs tendent à utiliser des packages offrant une grande simplicité d'utilisation au détriment de l'aspect optimisation. A titre d'exemple, les packages du tidyverse sont excellents. Ils offrent une syntaxe cohérente et simple à utiliser. Cependant, un package comme data.table, avec une syntaxe moins orthodoxe, disons-le, offre une optimisation des ressources beaucoup plus aboutie. Au travers de ma présentation, j'aurais comment objectif d'énumérer les techniques les plus efficaces afin de maintenir un programme *R* efficient tout en préservant sa robustesse.

Mots-clefs : programmation – data – Package

Développement

R est un langage de programmation très utilisée dans la recherche universitaire. En effet, il jouit d'une grande réputation dans le domaine de la statistique et de la data science en général. Cet état des lieux implique qu'une grande partie des utilisateurs *R* ne soit pas familiarisée aux concepts liés à l'ingénierie logiciel et de ce fait ne dispose pas des connaissances techniques leur permettant d'appréhender de manière optimale les fondamentaux du langage. A titre d'exemple, peu d'utilisateurs *R* peuvent réellement énoncer les différences techniques réelles entre un *vecteur* et une *liste* et cela a un impact non-négligeable sur la qualité du code qui sera implémenté par la suite.

De plus, nombreux utilisateurs *R* utilisent des formats de données tabulaires pour le transfert des données. L'un des plus répondu étant le *CSV*. Par ailleurs, l'écosystème *R* offre des formats beaucoup plus optimisés : A titre d'exemple on peut citer les formats *rds, parquet, fst* ou *qs*. Rien que le fait d'utiliser le format *qs* au lieu du *CSV* à titre d'exemple offre un boost de performance significatif, tout étant égal par ailleurs.

Ces dernières années, R a acquis une notoriété dans le domaine du traitement et de la visualisation des données, d'excellents packages comme *dplyr, tidyr ou ggplot2 avec* une syntaxe aboutie ont permis d'initier un grand nombre de personnes provenant de secteurs d'activité différents à l'analyse des données. Toutefois, un package beaucoup plus puissant reste méconnu, voir évité par la communauté à cause d'un manque de publicité et d'une syntaxe à première vue compliquée. Ce package n'est autre que *data.table*.

Au travers de mon talk, j'énoncerais les meilleurs pratiques pour écrire du R de manière efficiente et robuste et ce à travers des exemples et du live coding. Je m'efforcerais de répondre aux questions liées au thème énoncé tout en considérant les aspects exogènes qui doivent être pris en considération lors de l'implémentation des techniques recommandées.