

Programmation lettrée et R Markdown



Programmation lettrée et R Markdown

Vincent Goulet

Professeur titulaire

École d'actuariat, Université Laval

Version 2024.02

© ⓘ ⓘ ⓘ 2018-2024 par Vincent Goulet. « Programmation lettrée et R Markdown » est mis à disposition sous licence **Attribution-Partage dans les mêmes conditions 4.0 International** de Creative Commons. En vertu de cette licence, vous êtes autorisé à :

- **partager** — copier, distribuer et communiquer le matériel par tous moyens et sous tous formats ;
- **adapter** — remixer, transformer et créer à partir du matériel pour toute utilisation, y compris commerciale.

L'Offrant ne peut retirer les autorisations concédées par la licence tant que vous appliquez les termes de cette licence.

Selon les conditions suivantes :



Attribution — Vous devez créditer l'œuvre, intégrer un lien vers la licence et indiquer si des modifications ont été effectuées à l'œuvre. Vous devez indiquer ces informations par tous les moyens raisonnables, sans toutefois suggérer que l'Offrant vous soutient ou soutient la façon dont vous avez utilisé son œuvre.



Partage dans les mêmes conditions — Dans le cas où vous effectuez un remix, que vous transformez, ou créez à partir du matériel composant l'œuvre originale, vous devez diffuser l'œuvre modifiée dans les mêmes conditions, c'est-à-dire avec la même licence avec laquelle l'œuvre originale a été diffusée.

Code source

 [Voir sur GitLab](#)

Couverture

Lémur catta (*Lemur catta*) dans le parc national de l'Isalo, Madagascar. Crédit photo : © Bernard Gagnon **CC BY-SA 3.0 non transposé**, via [Wikimedia Commons](#).

« Finalement, comme plusieurs d'entre-vous n'ont pas beaucoup de temps pour [...] reprendre tout ce que vous venez de produire en R pour en faire un rapport avec de multiple copier/coller/effacer/modifier/jeter/recommencer, il serait peut-être intéressant pour vous de commencer à utiliser R Markdown. »

— **Samuel Cabral Cruz**, analyste en actuariat chez DGAG, promotion 2015

Concepts généraux

Markdown

R Markdown

Documents simples

Mathématiques

Dernières observations

Documentation utile

Concepts généraux

Système de **programmation lettrée** qui permet de combiner le code informatique et le texte d'une analyse dans un même fichier.

- Encourage la **recherche reproductible**
- Prise en main rapide et facile grâce à la syntaxe simplifiée de **Markdown** pour le texte

« Un travail de recherche est dit reproductible si toutes les informations qui concernent ce travail, incluant, sans s'y limiter, le texte, les données, et le code de programmation, sont rendues disponibles de telle sorte que n'importe quel chercheur indépendant peut reproduire les résultats. »

(Vandewalle, Kovacevic et Vetterli, 2009 ; traduction de Desquilbet et collab, 2019)

En termes simples

- Tout conserver dans un **format** facile à sauvegarder, à modifier et à partager
- Générer les résultats d'un rapport à partir du **code informatique**
- Effectuer un **suivi des versions** automatisé de tout ce matériel

Programmation lettrée

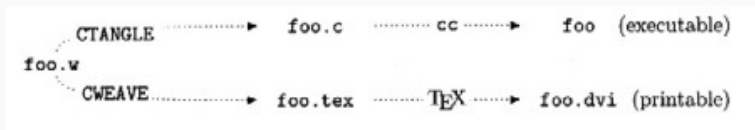
*I believe that the time is ripe
for significantly better
documentation of programs,
and that we can best
achieve this by considering
programs to be works of
literature. Hence, my title:
Literate Programming.*

— Donald Knuth, 1982



Photo : Vivian Cromwell. © Simons Foundation, via [QuantaMagazine.org](https://www.quantamagazine.org).

- Combiner un programme et sa documentation dans un même fichier dans un format facile à consulter pour un **humain**



- Plusieurs systèmes au fil du temps : **WEB** [↗](#) (Knuth, 1984), **CWEB** [↗](#) (Knuth et Levy, 1987), **noweb** [↗](#) (Ramsey, 1989), **Sweave** [↗](#) (Leisch, 2002), **knitr** [↗](#) (Xie, 2012)

Orienté vers la création de documents PDF.

- Texte en format \LaTeX
- Code informatique R

Avantages

- Qualité typographique, en particulier les équations mathématiques
- Simplicité de Sweave
- Universel et stable
- Documentation

Inconvénients

- Courbe d'apprentissage
- Peu d'intégration avec les outils de partage en ligne

R Markdown

Orienté vers la création de pages web et de documents PDF.

- Texte en format Markdown
- Code informatique R

Avantages

- Prise en main rapide et facile grâce à la syntaxe simplifiée
- Nombreux formats de sortie
- Intégration avec RStudio et GitHub


Inconvénients

- Nombreuses couches logicielles (Markdown, YAML, knitr, Pandoc, \LaTeX , HTML, ...)
- Documentation
- Prise en charge limitée des équations mathématiques
- \LaTeX requis pour équations complexes ou mises en forme particulières

Markdown



- Caractères sans mise en forme
- Universel (au type de codage près)
- Compatible avec les éditeurs de texte
- Pas de surprise, pas de maux de tête

Markdown  est un langage de balisage léger pour ajouter de la mise en forme **simple** dans un fichier de texte brut.

- Créé par John Gruber en 2004
- Dernière mise à jour en 2004 (!)
- Syntaxe simple à **lire** et à **écrire** pour titres, tableaux, liens, images, etc.
- Standard *de facto*

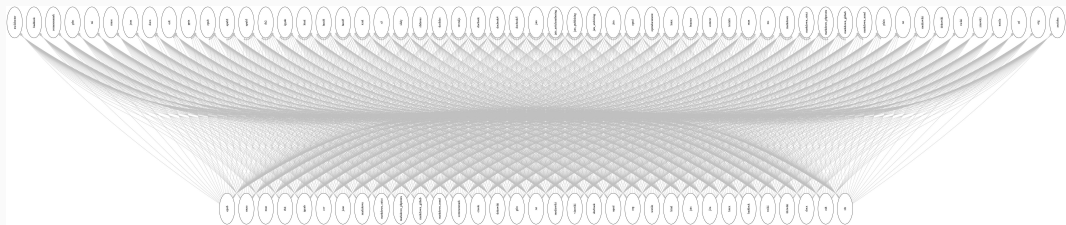
Syntaxe Markdown

Markdown permet d'indiquer simplement de l'**italique** ou du ****gras****.


Listes

- Épicerie
 1. Bananes
 2. Hamburger
 - * pain
 - * viande
- Quincaillerie

Convertisseur universel vers les formats PDF, HTML, DOC, etc.



Utiliser le fichier `exercice-markdown.md`.

1. Ouvrir le fichier dans RStudio.
2. Copier l'intégralité du texte du fichier et le coller dans [Markdown Live Preview](#) .
3. Comparer l'entrée et la sortie.

R Markdown

Processus de création d'un document en un coup d'oeil



Exemple de document

foo.Rmd

L'utilisateur de R interagit avec l'interprète en entrant des commandes à l'invite de commande:

```
```{r echo=TRUE}
```

```
2 + 3
```

```
```
```

La commande `exp(1)` donne `r exp(1)`, la valeur du nombre e .

Après traitement

```
knit("foo.Rmd") → foo.md
```

L'utilisateur de R interagit avec l'interprète en entrant des commandes à l'invite de commande:


```
` `` `r  
2 + 3  
` ``  
  
` ``  
## [1] 5  
` ``
```

La commande ``exp(1)`` donne 2.7182818, la valeur du nombre e .

Documents simples

Anatomie d'un fichier

Trois composantes principales d'un fichier R Markdown :

1. Entête de configuration **YAML**  (YAML Ain't Markup Language)
2. Texte en format Markdown
3. Code R sous forme de

blocs de code (*chunks*)

```
```${r}  
summary(cars)
```
```

expressions au fil du texte

```
contient `r nrow(cars)` lignes  
et `r ncol(cars)` variables
```


Ouvrir le fichier `exercice-introduction.Rmd` dans RStudio.

1. Identifier l'entête de configuration dans le fichier.
2. Identifier un bloc de code R.
3. Identifier des expressions R au fil du texte.

Formats de sortie

Quelques uns des formats pris en charge par R Markdown ([liste complète](#) .

- Document PDF (`pdf_document`; via \LaTeX)
- Document HTML (`html_document`)
- Diapositives avec ioslides (`ioslides_presentation`)
- Diapositives PDF avec Beamer (`beamer_presentation`; via \LaTeX)
- Word (`word_document`)



Utiliser le fichier `exercice-introduction.Rmd`.

1. Faire du répertoire dans lequel se trouve le fichier le répertoire de travail de R.
`Session | Set Working Directory | To Source File Location`
2. Charger le paquetage **rmarkdown** dans la session R.


```
| > library(rmarkdown)
```

3. Composer le document en format PDF.

```
| > render("exercice-introduction.Rmd", encoding = "UTF-8")
```

4. Répéter l'étape précédente après avoir changé `pdf_document` dans l'entête pour `html_document`.



Dans RStudio, le bouton  Knit ▼ lance directement la composition d'un document.



RStudio est livré avec sa propre version de Pandoc et c'est celle-ci qui est utilisée lors de toute composition lancée depuis l'éditeur. Pour composer dans R sans RStudio, vous devez disposer d'une version autonome de Pandoc.

Options des blocs de code R

Différentes options placées dans `{r }` permettent de contrôler le traitement d'un bloc de code.

echo affiche le code source si **TRUE** (par défaut)

eval exécute le code et affiche la sortie si **TRUE** (par défaut)

include affiche le contenu du bloc si **TRUE** (par défaut)



Soyez explicite dans les blocs de code, surtout avec **echo**.

Utilisation des options

- Afficher une expression sans l'évaluer ni afficher le résultat

```
```{r echo = TRUE, eval = FALSE}  
2 + 3
```
```

- Afficher un résultat sans montrer l'expression

```
```{r echo = FALSE, eval = TRUE}  
2 + 3
```
```

- Cacher complètement un calcul (mais effectuer ce calcul)

```
```{r include = FALSE}  
2 + 3
```
```

Configuration du format de sortie

Tout passe par l'entête YAML.

```
---
title: ""
output:
  html_document:
    toc: TRUE
    toc_float: TRUE
    number_sections: TRUE
    theme: flatly
    highlight: espresso
---
```



Dans la syntaxe YAML, l'indentation, la position des « : » et les espaces jouent un rôle.

Utiliser le fichier `exercice-configuration.Rmd`.

1. Composer le fichier tel quel et visualiser la présentation ioslides.
2. Ajouter les options suivantes dans l'entête et visualiser de nouveau la présentation :

```
transition: faster  
widescreen: yes
```

3. Modifier l'entête pour produire plutôt une présentation Beamer standard.

```
output:  
  beamer_presentation:
```

4. Modifier l'entête pour utiliser les options de Beamer suivantes :

```
latex_engine: xelatex  
theme: metropolis  
highlight: espresso
```


Mathématiques

- Décrire des mathématiques requiert un « langage » spécial
- Syntaxe de \LaTeX
- Toutes les fonctionnalités de \LaTeX ne sont pas disponibles, surtout en format HTML

Équations au fil du texte

Une équation « en ligne » directement dans le texte comme $(a + b)^2 = a^2 + 2ab + b^2$ est placée entre \$ \$.

Une équation « en ligne » directement dans le texte comme $(a + b)^2 = a^2 + 2ab + b^2$ est placée entre ` \$ \$ `.

Équations hors paragraphe

Les équations hors paragraphe séparées du texte principal comme

$$\int_0^{\infty} f(x) dx = \frac{1}{n!} \sum_{i=1}^n \alpha_i e^{x_i} f(x_i)$$

sont placées entre `$$` `$$`.

Les équations hors paragraphe séparées du texte principal comme

`$$`

`\int_0^{\infty} f(x) dx =`

`\frac{1}{n!}`

`\sum_{i = 1}^n \alpha_i e^{\{x_i\}} f(x_i)`

`$$`

sont placées entre ``$$` `$$``.

Consulter le chapitre 7 de [Rédaction avec \$\text{\LaTeX}\$](#) .

Le document est distribué avec TeX Live et donc déjà installé sur votre poste de travail.

Dans RStudio, ouvrez l'onglet Terminal et entrez à la ligne de commande du système d'exploitation

```
$ texdoc formation-latex-ul
```

Reproduire le texte ci-dessous dans les formats PDF et HTML.

La densité conjointe de X_1 et X_2 est simplement le produit des densités marginales :

$$f_{X_1 X_2}(x_1, x_2) = \frac{1}{\Gamma(\alpha)} x_2^{\alpha-1} e^{-(x_1+x_2)}, \quad x_1 > 0, x_2 > 0.$$



La commande `\quad` produit un grand espace : | |.

Reproduire avec R Markdown l'analyse sommaire des données de location BIXI Montréal présentée dans les fichiers `analyse-bixi.pdf` et `analyse-bixi.html`.

Utiliser l'échantillon des données de BIXI Montréal dans le fichier `bixi-sample.rds` livré avec la formation.


```
> bixi <- readRDS("bixi-sample.rds")
```

Dernières observations

R Markdown est simple à utiliser,
mais uniquement pour faire des choses simples

Si vous recherchez plus de contrôle




R Markdown est surtout orienté vers la création de pages web.

- Option pour les plus gros documents : **Bookdown** 
- Ajoute une autre (!) couche logicielle par-dessus R Markdown
- Si la mise en forme est importante, je recommande \LaTeX

R Markdown et Shiny sont des technologies très liées à Posit, la compagnie derrière RStudio.

- Créateur de knitr travaille pour Posit
- Documentation officielle dans le site de Posit
- RStudio incite à utiliser ces formats
- Communauté très active dans les réseaux sociaux, Stack Overflow, etc.

Documentation utile

- Site officiel 
- *Cheatsheet* de la syntaxe 
- Tutoriel interactif de 10 minutes 

- [Page d'accueil du projet R Markdown](#) ↗
- [Cheatsheet R Markdown](#) ↗
- [Guide de référence R Markdown](#) ↗
- [Options PDF de R Markdown](#) ↗
- [Guide knitr](#) ↗
- [Options knitr](#) ↗

Ce document a été produit par le système de mise en page \LaTeX avec la classe **beamer** et le thème Metropolis. Les titres et le texte sont composés en Fira Sans, les mathématiques en Fira Math et le code informatique en Fira Mono. Les icônes proviennent de la police Font Awesome.