Jupyter Notebook

https://www.jupyter.org	**Environnement interactif de programmation**	
	Système d'exploitation :	Multi-plateforme
	Développé par :	Jupyter community
3-Clause BSD License	Personne de contact :	Vandy.Berten@Smals.be

Fonctionnalités

Jupyter Notebook, projet Open Source issu de IPython, est une application web destinée à offrir un environnement interactif dans plusieurs dizaines de langages de programmation (dont, à l'origine, Julia, Python et R, d'où le nom JuPyteR), particulièrement adapté à l'analyse de données (data science).

Contrairement à la programmation traditionnelle (séquentielle), où on exécute un programme en démarrant toujours du même point d'entrée, puis en suivant le flux d'exécution défini par le programme, un « notebook » est un ensemble de blocs de code que l'on peut exécuter dans l'ordre désiré. L'ensemble de ces blocs partageront un contexte (appelé « kernel »), qui contiendra toutes les données.

On pourra par exemple avoir un premier bloc qui chargera un grand volume de données (avec un long temps d'exécution) dans un tableau T, puis un second qui calcule une statistique sur T, T étant stocké dans le « kernel ». On peut ensuite modifier ce second bloc, le ré-exécuter, sans devoir redémarrer le premier bloc de chargement, puisque T est toujours présent dans le kernel. L'output de chaque bloc apparait en dessous de celui-ci.

Combiné avec des librairies graphiques (matplotlib, ggplot…), des graphiques, interactifs ou non, peuvent être produits. Par ailleurs, un bloc peut contenir soit du code, soit du texte (markdown), permettant d'intégrer des commentaires (mise en page) et titres de sections du programme.

Une librairie d'extensions permet d'étendre les possibilités de l'environnement : création d'une table de matières, ajout de raccourcis clavier, intégration de LaTeX, « code prettifier », « code folding »…

Conclusions & Recommandations

Jupyter Notebook est un allié précieux de tout Data Scientist. Il offre une grande flexibilité dans son travail quotidien, particulièrement nécessaire à un mode de programmation très différent de la programmation traditionnelle (séquentielle ou évènementielle). Combiné avec JupyterHub, il permet de travailler dans des environnements multi-users.

Tests & Résultats

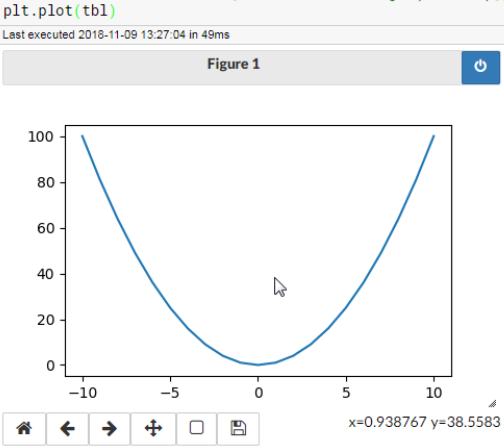
Dans l'exemple ci-joint, en Python, une variable « a » est créée dans le premier bloc. Quand ce bloc est exécuté (en cliquant sur le bouton adéquat dans la barre d'actions, ou via un « ctrl-enter »), la variable est disponible pour tous les autres blocs.

```
In [1]: a=5
        Last executed 2018-11-09 11:17:20 in 4ms

In [2]: a
        Last executed 2018-11-09 11:17:21 in 11ms
Out[2]: 5

In [3]: a += 1
        print(a)
        Last executed 2018-11-09 11:17:21 in 3ms
        6
```

L'output d'un bloc sera toujours la valeur de la dernière ligne, si celle-ci est définie (une assignation n'ayant pas de valeur en Python, le premier bloc n'a pas d'output), comme on peut le voir avec le 2ème bloc.

Tous les « prints » rencontrés dans un bloc s'affichent en dessous de celui-ci, comme avec le 3ème bloc. Notons que l'on peut maintenant ré-exécuter le 2ème bloc ; son output sera alors « 6 ». Bien qu'il soit possible d'exécuter séquentiellement l'ensemble des blocs (ce qui se fera en général quand le code a atteint un état stable), on sera très souvent amené à exécuter de multiple fois un bloc jusqu'à être satisfait de son contenu.

```
In [17]: import matplotlib.pyplot as plt
         import pandas as pd
         tbl = pd.DataFrame(index = range (-10, 11),
                            data = [x**2 for x in range (-10, 11)])
         plt.plot(tbl)
         Last executed 2018-11-09 13:27:04 in 49ms
```



En combinaison avec Pandas et Matplotlib, Jupyter (cf illustration ci-contre) offre un environnement puissant et flexible de Data Science pour Python.

L'approche « notebook » est similaire au mode de fonctionnement de RStudio, mais avec à notre sens plus de clarté :

- Dans RStudio, on exécute un ensemble de lignes sélectionnées, alors que dans Jupyter, on exécute l'ensemble d'un bloc clairement identifié ;
- Les outputs de RStudio sont affichés dans une fenêtre parallèle au code, alors qu'ils sont juste en dessous de chaque bloc dans Jupyter, ce qui permet de mieux faire le lien entre le code et son output.

Jupyter Notebook peut être installé localement (typiquement en installant Anaconda) sur un ordinateur personnel, ou sur un serveur, auquel cas il conviendra d'y installer également JupyterHub, permettant à plusieurs utilisateurs d'y accéder, chacun avec son propre environnement d'exécution et de fichiers.

Conditions d'utilisation & Budget

Jupyter Notebook est gratuit et Open Source. Il est nécessaire d'installer en parallèle les langages (Python, R, …), également gratuits.