



**HAL**  
open science

## Notebooks et science ouverte : FAIR mieux

Mariannig Le Béhec, Célya Gruson-Daniel, Clémence Lascombes, Émilien  
Schultz

► **To cite this version:**

Mariannig Le Béhec, Célya Gruson-Daniel, Clémence Lascombes, Émilien Schultz. Notebooks et science ouverte : FAIR mieux. 2024. hal-04485968

**HAL Id: hal-04485968**

**<https://hal.science/hal-04485968>**

Preprint submitted on 1 Mar 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Notebooks et science ouverte : FAIR mieux

Mariannig Le Béhec<sup>1,2\*</sup>, Célya Gruson-Daniel<sup>3,4</sup>, Clémence Lascombes<sup>3</sup>, Émilien Schultz<sup>5</sup>

1 ELICO, Université Claude Bernard Lyon 1

2 URFIST de Lyon - Université Claude Bernard Lyon 1

3 Inno<sup>3</sup>

4 COSTECH – Université de Technologie de Compiègne

5 CREST – ENSAE Paris

\*Adresse de contact : mariannig.le-behec@univ-lyon1.fr

**Attention ce document est un pré-print du 02/2024 avant soumission au Journal of Data Mining and Digital Humanities. En cas de citation, merci de vérifier la dernière version.**

## Résumé

Les *notebooks* sont aujourd’hui largement adoptés dans les pratiques numériques de recherche. Malgré leur omniprésence croissante, leurs caractéristiques, les rôles et usages associés aux *notebooks* ont pour le moment donné lieu à peu d’investigations dans une perspective d’études des sciences et des techniques (STS). Dans cet article, nous proposons une synthèse de travaux empiriques menés sur les *notebooks* afin d’identifier les principaux résultats existants, que cela concerne la classification de types de *notebooks*, les pratiques installées, les limites et améliorations proposées. Dans la continuité de cette synthèse qui souligne surtout l’existence de travaux dans le domaine de la science des données (*data science*) et non pas des pratiques de recherche en contextes académiques, nous discutons le rôle des *notebooks* comme vecteur et levier des principes FAIR<sup>1</sup> (*Findable, Accessible, Interoperable, Reusable*) associés à la science ouverte.

## Mots-clés

*notebook, literate programming, jupyter, open science, FAIR*

## INTRODUCTION

Largement adoptés dans les pratiques numériques de recherche, les *notebooks* sont entendus ici comme des supports permettant d’intégrer conjointement des éléments de contenu (notamment textuels), des éléments de programmation réalisant des traitements (code produits dans différents langages) et le résultat de ces traitements. Les *notebooks* regroupent diverses entités : documents d’abord écrits puis compilés pour obtenir des résultats, comme ceux permis par Rmarkdown<sup>2</sup>, ou documents directement interactifs comme ceux du projet

---

1 <https://force11.org/info/the-fair-data-principles/>

2 <https://rmarkdown.rstudio.com/>

Jupyter<sup>3</sup>, par la suite qualifiés de *notebooks* computationnels<sup>4</sup>. Ces documents se distinguent d'autres types de documents numériques aussi qualifiés de *notebooks*, comme les *electronic lab notebook*, qui font l'objet d'une littérature scientifique propre<sup>5</sup> et ne sont pas traités ici.

Malgré leur omniprésence croissante, spécialement dans l'enseignement de la programmation et du traitement des données, les caractéristiques, les rôles et les usages associés aux *notebooks* ont pour le moment donné lieu à peu d'investigations dans le domaine des études des sciences et des techniques (STS). Dans cet article, nous proposons une synthèse de travaux empiriques menés sur les *notebooks* afin d'identifier les principaux résultats existants, que cela concerne la classification de types de *notebooks*, les pratiques existantes, les limites de ces formats et les améliorations proposées.

Cette synthèse se base sur une recherche bibliographique menée sur différentes bases de données scientifiques (cf. [Méthodologie de la revue de littérature](#)), à partir de laquelle nous avons opéré une sélection d'articles. Une attention toute particulière a porté sur le repérage des pratiques liées à la science ouverte et des principes juridiques appliqués aux *notebooks*<sup>6</sup>.

Dans la continuité de cette synthèse, qui souligne surtout l'existence de travaux dans le domaine de la science des données (*data science*) et non pas des pratiques de recherche en contextes académiques, nous discutons le rôle des *notebooks* comme vecteur et levier des principes FAIR<sup>7</sup> (*Findable, Accessible, Interoperable, Reusable*) associés à la science ouverte. Nous ébauchons enfin quelques pistes d'étude à approfondir dans le cadre du projet NOOS (*Notebook for Open Science*).

## I. MÉTHODOLOGIE DE LA REVUE DE LITTÉRATURE

Constatant qu'il existe encore peu de travaux sur ce sujet encore en cours de stabilisation, notre objectif principal était d'identifier l'état actuel des connaissances produites autour des *notebooks* (2023). Notre synthèse est donc une « *scooping review* », non systématique, et se concentre de manière itérative sur les principaux travaux caractérisant le domaine.

Nous nous sommes concentrés sur des travaux empiriques concernant les *notebooks* computationnels. Différentes bases de données scientifiques ont été interrogées (Dimensions, BASE Api, Semantic Scholar) ainsi que la collection Zotero du Groupe de Travail Notebook<sup>8</sup>,

---

3 <https://jupyter.org/>

4 “A computational notebook consists of a sequence of cells, each containing code, prose, or media generated by the notebook’s computations (e.g., graphs), embodying a combination of literate programming and read-eval-print-loop (REPL) interaction.” (Liu et al., 2023)

5 Voir par exemple : Rabemanantsoa, Tovo, Dominique Pigeon, Nicolas Gilles Mathieu, Christophe Chipeaux, Simon Duvillard, Célya Gruson-Daniel, Marie Herbet, et al. 2021. « Report of the Working Group on Electronic Lab Notebooks ». Report, Comité pour la science ouverte. <https://doi.org/10.52949/30>.

6 Ce travail préliminaire à l'étude NOOS (*Notebook for Open Science*) — lauréat de l'appel à projet GIS Réseau URFIST 2022 — a permis de préciser la problématique et dessiner la méthodologie qui sera employée par la suite dans ce projet de recherche.

7 <https://force11.org/info/the-fair-data-principles/>

8 <https://www.zotero.org/groups/4416056/gt-notebooks/items/PGT9T72T/library>

collectif fondé en 2021. Les mots clefs et combinaisons employées étaient « *notebook* » OR « *interactive notebook* » OR « *computational notebook* » OR « *executable notebook* » OR « *jupyter notebook* » AND « *open science* ». «

Chaque lecture systématique des articles par les quatre auteurs et autrices (d’abord le résumé puis l’article) a permis de vérifier qu’il correspondait bien à une étude empirique sur les *notebooks*. Les réflexions générales sur les évolutions des outils ou du domaine, les descriptions de données pratiques en ingénierie logicielle n’ont pas été considérées. Nous avons ainsi sélectionné 12 articles, qui ont fait l’objet d’une étude comparative sur les méthodologies employées, les résultats principaux, les disciplines ou encore le type de *notebooks* ([cf. tableau annexe](#)).

## II. RÉSULTATS GÉNÉRAUX DE LA SYNTHÈSE

Nous revenons ici sur les principaux résultats identifiés par l’état de la littérature.

### 2.1 Une littérature récente ciblant surtout l’ingénierie logicielle et la data science

La littérature sur les *notebooks* est récente et se concentre surtout sur des enjeux liés à l’ingénierie informatique et logicielle ainsi que les interactions humains-logiciels. Le constat est que ces environnements logiciels sont utilisés à la fois par les acteurs de l’ingénierie informatique, les *data scientists* du secteur privé et les acteurs académiques. La distinction entre un usage à but industriel et un usage à but scientifique est parfois ténue et complexifie le travail de synthèse des pratiques.

Les enjeux liés à la recherche scientifique et académique sont abordés partiellement lorsque les auteurs et autrices proviennent de l’université (Quaranta et al., 2022). Cependant, dans le cas des entretiens avec des acteurs académiques, les spécificités disciplinaires sont complètement absentes (Quaranta et al., 2022; Rule et al., 2018). Les études ont souvent une approche solutionniste. Elles présentent alors des propositions de solutions technologiques pour résoudre les problèmes identifiés « *by having built-in, notebook-specific testing and linting frameworks as well as features for the refactoring and modularization of notebook code, we can expect data scientists to write high-quality notebooks without compromising on timeliness* » (Traduction libre : en disposant de cadres de test et de contrôle intégrés et spécifiques aux *notebooks*, ainsi que de fonctions de *refactoring* et de modélisation du code des *notebooks*, nous pouvons nous attendre à ce que les *data scientists* écrivent des *notebooks* de haute qualité sans compromettre la rapidité d’exécution)(Quaranta et al., 2022). Ainsi, les articles mettent en avant des solutions à désigner ou bien se réfèrent à des applications déjà existantes (Chattopadhyay et al., 2020; Kery et al., 2018).

Traversant plus généralement la littérature sur la programmation en recherche, la tension entre chercheur.se.s et ingénieur.e.s logiciel est cependant notée : « *Moreover, many notebook authors identify as scientists (e.g., chemists), and so may not have been exposed to concepts and skills related to reducing technical debt [...] and maintaining conceptual integrity, such as refactoring and software design principles* » (Traduction libre : De plus, de nombreux auteurs et autrices de *notebooks* s’identifient comme des scientifiques (par exemple, des chimistes) et n’ont donc peut-être pas été exposé.e.s aux concepts et aux compétences liées à la réduction de la dette technique [...] et au maintien de l’intégrité conceptuelle, tels que le *refactoring* et les principes de conception des logiciels) (Liu et al., 2023). Seules quelques questions spécifiques sont traitées comme la dissémination des résultats. Quelques articles

sont cependant dédiés à la recherche scientifique, mais ne soulignent que peu les particularités du domaine (Samuel & Mietchen, 2022; Wagemann et al., 2022).

Les articles se concentrent donc sur des pratiques de programmation, les considérations des acteurs sur les bonnes pratiques orientées vers un usage d'ingénierie logicielle ou encore les limitations de l'environnement logiciel associé aux *notebooks*. Les questions de l'intégration des *notebooks* dans une perspective plus générale des activités professionnelles, notamment dans la recherche scientifique, sont peu abordées, sauf avec (Samuel & Mietchen, 2022). Quelles places occupent les *notebooks* dans les laboratoires ? Existe-t-il une différenciation entre les disciplines ou les statuts d'utilisateurs et d'utilisatrices ? De même, les articles n'abordent pas les facteurs de diffusion des pratiques et des limites à l'adoption. Il n'existe par exemple aucune donnée sur la proportion d'utilisateurs et d'utilisatrices suivant les domaines professionnels ou les disciplines, ou encore des usages alternatifs. Il convient de souligner qu'une majorité des études a un caractère normatif tourné vers les bonnes pratiques à emprunter : « *All these studies suggest that to successfully support data science work — the use of notebooks should be disciplined and informed with shared best practices* » (Traduction libre : Toutes ces études suggèrent que pour soutenir avec succès le travail de science des données, l'utilisation des notebooks doit être structurée et s'appuyer sur des bonnes pratiques partagées) (Quaranta et al., 2022).

En termes de méthodologie d'étude, une partie importante des travaux se concentre sur l'analyse des traces numériques des *notebooks*. Celles-ci sont issues principalement d'une plateforme collaborative de gestion de versions de code, Github et une plateforme de compétition d'apprentissage automatique en sciences de données, Kaggle<sup>9</sup>, largement utilisées dans le milieu industriel de l'informatique. Cette démarche offre une étude approfondie de la structure des *notebooks* (Pimentel et al., 2021), de leur position dans les dépôts (Rule et al., 2018), mais aussi de leur histoire particulièrement avec les *commits*<sup>10</sup> (Raghunandan et al., 2023). La disponibilité des données numériques permet de définir différents indicateurs de qualité précis sur les *notebooks* eux-mêmes comme des métriques structurelles ou stylistiques du code (nombre de commentaires du code, nombre de fonctions, etc.) (Grotov et al., 2022). Pour autant, la question des échantillons utilisés et de leur représentativité se pose. Les auteur.rice.s, mais aussi les contextes de production des *notebooks*, sont investigués de manière limitée, à partir des informations disponibles sur les profils (Liu et al., 2023), ou via le proxy de la profession dans les entretiens. Les usages suivant le profil des personnes employant des *notebooks* sont peu différenciés.

En complément de l'analyse des données numériques, plusieurs travaux intègrent des entretiens semi-directifs auprès d'utilisateurs et utilisatrices de *notebooks*. Les pratiques sont questionnées ainsi que le jugement porté sur les *notebooks*. Néanmoins, les entretiens se limitent au périmètre des *notebooks* et ne portent pas sur le contexte et les activités plus larges de travail dans lesquelles les *notebooks* s'intègrent. Les aspects de socialisation aux outils et de formation ne sont aussi pas considérés. Une seule étude mentionne la passation d'un questionnaire (Chattopadhyay et al., 2020). Par ailleurs, dans le cadre de recherches visant à être opérationnelles en prototypant de nouveaux outils, des méthodes UX de tests sont mis en œuvre. L'observation en contexte est presque absente des méthodologies, mis à part les travaux de Chattopadhyay (ibid). Aucune étude ne met en perspective les *notebooks* dans une logique plus large d'archives et d'articulation avec d'autres documents et informations associées (données, métadonnées, licences, fichiers README, etc.).

---

9 Kaggle propose un système de médailles sur les *notebooks* et de rang pour les personnes : Novice, Contributor, Expert, Master et Grandmaster (Choetkiertikul and al. 2023)

10 Enregistrer les modifications dans le dépôt source: <https://git-scm.com/docs/git-commit/fr>

## 2.2 Les nouvelles pratiques des *notebooks* Jupyter

La littérature se concentre sur les *notebooks* computationnels et surtout sur ceux du projet Jupyter, qui sont les plus étudiés et les plus mentionnés. D'autres solutions de *notebooks* tels que RStudio, Sage, et leurs déclinaisons en plateforme (Microsoft Azure, Google Colab, Kaggle, Databricks, Apache Zeppelin) sont mentionnées, mais minoritaires.

Si la « reproductibilité » des résultats est la principale qualité recherchée et mise en avant dans plusieurs études (Pimentel et al., 2021; Samuel & Mietchen, 2022), les travaux montrent qu'elle est loin d'être atteinte dans la plupart des *notebooks* disponibles publiquement, notamment sur Github (cf. [L'identification de nombreuses limites et critiques](#)). La critique n'est cependant pas spécifique aux *notebooks* et concerne plus généralement le code (Trisovic et al., 2022)<sup>11</sup>. Les études montrent que la reproductibilité varie en fonction de différentes caractéristiques, par exemple la popularité des *notebooks*. Différents obstacles à la reproductibilité sont identifiés tels que la non-linéarité de l'organisation des *notebooks*, ou encore l'absence de documentation des bibliothèques utilisées, et de leur provenance, etc. (Ramasamy et al., 2023; J. Wang et al., 2020; Zhang et al., 2020). Par ailleurs, les *notebooks* computationnels introduisent de nouvelles contraintes, ce qui amène à complexifier les définitions sur la reproductibilité où il était uniquement indiqué la question de l'absence de code ou de données (Baker, 2016) ou la distinction disciplinaire (Leonelli, 2018). Aujourd'hui, il s'agit de considérer en plus dans ces définitions la documentation adéquate pour la mise à disposition du code, des données.

Les études soulignent également que les *notebooks* sont le lieu d'un relâchement des contraintes de systématisme des pratiques de programmation et des bonnes pratiques par ailleurs, stabilisées sur le code. Même si ces bonnes pratiques de programmation sont connues, les *notebooks* s'accompagnent d'une plus grande liberté prise et une tension constante entre qualité et rapidité de développement (Quaranta et al., 2022). Les *notebooks* se caractérisent alors par un style de programmation spécifique (Grotov et al., 2022) et l'introduction de nouvelles manières de documenter le code (A. Y. Wang et al., 2021). La perte de qualité s'explique, entre autres, par le fait que les *notebooks* sont aussi utilisés par des équipes pluridisciplinaires pouvant ajouter d'autres strates quant à la difficulté d'avoir une pratique de programmation et une documentation uniforme (A. Y. Wang et al., 2021).

Les études portant sur des échantillons très larges de *notebooks* notamment disponibles sur Github (Rule et al., 2018) permettent de montrer leur diversité et d'établir des classifications. Ils peuvent être au service de différents usages : matériaux d'enseignement ou de démonstration, étape dans une analyse de données, ou encore support d'entraînement. Les aspects pédagogiques apparaissent importants (Rule et al., 2018): sur 69 *notebooks* aléatoires, « 31 *notebooks* associated with courses, such as tutorials, class assignments, or course exercises » (Traduction libre : 31 *notebooks* associés aux enseignements, tels que les tutoriels,

---

11 L'article de Trisovic et al. pointe ainsi que plus de 74% des fichiers R ne peuvent être exécutés sans erreur dû à la qualité faible du code. « *We find that 74% of R files failed to complete without error in the initial execution, while 56% failed when code cleaning was applied, showing that many errors can be prevented with good coding practices.* » (Traduction libre : Nous constatons que 74 % des fichiers R échouent à se compiler sans erreur lors de l'exécution initiale, tandis que 56 % échouent lorsque le code est nettoyé, ce qui montre que de nombreuses erreurs peuvent être évitées grâce à de bonnes pratiques de codage) Trisovic, A., Lau, M.K., Pasquier, T. et al. A large-scale study on research code quality and execution. *Sci Data* 9, 60 (2022). <https://doi.org/10.1038/s41597-022-01143->

les devoirs de classe ou les exercices de cours) (Pimentel et al., 2021). La catégorisation de Liu and al. 2021 propose quant à elle cinq genres de *notebooks* :

- Analyse exploratoire (*Exploratory Analysis*), qui représente l’usage majoritaire;
- Exercice de programmation (*Programming Assignment*);
- Démonstration de la technologie (*Technology Demonstration*);
- Démonstration analytique (*Analytical Demonstration*)
- Matériel éducatif (*Educational Material*).

Les travaux soulignent largement le caractère non-linéaire et évolutif de ce type de document (Ramasamy et al., 2023). Deux pôles de pratiques (exploratoire et explicative) sont distingués : le premier pôle exploratoire éloigné des bonnes pratiques de l’ingénierie logicielle qui regroupe des tentatives, esquisses et morceaux de code et de résultats, et le second pôle explicatif qui correspond à des documents plus finalisés répondant à des contraintes plus strictes de narration (exposition de la démarche et de l’enchaînement des traitements) et de reproductibilité : “*These studies demonstrate a tension between exploration and explanation in constructing and sharing computational notebooks*” (Traduction libre : Ces études démontrent une tension entre l'exploration et l'explication dans la construction et le partage des *notebooks*) (Rule et al., 2018). Les travaux les plus récents se concentrent justement sur le caractère évolutif des *notebooks* qui peuvent avoir une trajectoire suivant leurs usages (Liu et al., 2023 ; Raghunandan et al., 2023). Il est donc indispensable de considérer les *notebooks* computationnels produits comme des entités hétérogènes et aller vers un travail plus fin de classification.

### 2.3 Bonnes pratiques conseillées

Une question centrale qui traverse la littérature est celle des « bonnes pratiques» (*best practices*), qui oscille entre une définition des méthodes efficaces et un objectif de qualité. À ce titre, de nombreux aspects se recoupent entre les articles, et une exhaustivité semble avoir été atteinte : « *it is worth noting that none of the interviewees mentioned best practices that had not been already identified through the literature review, thus increasing the confidence that we may have reached theoretical saturation, given the current state of the art and practice* » (Traduction libre : il convient de noter qu'aucune des personnes interrogées n'a mentionné de bonnes pratiques qui n'avaient pas déjà été identifiées dans le cadre de l'analyse documentaire, ce qui renforce la certitude que nous avons peut-être atteint la saturation théorique, compte tenu de l'état actuel de l'art et de la pratique) (Quaranta et al., 2022).

Le constat transversal est que ces bonnes pratiques sont connues, mais ne sont pas ou peu appliquées (Quaranta et al., 2022). Par exemple, Pimentel et al., 2021 listent 8 bonnes pratiques pour la reproductibilité des carnets. Il est possible de synthétiser ces recommandations des différents articles dans un tableau :

Catégorie générale	Bonnes pratiques
Rendre votre analyse traçable et reproductible	<ul style="list-style-type: none"> <li>• Utiliser un gestionnaire de révision</li> <li>• Gérer les dépendances du projet</li> <li>• Fournir des applications sans dépendances tierces</li> <li>• Placer les imports au début du fichier</li> <li>• Assurer le bon fonctionnement du code en complet et pas uniquement la partie modifiée</li> </ul>

<b>Écrire un code de qualité (i.e. un code qui peut être facilement partagé et réutilisé)</b>	<ul style="list-style-type: none"> <li>• Structurer votre code en modules (abstraire le code en fonctions et les placer dans un module dédié ; placer les dépendances au début du carnet)</li> <li>• Tester votre code</li> <li>• Nommer vos carnets de manière cohérente</li> <li>• Respecter les standards</li> <li>• Utilisez les chemins relatifs</li> <li>• Définir les <i>requirements</i></li> <li>• Documenter votre code pour vous et les autres</li> </ul>
<b>Exploiter le paradigme de la programmation lettrée</b>	<ul style="list-style-type: none"> <li>• Utiliser les titres Markdown pour structurer votre carnet</li> </ul>
<b>Garder votre carnet clair et concis</b>	<ul style="list-style-type: none"> <li>• Garder votre carnet clair</li> <li>• Garder votre carnet concis</li> </ul>
<b>Faire la différence entre les artefacts produits pendant le développement et la production</b>	<ul style="list-style-type: none"> <li>• Faire la différence entre les artefacts produits pendant le développement et la production</li> </ul>
<b>Adopter une diffusion ouverte</b>	<ul style="list-style-type: none"> <li>• Mettre à disposition votre carnet</li> <li>• Mettre à disposition vos données</li> </ul>

Tableau 1: Catalogue des bonnes pratiques spécifiques à l’usage des carnets extraits de la revue de littérature.

En plus de ces bonnes pratiques se pose l’intégration du *notebook* dans un environnement plus large. La dépendance de l’exécution des carnets à des environnements spécifiques n’est pas abordée directement : installation de logiciels, dépendances à certains environnements etc. Concernant la diffusion, Il est indiqué en complément d’ajouter un fichier README, de mettre une licence open source autorisant la réutilisation et de proposer des versions statiques et dynamiques des carnets qui ne nécessitent pas d’installation locale de Jupyter pour être lue ou exécutée (Quaranta et al., 2022). La notion de science ouverte et des travaux associés n’est pas présentée.

## 2.4 L’identification de nombreuses limites et critiques

Les *notebooks* semblent largement plébiscités et montrés en exemple, pour les aspects éducationnels, que ce soit la présentation de cours (Wagemann et al., 2022) ou des exercices et tutoriels (Pimentel et al., 2021). Pour autant, de nombreuses critiques leur sont adressées, et ce tout au long des actions et des interactions avec un *notebook* que ce soit du téléchargement des données, jusqu’à la dissémination ou collaboration sur un *notebook* (Chattopadhyay et al., 2020).

Ces difficultés rendent difficilement atteignables les attentes concernant les *notebooks* : reproductibilité, narration du code dans une dynamique de *literate programming* (Kery et al., 2018). Ainsi, comme le soulignent Wangeman et al. (2022), « *It seems that there is a misconception that content is automatically reproducible when made available as a Jupyter notebook* » (Traduction libre : Il semble qu’il y ait une fausse idée selon laquelle le contenu est automatiquement reproductible lorsqu’il est mis à disposition sous la forme d’un *notebook* Jupyter). Les critiques adressées aux *notebooks* sont souvent accompagnées de propositions d’améliorations ou d’un outillage pour combler les manquements (Quaranta et al., 2022; Ramasamy et al., 2023; Rule et al., 2018).

Un des points faibles majeurs, et qui touche plus spécifiquement les pratiques de recherche académique – concerne la question de la reproductibilité, qui s’avère être très faible dans les



études analysant un grand nombre de *notebooks*. Pimentel et al. (2021) montrent que seulement 14% des *notebooks* étudiés dans leurs travaux sont reproductibles.

Plusieurs raisons sont invoquées tout au long des articles :

- Le manque de documentation. Rule et al. (2018) indiquent que plus d'un quart des *notebooks* étudiés ne contient pas de documentation. Même si les *notebooks* sont employés dans un usage d'exploration et non d'explication et de narration des résultats, plusieurs éléments de documentation essentiels manquent telles que la liste des logiciels tiers employés (Ramasamy et al., 2023; A. Y. Wang et al., 2021) ou encore la provenance du code employé (Zhang et al., 2020). La démarche collaborative et les itérations nombreuses amplifieraient ce phénomène tout au long du cycle de vie du *notebook* (A. Y. Wang et al., 2021). Concernant la reproductibilité et la réutilisation, il est noté que « *Unfortunately, even seemingly simple reuse can become more complicated than expected, such as when the “earlier notebook uses absolute paths” (IP5), when the “cells have no designated format or function” (IP1) and can't be easily isolated, and when there are complex dependencies to bring into the new notebook* » (Traduction libre : Malheureusement, même une réutilisation apparemment simple peut s'avérer plus compliquée que prévu, par exemple lorsque le "*notebook* précédent utilise des chemins absolus" (IP5), lorsque les "cellules n'ont pas de format ou de fonction désignées" (IP1) et ne peuvent pas être facilement isolées, et lorsqu'il y a des dépendances complexes à intégrer dans le nouveau *notebook*). (Chattopadhyay et al., 2020);
- Le manque de cohérence dans l'exécution des cellules (Alena Guzharina & Alena Guzharina, 2020). La non-linéarité du *notebook* et la capacité d'exécuter certaines parties du *notebook* sans exécuter l'ensemble de haut en bas compliquent cette situation (avec un manque de transparence des mécanismes d'exécution);
- Le suivi de version. Dès 2015, un questionnaire UX, proposé par Jupyter, soulignait cette difficulté constante de ne pas pouvoir versionner les *notebooks*. Cela amène souvent à une multiplication des fichiers sans pouvoir faire un suivi des modifications apportées dans le temps (Kery et al., 2018);
- La qualité du code s'avère fréquemment de bas niveau (A. Y. Wang et al., 2021). Les « copier-coller » entre plusieurs *notebooks* et plus généralement une duplication de *notebooks* (Quaranta et al., 2022) participent à diffuser du code de basse qualité. La qualité plus faible du code serait liée à l'usage même du *notebook* comme outil exploratoire et non pas de démonstration ou présentation des résultats. La rapidité est préférée à la qualité et se rapproche de l'idée de « *messy notebook* » qui ressort dans plusieurs articles.

Par ailleurs, plusieurs critiques sont adressées sur le format des *notebooks* et les contraintes associées à l'usage de cette interface, amenant souvent à des erreurs ou la production d'un code de moindre qualité. Ainsi, plusieurs articles reviennent sur la lourdeur du *notebook* au-delà d'une certaine longueur, amenant fréquemment à des *crash* (Kery et al., 2018). Une autre critique est la difficulté de travailler avec des données massives ou encore celle de naviguer dans l'ensemble des documents par une structuration complexe et linéaire du code, sans pouvoir visualiser les étapes du *workflow* (Ramasamy et al., 2023). Le format des cellules ou bien les rendus de visualisation rendent parfois difficiles le maniement du *notebook* ou une dissémination effective avec des résultats peu appropriables (Chattopadhyay et al., 2020).

Plusieurs propositions de solutions tendent à répondre à ces éléments de frictions. Il s'agit par exemple de faciliter la sécurité des données confidentielles et la conservation de l'historique

des modifications, deux éléments vus comme des défis majeurs. Néanmoins, de ces études, l'amélioration des *notebooks* dans la perspective d'une science ouverte (disponibilité des données, conditions de réutilisation, usage de licences) est très peu traitée (cf. [Discussion : Les notebooks comme outils stratégiques de la science ouverte ?](#)).

### III. DISCUSSION : LES *NOTEBOOKS* COMME OUTILS AU SERVICE DE LA SCIENCE OUVERTE

Suite à la synthèse, nous nous intéressons plus particulièrement à l'articulation actuelle entre *notebooks* et principes de science ouverte. La science ouverte vise à favoriser une diffusion des connaissances dès qu'elles sont disponibles, en utilisant les technologies numériques et collaboratives<sup>12</sup>. Les articles, les données et les scripts (codes source) sont les principaux éléments ciblés pour le partage et l'ouverture. Le libre accès (*open access*) est depuis longtemps au cœur des préoccupations de la science ouverte. Les données jouent également un rôle essentiel et sont soumises à un ensemble de normes et de principes regroupés sous l'acronyme FAIR. Les codes sources et logiciels ont quant à eux fait l'objet d'une attention plus tardive des politiques publiques en matière de science ouverte alors que le mouvement de l'open source et de logiciels libres est largement ancré dans des principes organisationnels, économiques et juridiques.

Nous présentons trois points centraux pour avancer sur les enjeux des *notebooks* computationnels dans une perspective de science ouverte : une meilleure connaissance des pratiques des professionnels de la recherche, une prise en charge explicite des enjeux de FAIRisation des *notebooks*, la conceptualisation des *notebooks* dans un environnement plus large de circulation libre des codes et des productions scientifiques notamment par le biais de licences libres.

#### 3.1 Que font les professionnels de la recherche avec des *notebooks* ?

Les travaux existants se concentrent sur le périmètre large de la science des données (*data science*), sans intérêt spécifique pour le monde académique (recherche au sein d'établissement public, d'institut de recherche, d'université, etc.). La connaissance spécifique des pratiques dans ces milieux est donc largement absente. Or, elle est nécessaire pour une meilleure compréhension non seulement des usages réels de ces dispositifs, mais aussi des spécificités qui peuvent exister au sein même des mondes scientifiques, en considérant les différences entre disciplines, les statuts et les démarches de recherche. Certains travaux sur les usages numériques des professionnels de la recherche montrent ainsi des pratiques différenciées<sup>13</sup>. S'il existe des témoignages ou des retours d'expérience (Beg et al., 2021), il n'y a que très peu de travaux dédiés aux pratiques des professionnels de la recherche autour des *notebooks* : le seul article systématique portant sur le domaine biomédical (Samuel & Mietchen, 2022).

En particulier, les *notebooks* sont souvent présentés comme une nouvelle interface entre code source et publication par le biais des articles exécutables. Ces derniers visent à revenir au traitement des résultats et de garantir transparence et reproductibilité. Un article de The

12 An approach to the scientific process that focuses on spreading knowledge as soon as it is available using digital and collaborative technology. [https://research-and-innovation.ec.europa.eu/strategy/strategy-2020-2024/our-digital-future/open-science\\_en](https://research-and-innovation.ec.europa.eu/strategy/strategy-2020-2024/our-digital-future/open-science_en)

13 Par exemple, l'enquête *State of open science practices* in France (Béchéec et al., 2022) ou encore Décliner la Science Ouverte (Gruson-Daniel & Groupe Projet Réussir L'Appropriation De La Science Ouverte, 2022)

Atlantic de 2018 était ainsi intitulé « *The scientific paper is obsolete* »<sup>14</sup>. Cependant, les travaux sur les *notebooks* computationnels, particulièrement du projet Jupyter, montrent que ces supports sont largement utilisés pour des aspects exploratoires, et seule une petite partie est amenée à devenir des supports finalisés à part entière (cf. [Mieux situer les notebooks dans leur écosystème](#)). De plus, différents usages existent du résultat d'exercice à des étapes intermédiaires du prototypage d'application. Il serait ainsi intéressant d'identifier les différentes situations dans lesquelles ces *notebooks* sont utilisés. Si l'enseignement semble occuper une place importante, leur rôle dans les dynamiques collaboratives serait à creuser, notamment comme interface entre un monde scientifique réalisant les traitements et les « non-spécialistes », et comme intermédiaire dans les pratiques de publication scientifique.

Plus encore, la littérature existante ne pose pas vraiment la question ni des conditions d'adoption – ce qui amène un utilisateur ou une utilisatrice à commencer à utiliser un *notebook* – ni de l'interdépendance entre les *notebooks* et d'autres formats (scripts de programmation, logiciels, forge, etc.). Couplé au fait qu'il n'y a pas d'éléments sur les profils des principaux utilisateurs et utilisatrices, cela appelle à des travaux plus spécifiques liés à la diffusion de ce support dans différentes communautés pour identifier les leviers et les freins. Dans la mesure où de nombreuses limites et critiques sont émises contre ce format (cf. [L'identification de nombreuses limites et critiques](#)) qui soulignent des demandes différenciées suivant les habitudes des utilisateurs et utilisatrices, nous pouvons nous attendre à une adoption différenciée. Les communautés scientifiques déjà largement familiarisées à des outils de programmation, notamment en sciences et techniques médicales, et les communautés plus éloignées de ces outils, comme les sciences sociales, ne vont sûrement pas utiliser de la même manière ces nouveaux outils.

### 3.2 Aller vers la FAIRisation des *notebooks*

Bien que les critiques faites aux *notebooks* s'accompagnent généralement de recommandations en matière de formatage, d'écriture et d'outils pour améliorer les pratiques, les dimensions de diffusion et d'ouverture restent peu explorées. Dans les articles étudiés, les conseils de diffusion restent génériques (cf. [Tableau 1](#)). Les principes FAIR ou plus spécifiquement l'usage d'identifiants pérennes comme le DOI sont absents des propos. Il n'existe pas encore de guides pratiques pour faciliter l'application des principes de la science ouverte et de la gestion de données FAIR dans l'univers des *notebooks*<sup>15</sup>.

Comme la plupart des recherches sur les *notebooks* et leurs améliorations se concentrent sur les applications d'ingénierie des données, la relation entre les *notebooks* et la science ouverte, en termes de bonnes pratiques, n'est peu, voire pas, évoquée. Ceci alors même que la genèse de projets comme Jupyter s'inscrivent directement dans la perspective de la science ouverte (Schultz, 2023). Toutefois, il est possible d'identifier plusieurs rencontres entre science ouverte et *notebooks* :

- Le partage du code, en relation avec les articles scientifiques (Wofford et al., 2020). Les *notebooks* sont des outils clés pour mettre en œuvre les principes de la programmation littéraire, transformant les articles scientifiques en entités exécutables, pouvant être lus et exécutés simultanément;

---

14 <https://www.theatlantic.com/science/archive/2018/04/the-scientific-paper-is-obsolete/556676/>

15 Un groupe de travail FAIR4RS s'est intéressé au processus FAIR pour les logiciels, mais cela ne touche pas spécifiquement les *notebooks*, dont les usages et les finalités diffèrent (Katz et al., 2021).

- Les *notebooks* sont également perçus comme une interface pédagogique majeure pour l'apprentissage de la programmation dans divers domaines des sciences (Hanč et al., 2020);
- Ils offrent une grande flexibilité par rapport à d'autres outils, sans imposer de chaînes d'opérations complexes.
- Les *notebooks* facilitent aussi le partage et l'organisation des analyses computationnelles en respectant quelques règles simples de rédaction et d'écriture (Rule et al., 2019);
- La dimension open source et gratuite des *notebooks* facilite leur adoption, notamment dans un contexte pédagogique, et s'intègre dans une communauté en constante évolution, élargissant ainsi les fonctionnalités possibles.

La question de la reproductibilité des *notebooks* abordée dans la littérature croise directement les enjeux de la science ouverte. Plusieurs pistes restent à explorer. Elles concernent aussi bien la complétude des références aux sources et aux données, que les problèmes de standardisation des écosystèmes techniques, ou encore les conditions de citation des *notebooks*. Plusieurs chemins sont envisageables pour amener une meilleure prise en compte des critères FAIR dans les *notebooks*. Il peut s'agir d'inclure la nécessité d'être identifiés (*findable*), par exemple en réfléchissant aux enjeux d'accessibilité et d'indexation, notamment d'identification des versions des *notebooks* lors de l'archivage, ou de la réutilisation avec la mention de licences libres.

### 3.3 Mieux situer les *notebooks* dans leur écosystème

Au-delà des enjeux spécifiques à la recherche scientifique – que ce soit les pratiques des professionnels de la recherche ou la FAIRisation des supports – la revue de littérature montre que les travaux existant se concentrent presque exclusivement sur les *notebooks* eux-mêmes (usages ou contenus). En revanche, leur intégration plus large dans les écosystèmes numériques n'est que peu explorée. Deux points semblent importants à souligner.

Tout d'abord, le cadre juridique entourant les *notebooks* et leur environnement n'est que peu présent et abordé. En 2019, Schröder et al. montrent par exemple une absence majeure de licence sur des Jupyter *notebooks* avec au moins un tiers des ressources sans licences (Schröder et al., 2019). Lorsque des licences sont indiquées, les plus fréquentes sont la MIT, la GPLv et la CC0. Les choix d'une licence ou d'une autre n'ont pas été justifiés. Le choix de la licence *Creative Commons Zero* (CC0) est peu commun dans le domaine de l'open source où les principales distinctions se font entre les licences permissives (licence MIT) et copyleft (GPLv). Pour d'autres, des licences propriétaires sont apposées. Par ailleurs, dans les *notebooks* accompagnant la publication d'articles, il n'existe pas ou peu de recommandations pour citer le code source ou les données dans le cadre de *notebooks* (Edelmann et al., 2020).

Dans de nombreux cas explorés, les *notebooks* sont stockés sur des dépôts (notamment Github), faisant partie de projets de recherche plus généraux. La licence peut alors se poser dans un fichier annexe texte et non pas dans le *notebook* en tant que tel. Cette réflexion sur la licence pose donc l'intégration du *notebook* dans le cadre de répertoires plus larges, impliquant des scripts, des données et des documents tiers dans divers formats. À ceci s'accompagne l'ensemble des dépendances et des logiciels nécessaires pour l'exécution. Il serait nécessaire de décentrer la focale pour mieux comprendre l'écosystème associé aux *notebooks*, tant en termes d'infrastructures que de périmètre d'utilisation (avec quoi sont-ils utilisés en pratique). De nombreuses dépendances sont identifiées dans les articles

(notamment des services, comme Github, ou Binder). Pour autant, sont-elles toujours utilisées ? Peut-on identifier des ensembles différents d'écosystèmes ? Ces dépendances sont indispensables pour éviter de considérer les *notebooks* computationnels comme des entités fixes et autonomes.

#### IV. CONCLUSION : MENER L'ENQUÊTE

La littérature, qui se concentre sur les *notebooks* computationnels du projet Jupyter, rend visible un ensemble de nouvelles pratiques et favorise la préconisation de bonnes pratiques. Alors que la question de l'open source et de l'ouverture des savoirs scientifiques est centrale dans la genèse des *notebooks* (IPython devenu Jupyter), elle n'est que peu abordée dans les enquêtes existantes qui se concentrent plus largement sur la science des données. Si les bonnes pratiques identifiées peuvent s'appliquer aussi aux praticiens et praticiennes de la recherche, il est probable que les enjeux d'exploration d'une part, et de publication scientifique d'autre part, amènent à problématiques spécifiques.

Dans cet article, nous avons décrit, sur la base d'une sélection d'articles les plus représentatifs de l'étude empirique des *notebooks*, les limites et critiques adressées aux *notebooks* tout autant que les bonnes pratiques proposées et l'outillage associé. Nous avons ensuite décidé d'ouvrir une discussion sur un point aveugle de cette revue de littérature que sont les enjeux des *notebooks* comme levier d'appropriation et d'intégration d'une démarche science ouverte, via plus particulièrement la mention des principes FAIR. Même si la reproductibilité est un élément phare mis en avant par l'utilisation des *notebooks* dans les discours sur la science ouverte, elle se relève difficilement atteinte dans la pratique. Notre but a ainsi été d'esquisser quelques pistes de FAIRisation des *notebooks* qui nécessite d'une part, de prendre en considération les différents publics d'utilisateurs et d'utilisatrices de ces outils dans la recherche académique (communautés disciplinaires, épistémiques, de pratique, etc.). D'autre part, il s'agit d'intégrer dans la réflexion l'usage des *notebooks* dans un écosystème d'infrastructures et d'outils de recherche participant, eux aussi, à la mise en œuvre d'une circulation ouverte des connaissances et de leur valorisation.

Dans le cadre de la suite du projet NOOS, notre but est par le biais d'un *focus group* et d'entretiens, de mieux cartographier les pratiques et usages des *notebooks* et leur articulation avec les enjeux de science ouverte que sont l'accès ouvert aux publications scientifiques, l'ouverture des données et des logiciels, l'appropriation des savoirs par des communautés variées. Dans le but de proposer des modèles de *notebooks* FAIR, une autre partie de l'étude vise à analyser un ensemble de *notebooks* et leur environnement, qui fera l'objet d'une cartographie, pour définir en fonction de leurs types et objectifs, les principes de science ouverte les plus importants à intégrer.

#### V. RÉFÉRENCES ET CITATIONS

##### 5.1 References

- Alena Guzharina & Alena Guzharina. (2020, décembre 17). *We Downloaded 10,000,000 Jupyter Notebooks From Github – This Is What We Learned*. The JetBrains Blog. <https://blog.jetbrains.com/datalore/2020/12/17/we-downloaded-10-000-000-jupyter-notebooks-from-github-this-is-what-we-learned/>
- Baker, M. (2016). 1,500 scientists lift the lid on reproducibility. *Nature*, 533(7604), Article 7604. <https://doi.org/10.1038/533452a>

- Beg, M., Taka, J., Kluyver, T., Kononov, A., Ragan-Kelley, M., Thiery, N. M., & Fangohr, H. (2021). Using Jupyter for Reproducible Scientific Workflows. *Computing in Science & Engineering*, 23(2), 36-46. <https://doi.org/10.1109/MCSE.2021.3052101>
- Chattopadhyay, S., Prasad, I., Henley, A. Z., Sarma, A., & Barik, T. (2020, avril 1). *What's Wrong with Computational Notebooks? Pain Points, Needs, and Design Opportunities*. ACM CHI Conference on Human Factors in Computing Systems. <https://www.microsoft.com/en-us/research/publication/whats-wrong-with-computational-notebooks/>
- Edelmann, A., Wolff, T., Montagne, D., & Bail, C. A. (2020). Computational Social Science and Sociology. *Annual review of sociology*, 46(1), 61-81. <https://doi.org/10.1146/annurev-soc-121919-054621>
- Grotov, K., Titov, S., Sotnikov, V., Golubev, Y., & Bryksin, T. (2022). *A Large-Scale Comparison of Python Code in Jupyter Notebooks and Scripts* (arXiv:2203.16718). arXiv. <https://doi.org/10.48550/arXiv.2203.16718>
- Hanč, J., Štrauch, P., Paňková, E., & Hančová, M. (2020). *Teachers' perception of Jupyter and R Shiny as digital tools for open education and science* (arXiv:2007.11262). arXiv. <https://doi.org/10.48550/arXiv.2007.11262>
- Katz, D. S., Gruenpeter, M., & Honeyman, T. (2021). Taking a fresh look at FAIR for research software. *Patterns*, 2(3). <https://doi.org/10.1016/j.patter.2021.100222>
- Kery, M. B., Radensky, M., Arya, M., John, B. E., & Myers, B. A. (2018). The Story in the Notebook : Exploratory Data Science using a Literate Programming Tool. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 1-11. <https://doi.org/10.1145/3173574.3173748>
- Leonelli, S. (2018). Rethinking Reproducibility as a Criterion for Research Quality. In *Including a Symposium on Mary Morgan : Curiosity, Imagination, and Surprise* (Vol. 36B, p. 129-146). Emerald Publishing Limited. <https://doi.org/10.1108/S0743-41542018000036B009>
- Liu, E. S., Lukes, D. A., & Griswold, W. G. (2023). Refactoring in Computational Notebooks. *ACM Transactions on Software Engineering and Methodology*, 32(3), 77:1-77:24. <https://doi.org/10.1145/3576036>
- Pimentel, J. F., Murta, L., Braganholo, V., & Freire, J. (2021). Understanding and improving the quality and reproducibility of Jupyter notebooks. *Empirical Software Engineering*, 26(4), 65. <https://doi.org/10.1007/s10664-021-09961-9>
- Quaranta, L., Calefato, F., & Lanubile, F. (2022). Eliciting Best Practices for Collaboration with Computational Notebooks. *Proceedings of the ACM on Human-Computer Interaction*, 6(CSCW1), 1-41. <https://doi.org/10.1145/3512934>
- Raghuandan, D., Roy, A., Shi, S., Elmqvist, N., & Battle, L. (2023). Code Code Evolution : Understanding How People Change Data Science Notebooks Over Time. *Conference on Human Factors in Computing Systems - Proceedings*. <https://doi.org/10.1145/3544548.3580997>
- Ramasamy, D., Sarasua, C., Bacchelli, A., & Bernstein, A. (2023). Visualising data science workflows to support third-party notebook comprehension : An empirical study. *Empirical Software Engineering*, 28(3), 58. <https://doi.org/10.1007/s10664-023-10289-9>
- Rule, A., Birmingham, A., Zuniga, C., Altintas, I., Huang, S.-C., Knight, R., Moshiri, N., Nguyen, M. H., Rosenthal, S. B., Pérez, F., & Rose, P. W. (2019). Ten simple rules for writing and sharing computational analyses in Jupyter Notebooks. *PLOS Computational Biology*, 15(7), e1007007. <https://doi.org/10.1371/journal.pcbi.1007007>
- Rule, A., Tabard, A., & Hollan, J. D. (2018). Exploration and Explanation in Computational Notebooks. *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, 1-12. <https://doi.org/10.1145/3173574.3173606>
- Samuel, S., & Mietchen, D. (2022). *Computational reproducibility of Jupyter notebooks from biomedical publications*. <https://doi.org/10.48550/ARXIV.2209.04308>
- Schröder, M., Krüger, F., & Spors, S. (2019). *Reproducible Research is more than Publishing Research Artefacts : A Systematic Analysis of Jupyter Notebooks from Research Articles* (arXiv:1905.00092). arXiv. <https://doi.org/10.48550/arXiv.1905.00092>
- Schultz, E. (2023). *Du laboratoire à Jupyter : La trajectoire d'un instrument logiciel libre de la science ouverte*. <https://hal.science/hal-04316428>
- Trisovic, A., Lau, M. K., Pasquier, T., & Crosas, M. (2022). A large-scale study on research code quality and execution. *Scientific Data*, 9(1), Article 1. <https://doi.org/10.1038/s41597-022-01143-6>
- Wagemann, J., Fierli, F., Mantovani, S., Siemen, S., Seeger, B., & Bendix, J. (2022). Five Guiding Principles to Make Jupyter Notebooks Fit for Earth Observation Data Education. *Remote Sensing*, 14(14). <https://doi.org/10.3390/rs14143359>
- Wang, A. Y., Wang, D., Drozdal, J., Liu, X., Park, S., Oney, S., & Brooks, C. (2021). What Makes a Well-Documented Notebook? A Case Study of Data Scientists' Documentation Practices in Kaggle. *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, 1-7. <https://doi.org/10.1145/3411763.3451617>
- Wang, J., Kuo, T., Li, L., & Zeller, A. (2020). Restoring reproducibility of Jupyter notebooks. *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings*, 288-289. <https://doi.org/10.1145/3377812.3390803>
- Wofford, M. F., Boscoe, B. M., Borgman, C. L., Pasquetto, I. V., & Golshan, M. S. (2020). Jupyter Notebooks as Discovery Mechanisms for Open Science : Citation Practices in the Astronomy Community. *Computing in Science & Engineering*, 22(1), 5-15. <https://doi.org/10.1109/MCSE.2019.2932067>
- Zhang, A. X., Muller, M., & Wang, D. (2020). *How do Data Science Workers Collaborate? Roles, Workflows, and Tools* (arXiv:2001.06684). arXiv. <http://arxiv.org/abs/2001.06684>

## 5.2 Annexes

### ANNEXE 1 Tableau des 12 références identifiées



Référence	Titre	Spécialité	Type de notebooks	Objectif	Méthodologie	Résultat principal	Science?	Recommandations?	Pratiques?
Raghunandan et al., 2023	<i>Code Code Evolution: Understanding How People Change Data Science Notebooks Over Time</i>	Computational engineering	Jupyter	Étudier l'évolution des notebooks dans la construction du sens	2574 notebooks de Github	Quantifier le type de notebook (exploratoire/explicatif)	Non	Outiller l'identification de la position du notebook sur l'axe E/E	Indirectement, une échelle de stabilisation
Liu et al., 2023	<i>Refactoring in Computational Notebooks</i>	Computational engineering	Jupyter	Étudier l'évolution du code dans les notebooks publics	200 notebooks avec commit sur Github	Des pratiques différentes suivant les genres des notebooks & des auteurs	Partiellement	Sur l'amélioration des outils de refactorisation	Indirectement, les évolutions des notebooks
Ramasamy et al., 2023	<i>Visualising data science workflows to support third-party notebook comprehension: an empirical study</i>	Computational engineering	Jupyter	Développer une stratégie de visualisation des notebooks	470 notebooks et expérimentation contrôlée d'utilisateurs avec 35 data scientist	Clarification d'une terminologie sur les data science workflow ; possibilité d'améliorer l'usage des notebooks par la visualisation du workflow	Partiellement	outil de visualisation ; améliorer l'explicitation du workflow	oui, réusage de notebook
Grotov et al., 2022	<i>A Large-Scale Comparison of Python Code in Jupyter Notebooks and Scripts</i>	Computational engineering	Jupyter	Étudier le style du code écrit dans les notebooks Jupyter	847881 notebooks avec licence ouverte	Le code des notebooks a un style différent (plus faible complexité ; plus d'interdépendance ; plus d'erreurs de style)	Non	Un package (linter) pour mesurer le style	Non
Samuel et al., 2024	<i>Computational reproducibility of Jupyter notebooks from biomedical publications</i>	Computational engineering	Jupyter	Tester la reproductibilité de notebooks associés à des publications biomédicales et l'écart entre les préconisations et les pratiques	9625 notebooks issus de publications de Pubmed	Discute la notion de reproductibilité. Montre la faible reproductibilité	Oui	Meilleure documentation	Non
Wagemann et al., 2022	<i>Five Guiding Principles to Make Jupyter Notebooks Fit for Earth Observation Data Education</i>	Eath Science	Jupyter	Présenter la démarche pour créer des notebooks pour un cours et les	Création de 70 notebooks	Identification de principes : importance des cellules de texte ; éléments de navigation ; suivre les principes de la programmation scientifique ; utiliser l'écosystème Jupyter pour partager ;	Oui	Améliorer les caractéristiques finales des notebooks	Restitution d'une expérience



				grandes principes	viser vers la reproductibilité				
Quaranta et al., 2022	<i>Eliciting Best Practices for Collaboration with Computational Notebooks</i>	Computational engineering	Jupyter	Identifier les meilleures pratiques collaboratives avec les <i>notebooks</i> chez les <i>data scientists</i>	Revue systématique de littérature ; 22 entretiens avec des <i>data scientists</i> ; 1380 <i>notebooks</i> de Kaggle	Liste de 17 bonnes pratiques issues de la littérature (tableau); Les individus sont au courant des bonnes pratiques mais ne les mettent pas forcément en application	Partiellement	Améliorer <i>built-in</i> l'environnement du <i>notebook</i>	Oui, de nombreux verbatims d'entretiens
Wang et al., 2021	<i>What Makes a Well-Documented Notebook? A Case Study of Data Scientists' Documentation Practices in Kaggle</i>	Computational engineering	Jupyter	Comprendre les meilleures pratiques de documentation utilisées par les <i>data scientists</i>	80 <i>notebooks</i> de Kaggle avec la meilleure notation	Liste de 9 types d'usage des cellules markdown (tableau)	Non	Non	Non
Pimentel et al., 2021	<i>Understanding and improving the quality and reproducibility of Jupyter engineering notebooks</i>	Computational engineering	Jupyter	Analyser les caractéristiques des <i>notebooks</i> qui limitent la reproductibilité à travers une série de questions (nommage, ordre d'un outil avec d'exécution, etc.)	1024269 <i>notebooks</i> de Github, sous échantillon de 38063 <i>notebooks</i> populaires ; expérimentation 12 participants	Détail de la structure des <i>notebooks</i> sur Github et de leur faible reproductibilité ; Liste de 8 meilleures pratiques ; <i>notebooks</i> populaires davantage reproductibles ; beaucoup de <i>notebooks</i> éducationnels ; proposition d'un <i>linting tool</i>	Non	bonnes pratiques et outil dédié	Non
Chattopadhyay et al., 2020	<i>What's Wrong with Computational Notebooks? Pain Points, Needs, and Design Opportunities</i>	Computational Science	Jupyter, Colab, Databricks, RStudio	Identifier les frictions introduits par l'usage des <i>notebooks</i> pour les <i>data scientists</i>	20 entretiens et questionnaire (N=156)	Tableau des problèmes identifiés (9)	Non	Non	Oui, via les entretiens
Rule et al., 2018	<i>Exploration and explanation in Computational notebooks interaction</i>	Human-Computer interaction	Jupyter	Décrire quels sont les usages que font les <i>data scientists</i> des <i>notebooks</i>	1230000 <i>Notebooks</i> sur Github ; échantillon de 200 <i>notebooks</i> liés à une publication scientifique ; 15 entretiens avec des <i>data</i>	Structure des dépôts Github contenant des <i>notebooks</i> ; usage du texte variables dans les <i>notebooks</i> académiques	Oui en partie	Non	Oui avec les entretiens

Kery et al., 2018	<i>The story in the notebook: Exploratory data science using a literate programming tool</i>	<i>Human- Computer interaction</i>	Jupyter	Comprendre ce que les <i>data</i> <i>scientists</i> conservent de leur explorations	<i>scientists</i> académiques	Entretiens avec 21 <i>data</i> <i>scientists</i>	Les <i>notebooks</i> sont souvent scratch pad et utilisé pour partager les résultats. Plusieurs stratégies d'organisation du <i>notebook</i> existent, dont un « <i>expand and reduce</i> », mais généralement avec une narration non linéaire.	Non	Proposition d'une loupe historique pour revenir sur les versions	Oui, surtout sur les explorations
-------------------	--	--	---------	---	----------------------------------	--	---	-----	--	--------------------------------------