

# R: Rédiger

Eric Marcon

02 novembre 2020

R Markdown

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Ecrire avec RMarkdown

# Principes

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Markdown est un format d'écriture très simple et lisible.

Pandoc convertit les documents Markdown en LaTeX (à compiler en PDF), HTML, Word. . .

RMarkdown étend Markdown pour R et Bookdown étend RMarkdown.

knitR exécute le code R à l'intérieur des documents et appelle RMarkdown.

# Document RMarkdown simple

Dans RStudio : *File / New File / R Markdown...*


Eric Marcon


Ecrire avec  
RMarkdown


Intégration  
continue


Figures pour  
la publication

New R Markdown

 Document

 Presentation

 Shiny

 From Template

**Title:**

**Author:**

**Default Output Format:**

☒ HTML

Recommended format for authoring (you can switch to PDF or Word output anytime).

☐ PDF

PDF output requires TeX (MiKTeX on Windows, MacTeX 2013+ on OS X, TeX Live 2013+ on Linux).

☐ Word

Previewing Word documents requires an installation of MS Word (or Libre/Open Office on Linux).

# Syntaxe

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Etudier le modèle :

- En-tête YAML et premier bout de code ;
- Formatage du texte ;
- Bouts de code.

Anti-sèche.

# Tricotage

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

- Tester les 3 formats : HTML, PDF, Word.

PDF nécessite LaTeX.

Le même document peut être utilisé sous différents format sans réécriture.

# Modèles

Des packages fournissent des modèles.

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

New R Markdown

- Document
- Presentation
- Shiny
- From Template

**Template:** [Using R Markdown Templates](#)

|                                     |             |
|-------------------------------------|-------------|
| R Journal Submission                | {rticles}   |
| Article EcoFoG                      | {EcoFoG}    |
| Presentation EcoFoG                 | {EcoFoG}    |
| GitHub Document (Markdown)          | {rmarkdown} |
| Package Vignette (HTML)             | {rmarkdown} |
| Association for Computing Machinery | {rticles}   |
| American Chemical Society           | {rticles}   |

This template contains multiple files. Create a new directory for these files:

**Name:**

Untitled

**Location:**

~/R/EcoFoG [Browse...](#)

# Modèles EcoFoG

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Présentation : Beamer et HTML (utiliser *ioslides*).

Article : PDF pour l'autoarchivage et HTML pour la lecture.  
Word possible.

Ouvrage (Mémoire de master, Thèse, HDR, livre) : PDF et  
HTML.

Démonstration

Documentation dans les modèles.

- Ouvrir les modèles.



# Méthode de travail

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Un projet R contient tout :

- Modèle de document ; Fichiers nécessaire à la mise en forme (styles de texte, de bibliographie, . . . ) ;
- Données ; Code R pour produire les résultats, y compris les figures ;
- Figures additionnelles.

Ce n'est pas un package :

- Un package a une organisation formelle inutile pour un article.

# Création du projet

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Utiliser l'assistant *Nouveau Fichier /R Markdown...* /A partir d'un modèle / Article EcoFoG.

Un nouveau dossier est créé. Le transformer en projet R:  
*Nouveau projet /A partir d'un dossier existant.*

Tricoter pour vérifier le fonctionnement.

Possibilité de tricoter en HTML pour gagner du temps.

# Données

R Markdown

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Placer les données dans le projet, dans un format lisible par R (typiquement, CSV).

Lire les données dans le préambule de l'article.

# Calculs et figures

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Placer les calculs dans des bouts de code dans la section Matériels et Méthodes.

Utiliser les options des bouts de code :

- `echo` : affichage du code dans l'article (`FALSE` pour la publication) ;
- `cache` : pour ne pas répéter les calculs à chaque compilation.

Les figures sont produites directement par le code :

- insérer les bouts de code contenant les commandes `plot` dans la section *Résultats*.

# Bibliographie

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Utiliser sa bibliographie générale, produite par Mendeley :

- Pas de perte de temps pendant la rédaction ;
- Produire une bibliographie autonome à la fin avec Jabref (*Tools /New Sublibrary based on AUX file*).

Ou utiliser directement une bibliographie spécifique, dans un fichier bib autonome.

Voir la Documentation.

# Séparer l'atelier et le magasin

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Modèles EcoFoG sauf *ouvrage* : script *GitHubPages.R*

Les fichiers produits (PDF, HTML, libs) sont déplacés dans  
*/docs*.

Le fichier */README.md* est dupliqué dans */docs*.

- Tricoter aux formats PDF et HTML.
- Exécuter *GitHubPages.R*.

Modèles EcoFoG *ouvrage* : *Build Book* tricote tout dans */docs*.

# Collaboration

Résumé

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Passer le projet sous Git et le pousser sur GitHub.

Ajouter des collaborateurs.

# Présentation

R Markdown

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

## Activer les pages web du dépôt GitHub :

- Settings, GitHub Pages :
  - Source = Master Branch / docs Folder
  - choisir un thème.



# Présentation

• R Markdown

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Dans README.md, ajouter les liens vers les fichiers produits :

- HTML pour la lecture en ligne ;
- PDF pour le téléchargement.

Révisé par

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

# Intégration continue

# Objectifs

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Sous-traiter à Travis-CI (service en ligne) la construction des livres.

Permet une mise à jour en continu, à chaque push.

# Moyens

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Ouvrir un compte sur Travis

Activer le projet de livre

Ajouter un script `.travis.yml`(cours, chap. 6)

Pages GitHub sur la branche `gh-pages`

Rédiger

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Figures pour la publication

# Principes

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Produire des figures à utiliser hors des documents Markdown.

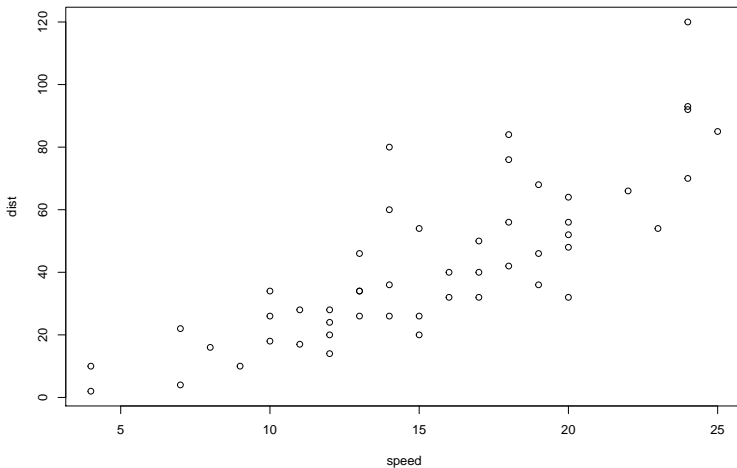
Continuité entre l'analyse de données et la production de figures.

Eviter les copier-coller : créer directement des fichiers pour contrôler les tailles relatives.

Si les données changent, les figures sont refaites par le script.

# Exemple

```
plot(cars)
```



Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Demander à R d'écrire dans un fichier plutôt qu'à l'écran

```
postscript("Fig1.eps", width = 6, height = 4, horizontal = FALSE)  
plot(cars)  
dev.off()
```

```
## pdf  
## 2
```

postscript crée un fichier EPS, pdf un fichier PDF et svg un fichier SVG.

bmp, jpeg, png, tiff créent des fichiers raster.



# Usage

Eric Marcon

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Fichiers vectoriel pour les figures : PostScript ou PDF pour LaTeX, MetaFile pour Word:

```
library("devEMF")  
emf("Fig1.emf", width = 6, height = 4)  
plot(cars)  
dev.off()
```

```
## pdf  
## 2
```

Fichiers raster pour les rasters ou demandes particulières.

**ragg** améliore le rendu des figures en PNG, JPEG et TIFF.

Fonctions : `agg_png()`, `agg_jpeg()` et `agg_tiff()`.

Utilisable par RStudio  $\geq 1.4$  et par **knitr**.

# Points de vigilance

✎ Rédiger

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

Taille des caractères.

Couleurs ou non.

Ne passer en raster qu'en tout dernier recours. 300 dpi minimum.

Utiliser la documentation des fonctions pour les finitions (polices, taille, transparence. . .).

# Exemple de PLOS

Écrit avec

Eric Marcon

Ecrire avec  
RMarkdown

Intégration  
continue

Figures pour  
la publication

## Instructions

Les seuls formats acceptés sont PostScript et TIFF.

Les tailles sont précisées.

R n'encapsule pas les polices dans les fichiers EPS :

- utiliser `embedFonts()` pour le faire (nécessite *Ghostscript*)
- ou utiliser *Inkscape* pour transformer les polices en courbes avant publication.