



EXPOSE

LOGICIEL R



REDIGE PAR :

Larry SANDJO
Rasmane BAMOGO
Ousseynou DIOP
Albert ZINABA

SUPERVISE PAR :

Aboubacar HEMA
Data Analysis



Albert ZINABA
ISE1 Eco



Rasmane BAMOGO
ISE1MATHS



Ousseynou DIOP
ISEP3



Larry SANDJO
ISE1 MATHS



R MARKDOWN



Table des matières

I	Introduction	7
I.1	C'est quoi au fait R Markdown ?	7
I.2	Pourquoi R Markdown est-il aussi important ?	7
II	Utilisation de R markdown	9
II.1	Installation du package de R markdown	9
II.2	Présentation de l'environnement R Markdown	9
II.2.1	Structure d'un document R Markdown	9
II.2.2	En-tête YAML	11
II.2.3	Table des matières	11
II.2.4	Les styles et thèmes	12
II.2.5	L'option params	12
II.2.6	Chunks de code	16
III	Personnalisation d'un document R markdown	19
III.1	Les notes de bas de page	19
III.2	Ajout de bibliographie	19
III.3	Insertion de tableau	19
III.4	Insertion de lien	20
III.5	Insertion d'images	21
III.6	Equations mathématiques	22
III.6.1	Equations en ligne	22
III.6.2	Équations centrées	22
III.6.3	Notation LaTeX	23
III.7	Packages et extensions	23
III.7.1	Création du modèle	24
III.7.2	Utilisation du modèle	24
III.7.3	Personnalisation supplémentaire	25
III.8	Compléments	26
III.8.1	R markdown Cheat Sheet	26
III.8.2	Le visual de R Markdown.	26
III.8.3	Rapport rapide à partir d'un script R	28
IV	Formats de sortie avec R Markdown	29
IV.1	Creation beamer	29
IV.1.1	Etapes 1 :	29
IV.1.2	Etapes 2 :	29
IV.1.3	Etapes 3 :	30
IV.2	Personnaliser le beamer	30
IV.2.1	Thèmes :	30
IV.2.2	Thème et couleur :	30
IV.2.3	Exemple :	30
IV.3	Niveau coulissant :	31
IV.3.1	Remarque	31
IV.3.2	Exemple :	31
IV.4	Quelques statistiques	31
IV.4.1	Résultat	31
IV.4.2	Mise en forme avec KableExtra	32

IV.4.3	Résultat	32
IV.5	Un peu de statistiques spatiales(1)	32
IV.5.1	Résultat	32
IV.6	Un peu de statistiques spatiales(2)	33
IV.6.1	Résultat	33
IV.7	Autres fonctionnalités	34
V	R Markdown et LaTeX	35
VI	R Markdown, HTML et CSS	38
VII	Creation de livre avec bookdown de R markdown	39

Liste des tableaux

2	Evenements G5 Sahel	32
---	-------------------------------	----

Table des figures

1	Une sortie de promotion à l'ENSAE	22
2	première étape création beamer	29
3	deuxieme étape création beamer	29
4	troisième étape création beamer	30
5	Balises LaTeX	37

I Introduction

La création d'un rapport, d'un mémoire ou même d'une thèse à caractère scientifique nécessite très souvent d'y faire apparaître des tableaux, des graphiques voire même des lignes de codes explicites. Les logiciels classiques tels que Ms Word et LaTeX requièrent des procédés ambigus pour insérer de tels éléments. R Markdown est un outil qui réponds à de telles préoccupations.

I.1 C'est quoi au fait R Markdown ?

Avant de définir R markdown, nous allons au préalable définir le langage Markdown. Le langage Markdown est un langage de balisage léger qui permet de formater du texte de manière simple et lisible, en utilisant des caractères typographiques courants. R Markdown est une variante de markdown qui est spécifique à R. Il permet d'écrire un document en utilisant markdown pour produire du texte et pour incorporer du code R, Python, SQL et bien d'autres et afficher leurs sorties. Les fichiers R Markdown ont une extension `.Rmd`.

I.2 Pourquoi R Markdown est-il aussi important ?

R Markdown est un outil très important. En effet, R Markdown offre une grande flexibilité et diversité en termes de formats de sortie pour les documents. Il permet les formats de sortie suivant :

- **HTML** : Les documents HTML peuvent être interactifs et inclure des éléments comme des graphiques interactifs (avec des bibliothèques comme Plotly), des tableaux de données dynamiques (avec DataTables), etc.
- **PDF**
- **Word**
- **Présentations** : R Markdown prend en charge la création de présentations au format HTML ou PDF.
- **Dashboards interactifs** : Avec des packages comme flexdashboard ou shiny, on peut créer des tableaux de bord interactifs à partir de R Markdown. Ces tableaux de bord peuvent inclure des graphiques interactifs, des widgets de contrôle, etc.
- **Livres électroniques (e-books)** : On peut générer des livres électroniques au format EPUB ou Kindle à partir de R Markdown. Cela peut être utile si vous écrivez un livre ou un document plus long et que vous souhaitez le distribuer sous forme de livre électronique.
- **Beamer (pour les présentations LaTeX)**

D'autre part, R Markdown est un outil puissant qui permet de combiner du code R, des résultats de code, et du texte formaté dans un document unique. Voici quelques-unes de ses principales fonctionnalités :

- **Intégration de code** : Vous pouvez intégrer du code R directement dans votre document Markdown en utilisant des blocs de code délimités par trois backticks (`````). Ce code peut être exécuté lors de la génération du document, et les résultats peuvent être affichés directement dans le document.
- **Rendu dynamique** : R Markdown permet de générer une grande variété de formats de sortie, y compris HTML, PDF, Word, présentations (via HTML ou PDF), et bien d'autres. Vous pouvez spécifier le format de sortie dans les métadonnées du document ou lors de la génération du document.
- **Mixage de formats de texte** : Vous pouvez utiliser du texte formaté en Markdown pour structurer votre document (titres, listes, etc.), ainsi que du texte en LaTeX pour des formattages plus avancés (équations mathématiques, références bibliographiques, etc.).
- **Incorporation de graphiques** : R Markdown permet de générer et d'inclure des graphiques générés par R (à l'aide de ggplot2, base R, etc.) directement dans le document

final. Les graphiques sont automatiquement insérés à l'endroit où vous spécifiez le code pour les créer.

- **Support de nombreux langages de programmation** : Bien que son nom suggère une utilisation principale avec R, R Markdown prend en charge d'autres langages de programmation comme Python, Julia, et bien d'autres. Vous pouvez intégrer et exécuter du code dans ces langages dans le même document.
- **Personnalisation avancée** : Vous pouvez personnaliser de nombreux aspects de votre document final, y compris le style, les polices, les marges, etc., en utilisant des options spécifiques au format de sortie ou en passant par des fichiers de configuration.
- **Facilité d'utilisation** : R Markdown est conçu pour être convivial, même pour les débutants. Sa syntaxe est relativement simple et intuitive, ce qui en fait un choix populaire pour la génération de rapports, de documents de recherche, de présentations, etc.

Tout autre détail sur le langage R Markdown peut être obtenu à partir du lien suivant : <http://rmarkdown.rstudio.com>.

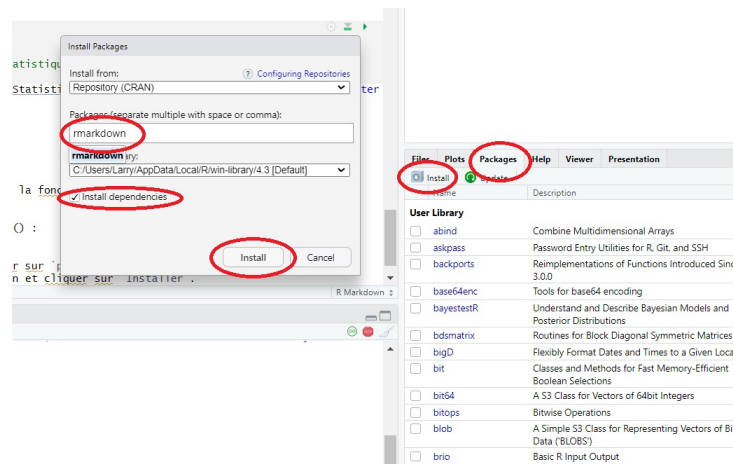
II Utilisation de R markdown

II.1 Installation du package de R markdown

Pour installer le package R Markdown, on peut utiliser directement la fonction `install.packages()` : `install.packages("rmarkdown")`

Pour charger le package R Markdown, on utilise la fonction `library()` : `library(rmarkdown)`

On peut aussi aller dans la fenêtre de présentation et puis cliquer sur packages ensuite install et enfin sélectionner l'installation à partir du CRAN, entrer le nom du package Rmarkdown et cliquer sur Installer.



NB : Toujours prendre la peine de cocher sur `install dependencies`.

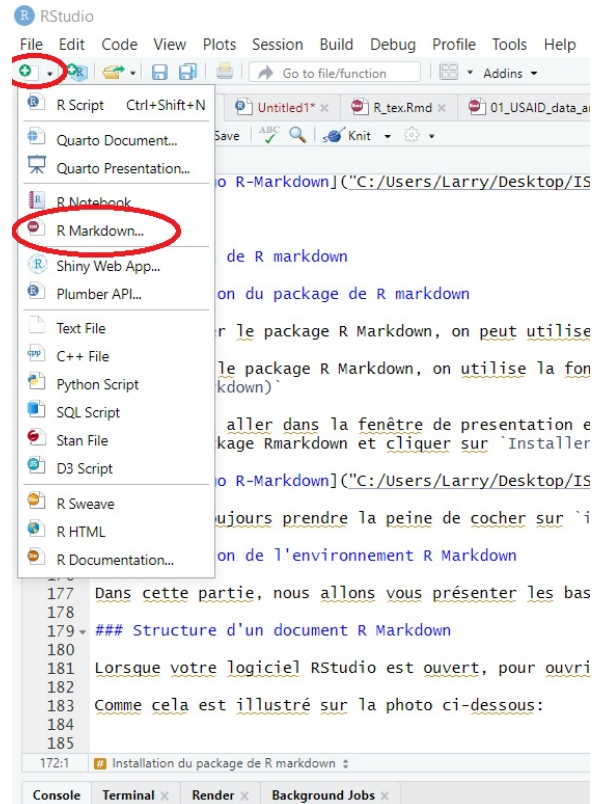
II.2 Présentation de l'environnement R Markdown

Dans cette partie, nous allons vous présenter les bases de R markdown.

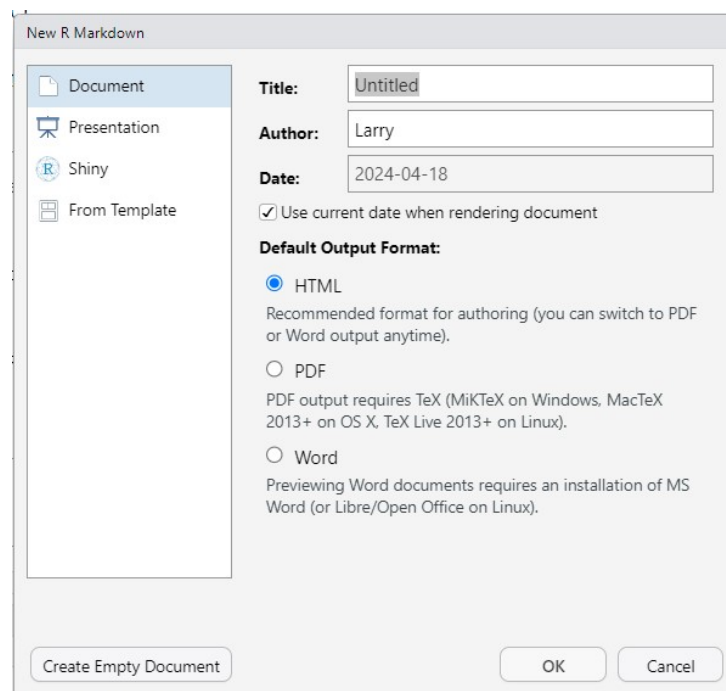
II.2.1 Structure d'un document R Markdown

Lorsque votre logiciel RStudio est ouvert, pour ouvrir un fichier R markdown, vous pouvez cliquer sur `new file` et ensuite choisir le fichier R markdown.

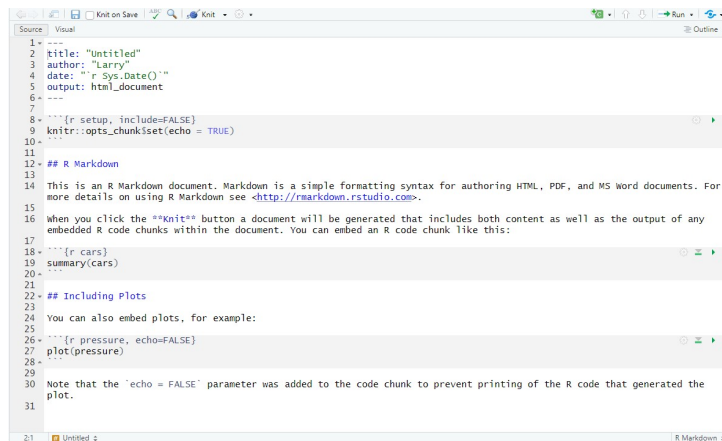
Comme cela est illustré sur la photo ci-dessous :



La fenêtre ci-dessous s'ouvre alors sur laquelle vous pouvez spécifier le titre de votre document, le nom de l'auteur, la date et le format de sortie (HTML, pdf ou Word) :



Par défaut, la page qui s'affiche est la suivante :



```

1 ---
2 title: "Untitled"
3 author: "Larry"
4 date: "r Sys.Date()"
5 output: html_document
6 ---
7
8 ```{r setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE)
10 ```
11
12 ## R Markdown
13
14 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For
15 more details on using R Markdown see <http://rmarkdown.rstudio.com>.
16
17 When you click the "Knit" button a document will be generated that includes both content as well as the output of any
18 embedded R code chunks within the document. You can embed an R code chunk like this:
19
20 ```{r cars}
21 summary(cars)
22 ```
23
24 ## Including Plots
25
26 You can also embed plots, for example:
27
28 ```{r pressure, echo=FALSE}
29 plot(pressure)
30 ```
31
32 Note that the 'echo = FALSE' parameter was added to the code chunk to prevent printing of the R code that generated the
33 plot.

```

Vous pouvez déjà y voir le titre # R markdown et apercevoir quelques chunks (pas d'inquiétude on verra ce que c'est dans la suite).

II.2.2 En-tête YAML

L'en-tête d'un document R Markdown (parfois appelé YAML header) est délimité par deux lignes de pointillés et contient les métadonnées du document (titre, auteurs, options générales de mise en page...). Il contient au minimum le titre du document et le format de sortie. Il peut être enrichi d'autres champs pour modifier certaines métadonnées (par exemple la date) ou le style du document compilé. Voici un exemple d'en-tête :

```

title: "Partie_BAMOGO"
author: "BAMOGO"
date: "2024-03-14"
output:
  word_document:
    toc: true
    number_sections: true
    reference_docx: garde.docx
  html_document:
    toc: true
---

```

De nombreux champs supplémentaires peuvent être ajoutés. Par exemple, les références bibliographiques sont listées dans un fichier séparé et c'est R Markdown lui-même qui se charge de la mettre en forme et de la lier à des références dans le texte.

II.2.3 Table des matières

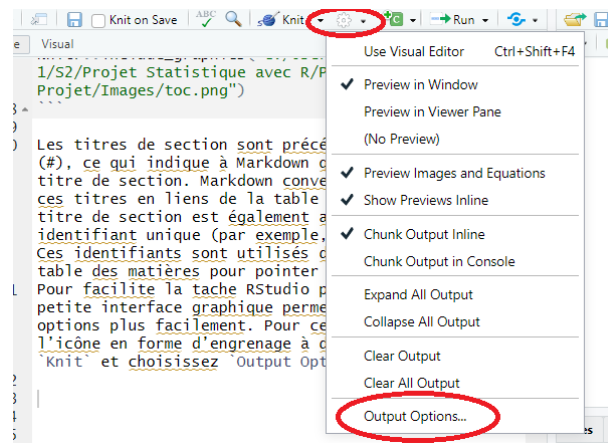
Pour inclure une table des matières dans un document R Markdown généré avec RStudio, vous pouvez utiliser l'option `toc` dans les métadonnées YAML. Voici comment vous pouvez le faire L'option `toc: true` dans le bloc YAML indique à R Markdown de générer une table des matières ou passer menu Edit ensuite Output Options et activer la partie table of contents.

```

title: "Partie_BAMOGO"
author: "BAMOGO"
date: "2024-03-14"
output:
  word_document:
    toc: true
    number_sections: true
    reference_docx: garde.docx
  html_document:
    toc: true

```

Les titres de section sont précédés par un numéro #, ce qui indique à Markdown qu'il s'agit d'un titre de section. Markdown convertira automatiquement ces titres en liens de la table des matières. Chaque titre de section est également associé à un identifiant unique (par exemple, # première-section). Ces identifiants sont utilisés dans les liens de la table des matières pour pointer vers chaque section. Pour faciliter la tâche RStudio propose néanmoins une petite interface graphique permettant de changer ces options plus facilement. Pour cela, cliquez sur l'icône en forme d'engrenage à droite du bouton Knit et choisissez Output Options.



II.2.4 Les styles et thèmes

Il est possible avec Rmarkdown d'importer un document word et importer les styles qui y sont et appliquer à notre document de sortie. Pour le faire, il faut faire appel à l'option des métadonnées `reference_docx`. Cette option prend comme valeur le chemin de notre fichier en question. Voici un exemple :

```

---
title: "Partie_BAMOGO"
author: "BAMOGO"
date: "2024-03-14"
output:
  word_document:
    toc: true
    number_sections: true
    reference_docx: garde.docx
  html_document:
    toc: true

```

II.2.5 L'option params

L'option `params` dans l'en-tête d'un chunk R Markdown est utilisée pour spécifier des paramètres externes que vous souhaitez passer au chunk lors de son exécution. Ces paramètres peuvent

être définis à l'extérieur du document R Markdown et injectés dans le document au moment de son exécution. Pour illustrer son importance nous pouvons utiliser l'exemple d'une base contenant des informations des individus. Ces individus sont dans divers secteurs activités tel que le commerce, l'agriculture, l'élevage, l'artisanat, etc. Nous pouvons utiliser l'option `parms` au niveau de l'en-tête pour générer un rapport pour chaque secteur d'activités. Il n'y pas de limite concernant le nombre de paramètres à insérer.

```
title: "Partie_BAMOGO"
author: "BAMOGO"
date: "2024-03-14"
output:
  word_document:
    toc: true
    number_sections: true
    reference_docx: garde.docx
    params:
      domaines: "agriculture"
      numero: 12
  html_document:
    toc: true
---
```

Application

— Préambule :

Comme application, nous avons écrit un code R markdown qui fait l'équivalent du publipostage sous Word.

À partir de l'échantillon ci-dessous :

Diplôme d'Étudiant

Université

Diplôme de l'Université Université

Ce document certifie que **Étudiant** a complété avec succès tous les requis nécessaires pour l'obtention du diplôme de **Diplôme** le **Date de Graduation**.

Université:

Université

Date d'Émission:

Date de Graduation

Félicitations pour cette réussite remarquable!

Signature:

Président de l'Université

Nous voulons remplacer automatiquement **Étudiant**, **Diplôme**, **Date de Graduation** et **Univer-**

sité par leurs valeurs correspondantes.

— Processus et codes :

À la base, nous avons un document nommé `templedi` comme échantillon.

```

4  output:
5    word_document: default
6    reference_docx: C:\Users\Larry\Desktop\S2\logiciel R\cours\Nouveau dossier\Doc2.docx
7    pdf_document: default
8    html_document:
9      df_print: paged
10  params:
11    student_name: Étudiant
12    degree: Diplôme
13    graduation_date: Date de Graduation
14    university: Université
15  ---
16  |
17  # Diplôme de l'Université `r params$university`
18
19  Ce document certifie que r params$student_name a complété avec succès tous les requis nécessaires pour l'obtention du diplôme de
20  r params$degree le r params$graduation_date.
21
22  Université:
23  `r params$university`
24
25  Date d'Émission:
26  `r params$graduation_date`
27
28  Félicitations pour cette réussite remarquable!
29
30  Signature:
31  _____
32  Président de l'Université
33
34
35
36
37
38
39
40

```

Dans ce document R Markdown, nous avons les paramètres `student_name`, `degree`, `graduation_date` et `university` qui sont insérés dans l'option `param`. Ce sont eux qui à chaque fois qu'ils sont modifiés dans l'entête YAML permettent de changer automatiquement le document.

Pour créer des documents automatiquement, nous avons écrit un script R qui permet de générer automatiquement les documents à l'aide de la commande `rmarkdown::render()` ; le script en question est le suivant :

```

library(rmarkdown)

# Exemple de Dataframe avec les informations des étudiants
students <- data.frame(
  name = c("Ndaye Fama", "Sandjo Larry", "Bargo Alferd"),
  degree = c("diplôme Analiste statisticien", "diplômes d'ingénieur statisticien-économiste", "master statistiques agricole"),
  graduation_date = c("15 juin 2024", "12 juillet 2023", "23 septembre 2025"),
  university = c("Ecole nationale de la statistiques et l'analyse économique", "Ecole nationale de la statistiques et l'analyse économique", "Ecole nationale de la statistiques et l'analyse économique")
)

# Chemin du fichier R Markdown template
template <- "C:\\Users\\Larry\\Desktop\\ISE 1\\S2\\Projet Statistique avec R\\Projet R Markdown\\Bamogo\\Option_param\\templedi.Rmd"

# Générer un diplôme pour chaque étudiant
for (i in 1:nrow(students)) {
  params <- list(
    student_name = students$name[i],
    degree = students$degree[i],
    graduation_date = students$graduation_date[i],
    university = students$university[i]
  )

  # Nom du fichier de sortie
  output_file <- paste0("diplome_", gsub(" ", "_", students$name[i]), ".docx")

  # Rendre le document R Markdown
  rmarkdown::render(
    input = template,
    output_file = output_file,
    params = params,
    envir = new.env(parent = globalenv())
  )
}

```

Comme vous le constatez la variable `student` contient les informations à entrer dans la partie de l'options `param`. La boucle `for` juste en bas permet de compiler chaque document pour les différentes informations.

— Sorties :

Après execution du script, on a les différentes sorties suivantes :

Diplôme d'Étudiant

École nationale de la statistiques et l'analyse économique

Diplôme de l'Université École nationale de la statistiques et l'analyse économique

Ce document certifie que **Sandjo Larry** a complété avec succès tous les requis nécessaires pour l'obtention du diplôme de **diplômes d'ingénieur statisticien-économiste** le **12 juillet 2023**.

Université:

École nationale de la statistiques et l'analyse économique

Date d'Émission:

12 juillet 2023

Félicitations pour cette réussite remarquable!

Signature:

Président de l'Université

Diplôme d'Étudiant

École nationale de la statistiques et l'analyse économique

Diplôme de l'Université École nationale de la statistiques et l'analyse économique

Ce document certifie que **Ndaye Fama** a complété avec succès tous les requis nécessaires pour l'obtention du diplôme de **diplôme Analiste statisticien** le **15 juin 2024**.

Université:

École nationale de la statistiques et l'analyse économique

Date d'Émission:

15 juin 2024

Félicitations pour cette réussite remarquable!

Signature:

Président de l'Université

Diplôme d'Étudiant

École nationale de la statistiques et l'analyse économique

Diplôme de l'Université École nationale de la statistiques et l'analyse économique

Ce document certifie que **Bargo Alferd** a complété avec succès tous les requis nécessaires pour l'obtention du diplôme de **master statistiques agricole** le **23 septembre 2025**.

Université:

École nationale de la statistiques et l'analyse économique

Date d'Émission:

23 septembre 2025

Félicitations pour cette réussite remarquable!

Signature:

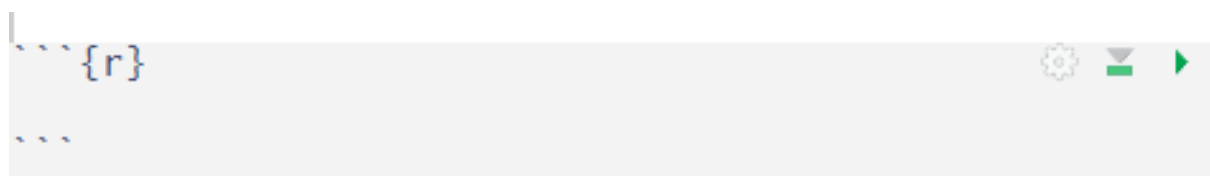
Président de l'Université

On constate alors que les différentes informations apparaissent chacune sur son document spécifique.

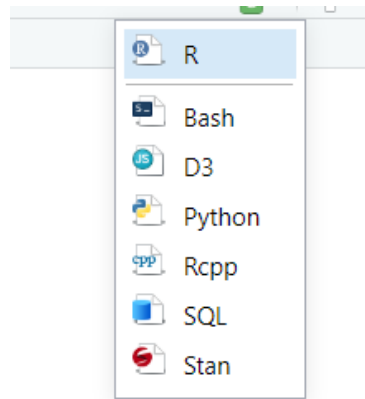
NB : Rassurez-vous que votre script R et votre document R markdown utilisé comme template soient dans le **même dossier**.

II.2.6 Chunks de code

En plus du texte libre au format Markdown, un document R Markdown contient, comme son nom l'indique, du code R. Celui-ci est inclus dans des blocs (chunks) délimités par la syntaxe suivante :



Comme cette suite de caractères n'est pas très simple à saisir, vous pouvez utiliser le menu Insert de RStudio et choisir le type de chunks que vous voulez



Ou utiliser le raccourci clavier **Ctrl+Alt+i**. Quand votre curseur se trouve dans un bloc, vous pouvez saisir le code R que vous souhaitez, l'exécuter, utiliser l'autocomplétion, exactement comme si vous vous trouviez dans un script R. Vous pouvez également exécuter l'ensemble du code contenu dans un bloc à l'aide du raccourci clavier **Ctrl+Maj+Entrée**. Le corps d'un document R Markdown comprend deux types de blocs, qu'on peut alterner librement, les chunks ont deux caractéristiques importantes à savoir : + Ces chunks peuvent être nommés (il est même recommandé de le faire); + Des options peuvent être spécifiées. Ces options (détaillées plus bas) permettent par exemple de ne pas faire figurer l'output du code dans le document final ou inversement de ne montrer que l'output du code et non le code l'ayant généré.

Quelques options des chunks Les options de chunk peuvent être personnalisées avec les options `knitr`, des arguments définis dans les `{ }` de l'en-tête d'un chunk. Ci-dessus, nous utilisons cinq arguments les plus utilisés :

Option	Description
<code>eval</code>	Permet de spécifier si le code contenu dans le chunk doit être évalué ou non.
<code>echo</code>	Contrôle l'affichage du code source R dans le document (TRUE pour afficher, FALSE pour ne pas afficher).
<code>include</code>	Indique si le contenu du chunk doit être inclus dans le document (TRUE pour inclure, FALSE pour exclure).
<code>message</code>	Contrôle l'affichage des messages générés par le code R (TRUE pour afficher, FALSE pour ne pas afficher).
<code>warning</code>	Contrôle l'affichage des avertissements générés par le code R (TRUE pour afficher, FALSE pour ne pas afficher).
<code>error</code>	Contrôle l'affichage des erreurs générées par le code R (TRUE pour afficher, FALSE pour ne pas afficher).
<code>fig.path</code>	Spécifie le chemin où les graphiques générés par le chunk seront sauvegardés.
<code>fig.cap</code>	Ajoute une légende à un graphique généré par le chunk.
<code>fig.width</code>	Spécifie la largeur en pouces du graphique généré par le chunk.
<code>fig.height</code>	Spécifie la hauteur en pouces du graphique généré par le chunk.
<code>fig.align</code>	Aligne le graphique généré par le chunk dans le document (par exemple, "center" pour centrer le graphique).
<code>out.width</code>	Contrôle la largeur de l'affichage du résultat (graphique, tableau, etc.) en pourcentage, pixels ou pouces.
<code>out.height</code>	Contrôle la hauteur de l'affichage du résultat (graphique, tableau, etc.) en pourcentage, pixels ou pouces.
<code>collapse</code>	Permet de regrouper le code source dans une seule ligne (TRUE pour regrouper, FALSE pour conserver la mise en forme).
<code>comment</code>	Ajoute un commentaire au chunk.
<code>cache</code>	Active la mise en cache des résultats du chunk pour une utilisation ultérieure.
<code>cache.path</code>	Spécifie le chemin où les résultats mis en cache seront sauvegardés.
<code>results</code>	Contrôle l'affichage des résultats du code (par exemple, "hide" pour ne pas afficher, "asis" pour afficher tel quel).

Consultez le Guide de référence de R Markdown pour une liste complète des options de chunk knitr. Il est possible aussi de définir des options globales qui s'appliquent à chaque chunk de votre fichier, appelez knitr : :opts_chunks et dans un chunk de code. Knitr traitera chaque option que vous transmettez à knitr : :opts_chunkset comme un défaut global pouvant être écrasé dans les en-têtes de chunk individuels.

III Personnalisation d'un document R markdown

III.1 Les notes de bas de page

Markdown nous permet d'éditer des notes de bas de page. Il suffit d'ajouter tout simplement dans votre texte un numéro d'annotation, et reprenez ce numéro en bas de votre page dans une note de bas de page. Markdown créera alors automatiquement une ligne. L'insertion de notes de bas de page numérotées s'effectue à l'aide de crochets (`[]`) contenant un accent circonflexe et une référence qui peut être soit un nombre, soit un texte (mais sans espace ou autre caractère blanc).

Application

Le numéro de la note de bas de page est un lien qui renvoie vers la note ¹. Une flèche de retour est proposée pour ramener au texte lorsque le document créé est un document html ².

III.2 Ajout de bibliographie

Pour insérer une bibliographie dans un document R Markdown, on peut utiliser des fichiers de style CSL (Citation Style Language) et des logiciels comme Zotero ou Mendeley pour gérer vos références bibliographiques. Voici une méthode générale pour le faire :

Utilisez un logiciel de gestion de références bibliographiques tel que Zotero, Mendeley ou EndNote pour collecter et organiser vos références. Assurez-vous d'exporter votre bibliographie dans un format compatible CSL, tel que BibTeX (.bib) ou CSL JSON (.json).

Téléchargez un fichier de style CSL correspondant à vos besoins. CSL est un langage de style de citation qui contrôle le format de citation et de bibliographie dans les documents. Vous pouvez trouver des styles CSL sur le site officiel de CSL <https://citationstyles.org/>.

Configurer votre document R Markdown : Dans votre document R Markdown, vous pouvez spécifier le style de citation et la bibliographie à utiliser dans le YAML en-tête. Assurez-vous que le chemin vers votre fichier de bibliographie (biblio.bib) et votre fichier de style CSL (style.csl).

Configurer votre document R Markdown : Dans votre document R Markdown, vous pouvez spécifier le style de citation et la bibliographie à utiliser dans le YAML en-tête. Assurez-vous que le chemin vers votre fichier de bibliographie (biblio.bib) et votre fichier de style CSL (style.csl).

III.3 Insertion de tableau

Pour insérer des tableaux basiques dans un document RMarkdown, on peut utiliser la syntaxe Markdown pour créer des tableaux simples.

La syntaxe Markdown permet de créer un tableau en utilisant des barres verticales `|` pour séparer les colonnes et des tirets `-` pour séparer les lignes.

Pour personnaliser votre tableau en ajoutant des éléments Markdown supplémentaires, utilisez deux points : dans les barres verticales pour aligner le texte dans les colonnes. Par exemple : `—` : pour centrer, `:—` pour aligner à gauche, `—` : pour aligner à droite.

Après avoir tricoté le document RMarkdown, le tableau sera formaté correctement dans la sortie finale, que ce soit dans un fichier PDF, HTML ou autre, en fonction du format que vous avez choisi pour votre document. Pour des fonctionnalités plus avancées ou de tableaux plus complexes, on pourra également utiliser des packages R spécifiques, comme `kable()` de `knitr`, pour créer et personnaliser des tableaux à partir de données R.

1. L'intérêt de la note de bas de page

2. Si le document est html

Application

Par exemple, lorsqu'on entre le code ci-dessous,

```
**Application**
```

<u>Nom</u>	<u>Prenom</u>	<u>Classe</u>
BAMOGO	Rasmane	ISE-MATH
SANDJO	Larry	ISE-MATH
DIOP	Cherif Ousseynou	ISEP
ZINABA	Albert	ISE-ECO

On obtient la sortie suivante.

Nom	Prenom	Classe
BAMOGO	Rasmane	ISE-MATH
SANDJO	Larry	ISE-MATH
DIOP	Cherif Ousseynou	ISEP
ZINABA	Albert	ISE-ECO

Il est aussi possible de faire des tableaux à l'aide des codes R à partir des chunks. Les fonctions telles que `kable()` de la librairie `knitr` et `tbl_summary()` de `gtsummary` sont vivement conseillées.

III.4 Insertion de lien

R markdown permet de créer des liens qui facilite la navigation vers le web ,local ou même dans notre document.

— Insertion du lien web :

La syntaxes est la suivante entre crochet le titre du lien suivis d'une parenthèse contenant le lien web.

Exemple :

Le code suivant :

```
[Lien vers zoreto](https://www.zotero.org/)
```

Permet d'avoir le lien suivant : Lien vers zoreto

— Lien vers un fichier local :

La syntaxe est la suivante ; entre crochet le titre du lien suivis d'une parenthèse contenant le chemin d'accès au document

Exemple :

Voici une photo du code :

```
[lien vers le TP](C:\\Users\\PC\\Desktop\\S2\\ADD\\ACM\\TP\\TPdeACF.docx)
```

lien vers le TP

— Lien vers un titre du document :

La syntaxe est la suivante ; entre crochet contenant le titre du document suivi d'une parenthèse contenant le titre mais le titre doit être début par un # :

```
[Retour vers l'insertion](# Insertion d'un lien dans le document rmarkdown)
```

Retour vers l'insertion

Dans tous les cas le titre n'est pas obligatoire mais au cas où il n'y a pas de titre les crochets doivent être mis et laisser vide

III.5 Insertion d'images

Pour insérer des images dans un document RMarkdown, la syntaxe suivante :

```
.
```

Pour ce faire il faut s'assurer que le nom de l'image que vous avez mis est le bon, par exemple ici "imge.jpg" dans l'exemple précédent.

Lorsque vous rendrez votre document RMarkdown, l'image sera incorporée dans le document rendu. Assurez-vous également que vous avez les packages nécessaires pour le rendu de votre document RMarkdown avec des images, comme knitr et rmarkdown, installés et chargés dans votre environnement R.

Pour ajouter une description de l'image, pour pouvez insérer la description dans les crochets. Par exemple ["description"].

Application

Le code R markdown suivant permet d'obtenir la sortie ci-dessous :

```
![Une sortie de promotion à l'ENSAE](ENSAE_Image.JPG)
```

Il est aussi possible d'insérer des images à partir des chunks à l'aide de la fonction `include_graphics()` de la librairie `knitr`. En procédant ainsi, il est alors possible de contrôler la largeur de l'image à l'aide de l'option de chunk `out.width` et la longueur de par l'option `out.height`.



FIGURE 1 – Une sortie de promotion à l'ENSAE

III.6 Equations mathématiques

Pour insérer des équations mathématiques dans un document RMarkdown, on peut utiliser la syntaxe LaTeX, qui est largement utilisée pour la mise en forme mathématique. Quelques façons d'insérer des équations mathématiques :

III.6.1 Equations en ligne

Pour inclure des équations mathématiques directement dans le texte, on peut les entourer de symboles \$.

Exemple :

Lorsque l'on insère le code latex suivant $f(x) = x+y$ on obtient la sortie suivante : $f(x) = x+y$

III.6.2 Équations centrées

Pour afficher des équations mathématiques sur leur propre ligne, vous pouvez les entourer de symboles \$\$.

Exemple :

Le code suivant permet d'obtenir la sortie plus bas il utilise la balise LaTeX `\begin{aligned} \dots \end{aligned}`

```


$$\begin{aligned}
&\sum_{i=1}^n x_i &= -\frac{a_{n-1}}{a_n} \\
&\sum_{1 \leq i < j \leq n} x_i x_j &= +\frac{a_{n-2}}{a_n} \\
&\sum_{1 \leq i < j < k \leq n} &= -\frac{a_{n-3}}{a_n} \\
&\vdots & \\
&\sum_{1 \leq i_1 < i_2 < \dots < i_P \leq n} x_{i_1} x_{i_2} \dots x_{i_P}
\end{aligned}$$


```



```

\vdots&\\
\prod_{i=1}^n x_{i}&=(-1)^n \frac{a_{0}}{a_{n}}
\end{aligned}$$$

```

Sortie :

$$\begin{aligned}
 \sum_{i=1}^n x_i &= -\frac{a_{n-1}}{a_n} \\
 \sum_{1 \leq i < j \leq n} x_i x_j &= +\frac{a_{n-2}}{a_n} \\
 \sum_{1 \leq i < j < k \leq n} &= -\frac{a_{n-3}}{a_n} \\
 &\vdots \\
 \sum_{1 \leq i_1 < i_2 < \dots < i_p \leq n} x_{i_1} x_{i_2} \dots x_{i_p} &= (-1)^p \frac{a_{n-p}}{a_n} \\
 &\vdots \\
 \prod_{i=1}^n x_i &= (-1)^n \frac{a_0}{a_n}
 \end{aligned}$$

Il est aussi possible d'utiliser les codes LaTeX `\begin{equation} \dots \end{equation}`

Exemple :

$$E = mc^2 \quad (1)$$

III.6.3 Notation LaTeX

On peut également utiliser la notation LaTeX pour écrire des expressions mathématiques plus complexes, telles que les fractions, les indices, les exposants, les racines carrées, etc.

Exemple : Une division euclidienne

Avec le code latex suivant on obtient une la division euclidienne plus bas.

```

%%{=tex}
\renewcommand{\arraystretch}{1.2} \renewcommand{\arraycolsep}{2pt}
\begin{array}{rrrrrrrrrr} A=&4x^4&-4x^3&+5x^2&-4x&+1 & 2x^3&-3x^2&-x&+1 =B \\
&2x^4&-3x^3&-x&+1&=B & \backslash\backslash \\
&\backslash\backslash & -4x^4&+6x^3&+2x^2&-2x & \backslash\backslash \\
&\backslash\backslash & \backslash\backslash & 2x^3&+7x^2&-6x&+1 & \backslash\backslash \\
&\backslash\backslash & \backslash\backslash & -2x^3&+3x^2&+x&-1 & \backslash\backslash \\
&\backslash\backslash & \backslash\backslash & 10x^2&-5x&=R_1 & \backslash\backslash \\
&\backslash\backslash & \backslash\backslash & \end{array}
%

```

$$\begin{array}{r|l}
 A = & 4x^4 - 4x^3 + 5x^2 - 4x + 1 \\
 & -4x^4 + 6x^3 + 2x^2 - 2x \\
 \hline
 & 2x^3 + 7x^2 - 6x + 1 \\
 & -2x^3 + 3x^2 + x - 1 \\
 \hline
 & 10x^2 - 5x = R_1
 \end{array} \quad \begin{array}{l}
 2x^3 - 3x^2 - x + 1 = B \\
 2x + 1 = Q_1
 \end{array}$$

III.7 Packages et extensions

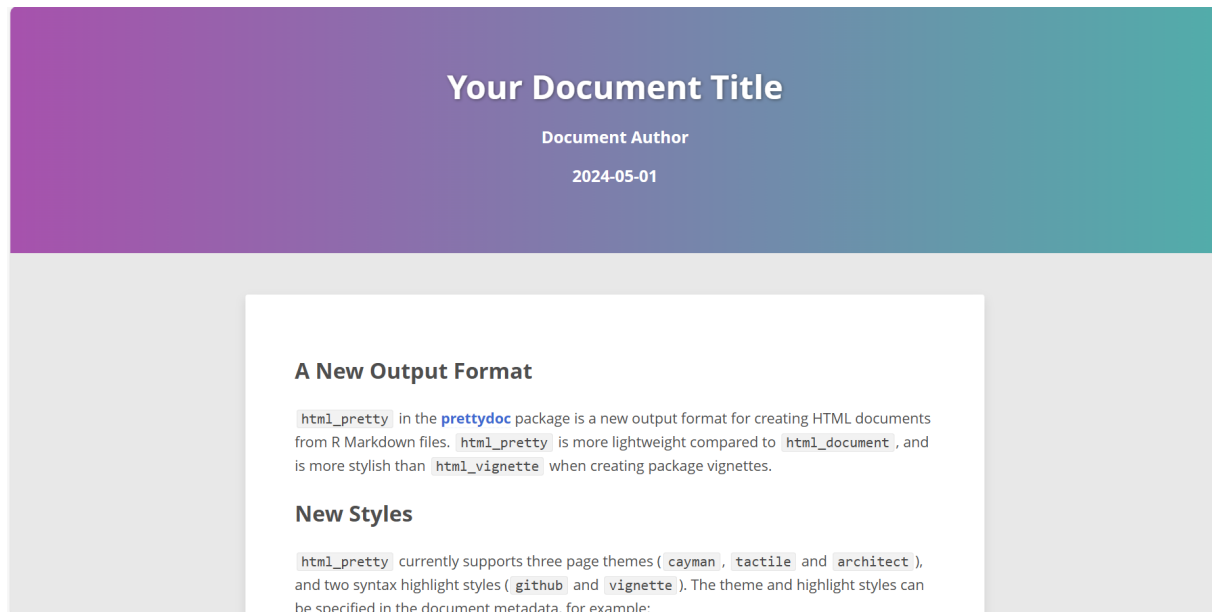
La création d'un modèle (ou Template) dans RMarkdown vous permet de définir une structure de document prédéfinie avec des paramètres par défaut, des styles personnalisés, des en-têtes,

des pieds de page, etc. Voici comment créer un modèle RMarkdown :

III.7.1 Création du modèle

1. Commencez par créer un nouveau document RMarkdown dans RStudio.
2. Personnalisez ce document selon vos besoins : définissez les en-têtes, les pieds de page, les styles CSS, les balises LaTeX personnalisées, etc.
3. Enregistrez ce document dans un répertoire spécifique que vous pouvez facilement retrouver.

Exemple d’une sortie d’un modèle :

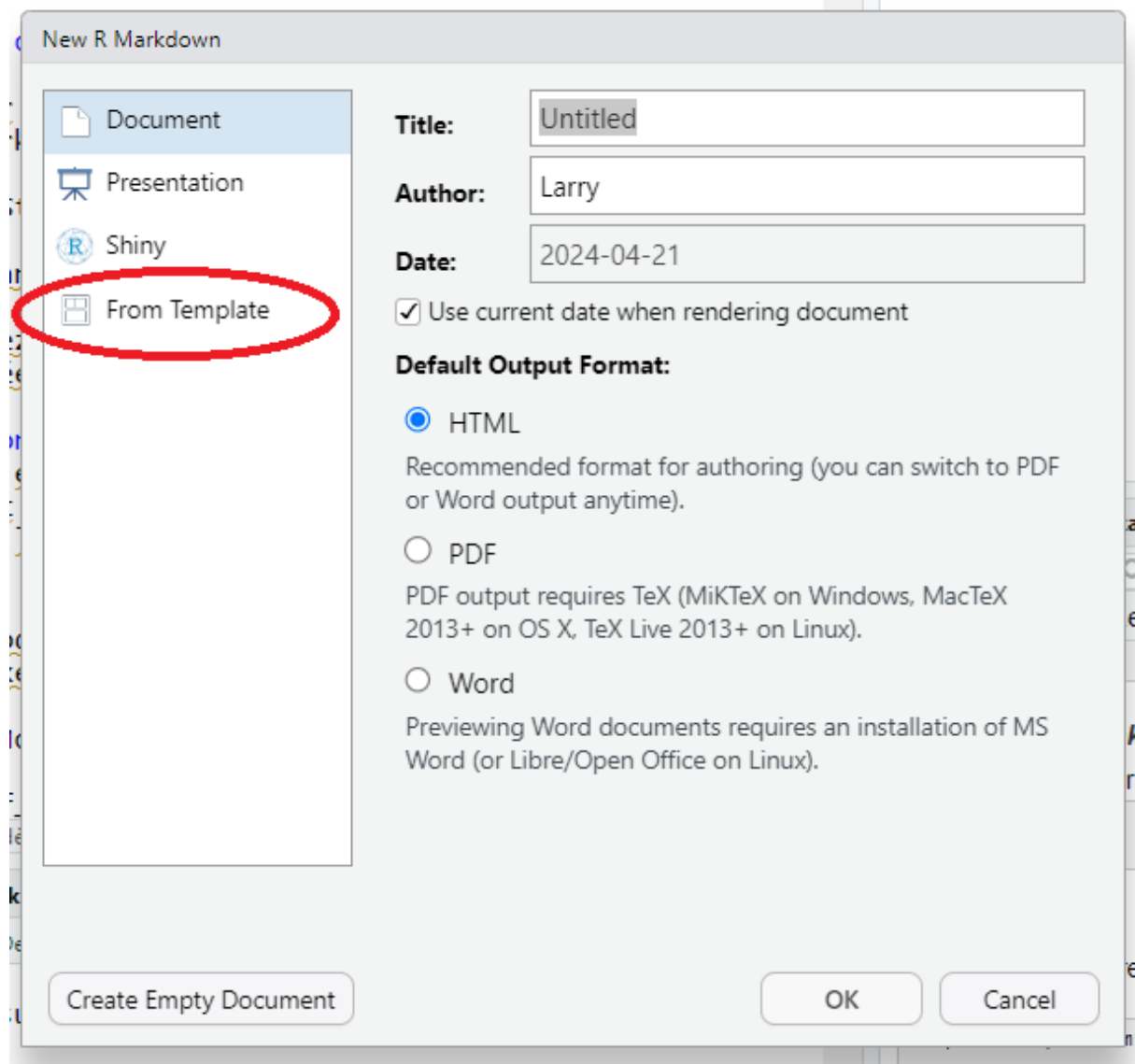


Ce type de modèle peut être facilement obtenu sur le site : <https://github.com/yixuan/prettydoc/?tab=readme-ov-file>

III.7.2 Utilisation du modèle

Pour utiliser ce modèle lors de la création d’un nouveau document RMarkdown :

1. Ouvrez RStudio.
2. Sélectionnez “File” > “New File” > “R Markdown...”.
3. Choisissez “From Template” et sélectionnez le modèle que vous avez créé.



III.7.3 Personnalisation supplémentaire

Nous pouvons également personnaliser davantage notre modèle en y ajoutant des paramètres YAML spécifiques, tels que des options pour les citations bibliographiques, le format de sortie, etc.

Exemple de modèle YAML :

Voici un exemple de YAML pour un modèle RMarkdown :

— title : “Mon Modèle RMarkdown”

output : pdf_document :

toc : true

number_sections : true

fig_caption : true

html_document :

toc : true

toc_float : collapsed : false

code_folding : hide —

Utilisation de données dynamiques :

Vous pouvez également créer des modèles qui intègrent des données dynamiques. Par exemple, vous pouvez utiliser des paramètres YAML pour spécifier des titres ou des noms d'auteurs qui seront remplis à la création du document.

En résumé, la création d'un modèle RMarkdown implique la personnalisation d'un document RMarkdown selon vos besoins, puis la sauvegarde de ce document dans un répertoire spécifique. Vous pouvez ensuite utiliser ce modèle comme base pour créer de nouveaux documents RMarkdown en sélectionnant "From Template" lors de la création d'un nouveau document dans RStudio.

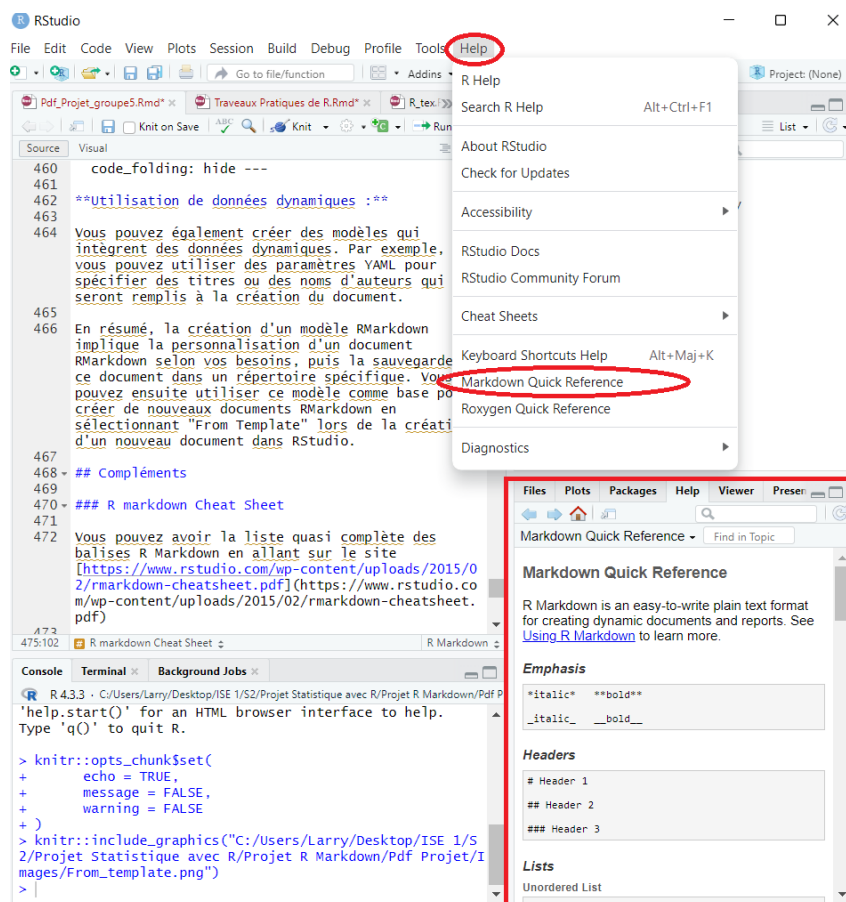
III.8 Compléments

III.8.1 R markdown Cheat Sheet

Vous pouvez avoir la liste quasi complète des balises R Markdown en allant sur le site

<https://www.rstudio.com/wp-content/uploads/2015/02/rmarkdown-cheatsheet.pdf>

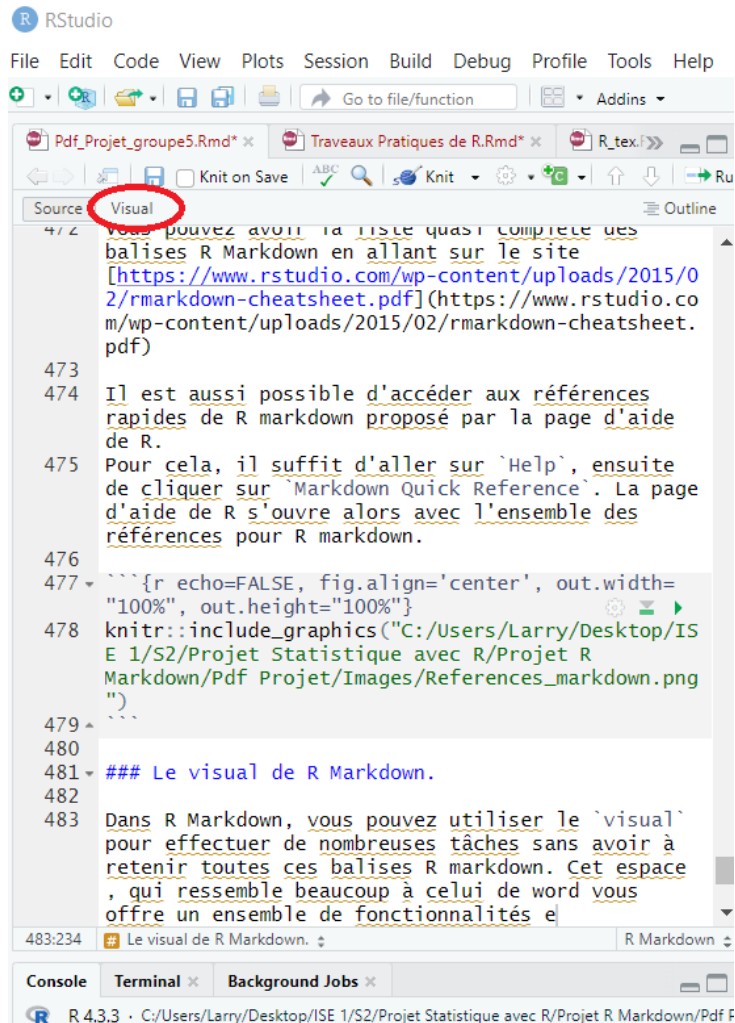
Il est aussi possible d'accéder aux références rapides de R markdown proposé par la page d'aide de R. Pour cela, il suffit d'aller sur Help, ensuite de cliquer sur Markdown Quick Reference. La page d'aide de R s'ouvre alors avec l'ensemble des références pour R markdown.



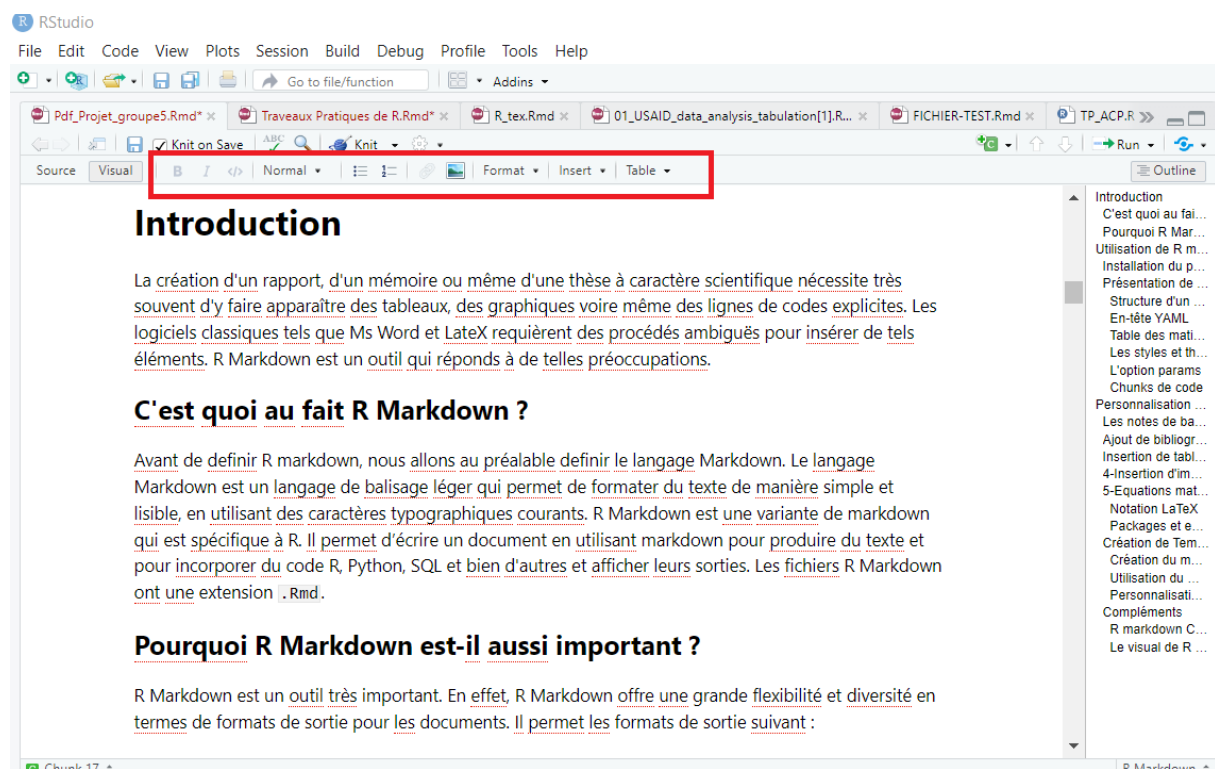
III.8.2 Le visual de R Markdown.

Dans R Markdown, vous pouvez utiliser le `visual` pour effectuer de nombreuses tâches sans avoir à retenir toutes ces balises R markdown. Cet espace, qui ressemble beaucoup à celui de word, vous offre un ensemble de fonctionnalités qui écrit des balises R Markdown à votre place.

Pour accéder, il suffit de cliquer sur `Visual` à côté de `Source`.



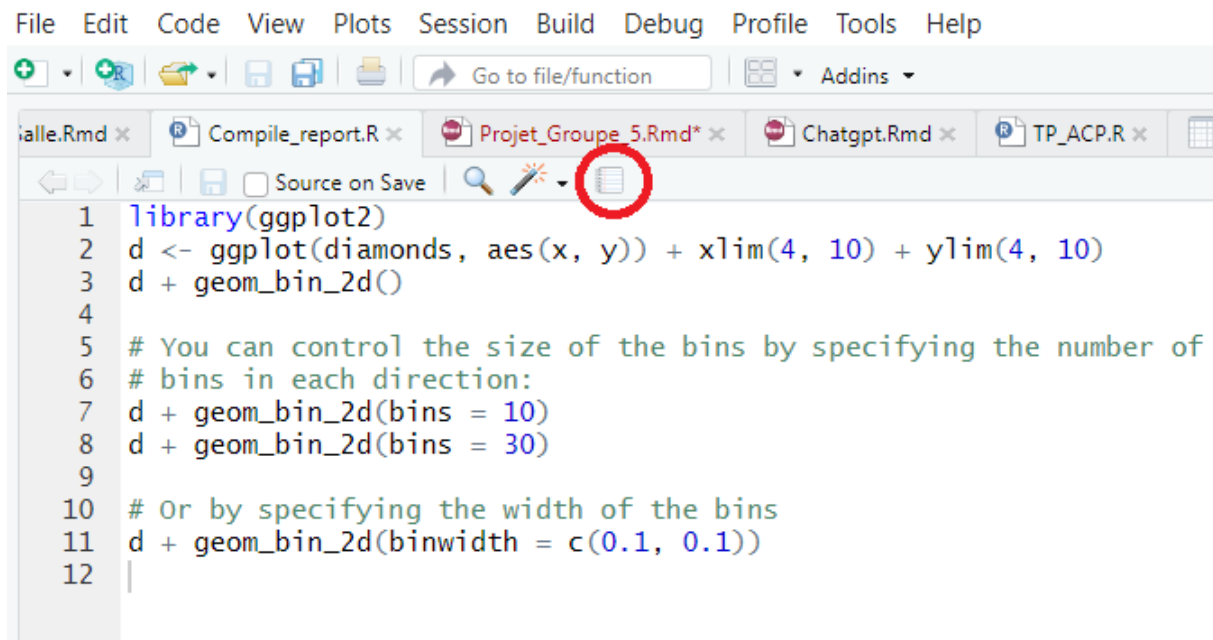
Vous pouvez ensuite utiliser l'ensemble des fonctionnalités telles que mettre en gras, mettre en italique, insérer des images, des tableaux, etc.



III.8.3 Rapport rapide à partir d'un script R

Si vos analyses sont présentes dans un script R et que ce script contient tout le nécessaire pour la réalisation de votre analyse (i.e. chargement des données et des packages requis), vous pouvez très facilement réaliser un rapport rapide au format HTML, Word ou PDF, contenant à la fois votre code et les sorties associées.

Il suffit de cliquer sur l'icône compile report dans le quadrant supérieur gauche.



Une autre façon de faire cela est d'utiliser la commande `rmarkdown::render("nom_script.R")`

IV Formats de sortie avec R Markdown

IV.1 Creation beamer

Une fois que le logiciel ouvert, vous pouvez suivre les étapes suivantes pour créer le beamer :

```
past <- here::here()
image_1 <- paste0(past, "/image_1.png")
image_2 <- paste0(past, "/image_2.png")
image_3 <- paste0(past, "/image_3.png")
```

IV.1.1 Etapes 1 :

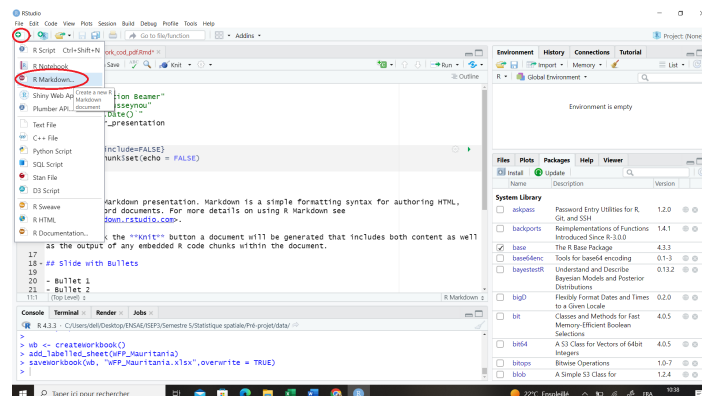


FIGURE 2 – première étape création beamer

IV.1.2 Etapes 2 :

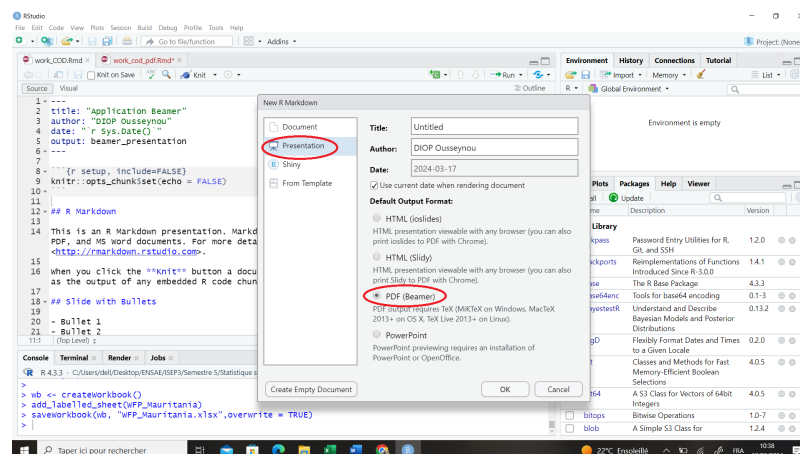


FIGURE 3 – deuxième étape création beamer

IV.1.3 Etapes 3 :

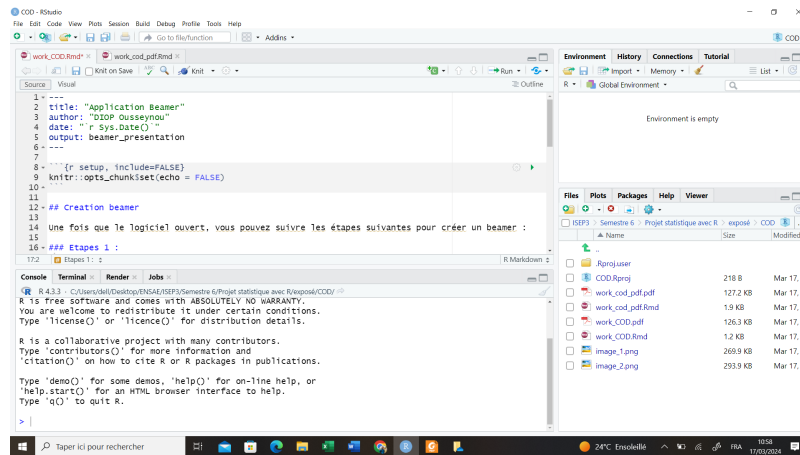


FIGURE 4 – troisième étape création beamer

IV.2 Personnaliser le beamer

Après la création du beamer, le fichier a un format simple. Il est possible de personnaliser la sortie.

IV.2.1 Thèmes :

Le Beamer offre une gamme de thèmes préconfigurés pour personnaliser la présentation, parmi lesquels on retrouve : AnnArbor, Antibes, Bergen, Berkeley, Berlin, CambridgeUS, Frankfurt, etc.

De plus, il propose également une palette de couleurs prédéfinies telles que : bleu, vert, cyan, magenta, jaune, noir, blanc, etc.

La matrice des thèmes et des couleurs est définie sur le site : *Thèmes & couleurs beamer*.

IV.2.2 Thème et couleur :

Pour ajouter un thème et une couleur, il suffit d'ajouter le syntaxe au niveau de l'output (YAML) :

- theme : “nom_theme”
- colortheme : “nom_couleur”

IV.2.3 Exemple :

title : “Application Beamer”

author : “DIOP Ousseynou”

date : “2024-05-23”

output : ”

beamer_presentation : ”

theme: "Madrid"

colortheme: "dolphin"

IV.3 Niveau coulissant :

Le niveau de coulissant correspond à la hiérarchie des titres dans une présentation Beamer. Dans RMarkdown, le paramètre *slide_level* contrôle cette structuration en spécifiant le niveau de titre qui doit générer une nouvelle diapositive dans la sortie Beamer.

IV.3.1 Remarque

Par défaut, *slide_level* est fixé à 2, ce qui signifie que les titres de niveau 1 (#) et de niveau 2 (##) créeront une nouvelle diapositive, tandis que les sous-titres de niveau 3 et supérieur seront inclus dans la même diapositive que leur titre parent.

IV.3.2 Exemple :

title : "Application Beamer"

author : "DIOP Ousseynou"

date : "2024-05-23"

output : "

beamer_presentation :"

theme: "Madrid"

colortheme: "dolphin"

slide_level : 2

IV.4 Quelques statistiques

La source de données utilisée provient de ACLED et concerne les incidents survenus dans les pays du G5 Sahel, entre autres.

```
past <- here::here()
past_data <- paste0(past, "/ACLED-Western_Africa (1).csv")
data <- read.csv("../Document_Rmarkdown/ACLED-Western_Africa (1).csv")
data_filter <- dplyr::filter(
  data,
  (type == "Violence against civilians" | type=="Battles") & (pays=="Burkina Faso" | p
tableau <- table(data_filter$pays, data_filter$type)
```

IV.4.1 Résultat

```
print(tableau)
```

```
##
##           Battles Violence against civilians
## Burkina Faso      1742                2521
## Mali              2203                2704
## Mauritania         12                 16
## Niger              577               1324
## Senegal            44                  38
```

TABLE 2 – Evenements G5 Sahel

	Battles	Violence against civilians
Burkina Faso	1742	2521
Mali	2203	2704
Mauritania	12	16
Niger	577	1324
Senegal	44	38

IV.4.2 Mise en forme avec KableExtra

```
library(kableExtra)
library(dplyr)

#conversion du tableau en un objet kable
tableau_kable <- as.data.frame.matrix(tableau)
table_form <- tableau_kable %>%
  kbl(caption="Evenements G5 Sahel") %>%
  kable_classic(full_width = F, html_font = "Cambria")
```

IV.4.3 Résultat

```
table_form
```

IV.5 Un peu de statistiques spatiales(1)

```
library(tidyverse)
library(sf)
library(tmap)
library(leaflet)

past <- here::here()
bkf <- read_sf("../Document_Rmarkdown/BKF/gadm41_BFA_2.shp")
tmap_mode("plot")
bkf_1 <- tm_shape(bkf) +
  tm_polygons() +
  tm_text("NAME_2", size = 0.5, col = "black", alpha = 0.8) +
  tm_layout("Burkina Faso", inner.margins=c(0,0,.1,0), title.size=.8) +
  tm_compass(type = "arrow", position = c("left", "top"))
```

IV.5.1 Résultat

```
bkf_1
```



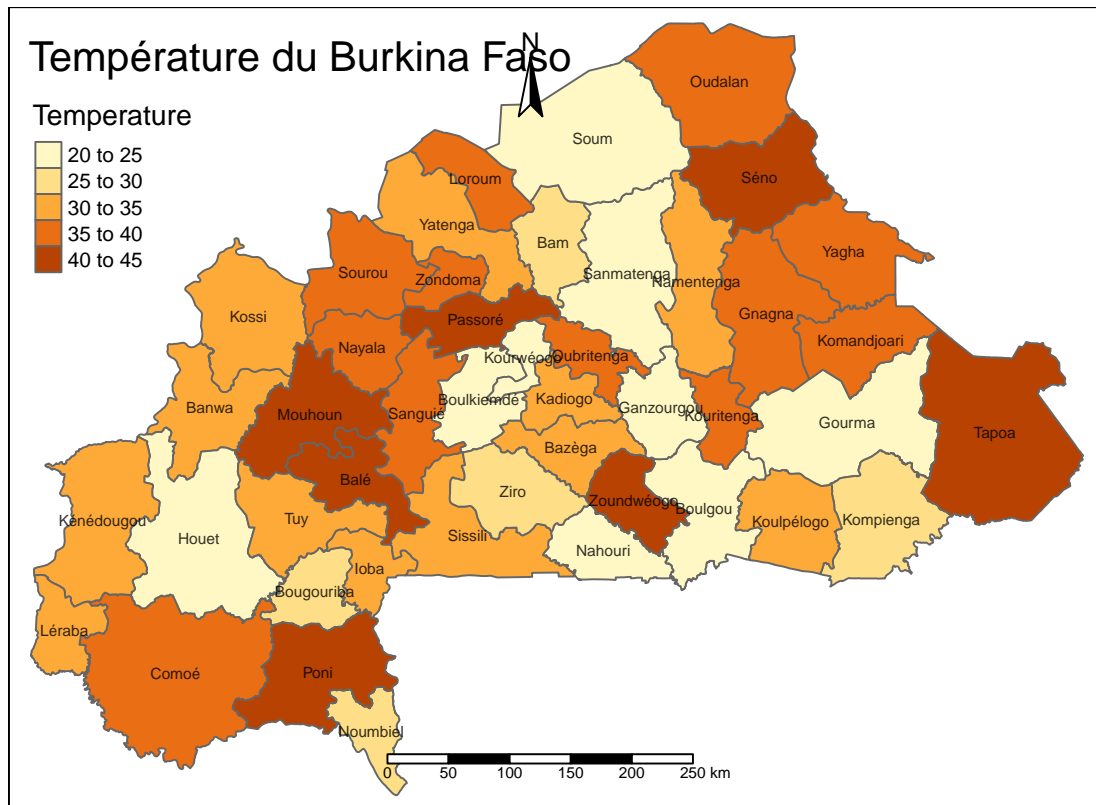

IV.6 Un peu de statistiques spatiales(2)

```
library(openxlsx)

way<- paste0(past, "/BKF//température.xlsx")
temperature <- read.xlsx("../Document_Rmarkdown/BKF/température.xlsx")
temperature <- data.frame(temperature)
bkf_2<- merge(bkf, temperature, by = "NAME_2", all.x = TRUE)
bkf_2_T <- tm_shape(bkf_2) +
  tm_polygons("Temperature", title = "Temperature") +
  tm_borders("white", lwd = 0.5) +
  tm_text("NAME_2", size = 0.5, col = "black", alpha = 0.8)+
  tm_layout(title = "Température du Burkina Faso")+
  tm_compass(type = "arrow", position = c("left", "top"))+
  tm_scale_bar(position = c("center", "bottom"))
```

IV.6.1 Résultat

```
bkf_2_T
```



IV.7 Autres fonctionnalités

Il y a d'autres fonctionnalités de beamer comme : générer la table des matières, les numéros des sections, les options de sorties, ajouter une image, les listes etc. Ces fonctionnalités sont discutées plus en détail dans les présentations générales de Rmarkdown.

V R Markdown et LaTeX

Si vous souhaitez concevoir des documents pdf très attrayant, comme celui-ci d'ailleurs, il est conseillé d'associer vos balises R markdown aux balises LaTeX. Nous vous présentons ici quelques balises LaTeX simple qui peuvent faire la différence dans vos documents R markdown. Tous les packages LaTeX nécessaires doivent être inclus dans l'entête YAML à l'aide de `header-include`: où vous devez lister ces packages.

Les pages de garde :

Pour insérer une page de garde dans votre document Rmd, vous devez avoir une page de garde faite à partir de Ms word par exemple. Cette page de garde doit être au **format pdf** et être **contenu dans le dossier où se trouve votre script R markdown**.

Avant d'inclure la page de garde, vous devez vous assurer d'inclure le package LaTeX qui permet de le faire au niveau de l'entête YAML. Il s'agit du package `\usepackage{pdfpages}`.

La balise permettant d'inclure la page de garde est `\includepdf{nom_de_la_page_de_garde}`.

La table des matières :

Nous avons déjà vu comment insérer la table de matière avec R markdown, mais il est conseillé d'utiliser la balise LaTeX `\tableofcontents` qui permet de générer une table de matière plus attrayante et qui donne la possibilité de numéroter automatiquement les parties de notre document.

Liste des tableaux et liste des figures :

Il est aussi possible d'inclure une liste des tableaux et des figures avec des balises latex `\listoftables` et `\listoffigures`.

Insérer une nouvelle page :

Vous l'aurez remarqué, l'insertion d'une nouvelle page dans R markdown se fait très souvent à l'aide de la balise latex `\newpage`.

Mettre un texte en couleur :

Dans vos document R markdown vous pouvez mettre vos titres ou vos textes en couleur à l'aide de la syntaxe `\textcolor{nom_couleur}{texte_ou_titre}`. Par exemple **Le texte suivant s'écrit en couleur bleu**.

Encadrer un texte :

Pour encadrer votre block de texte, vous pouvez utiliser le block de code `\begin{tcolorbox}[colframe = couleur_du_block, colback = couleur_background, title = titre_du_paragraphe]...(votre texte)...\end{tcolorbox}`.

Par exemple :

Block de texte

Le block de texte suivant nommé 'Block de texte' s'affichera dans un frame de couleur bleue et de fond gris.

Vous pouvez aussi utiliser le block de code `\begin{framed}...\end{framed}` du package latex `\usepackage{framed}`

Par exemple :

Ceci est un bloc de texte encadré.

Alignement de texte :

Vous pouvez aussi choisir l'alignement de votre texte gauche, droite ou centré à l'aide des blocks de code latex `\begin{flushleft}...\end{flushleft}`, `\begin{flushright}...\end{flushright}` et `\begin{center}...\end{center}`.

Exemple :

Texte centré.

Texte aligné à gauche.

Texte aligné à droite.

Labelisation d'équations :

On peut donner un nom à une equation et la labeliser à l'aide de la commande LaTeX `\label{eq:nom_equation}`.

Exemple :

$$E = mc^2 \tag{2}$$

On utilise la commande `\eqref{eq:nom_equation}` pour faire apparaître la référence de l'équation comme ci-dessous.

Comme nous pouvons le voir dans l'équation (2), l'énergie est proportionnelle à la masse.

En cliquant sur la référence, on peut revenir voir l'équation.

Entête et pieds de pages avec LaTeX

Il est possible de ce demander en consultant ce document comment ajouter les en-têtes et les pieds de pages dans son document R markdown pour les sorties pdf.

En effet cela nécessite une suite de codes LaTeX notamment `\renewcommand{}`, `\headrulewidth`,...

Le code LaTeX utilisé pour cela est le suivant :

```

\documentclass[=tex]{report}
\listoftables
\listoffigures
\newpage
\documentclass[=tex]{report}
\renewcommand{\headrulewidth}{1.5pt}
\let\oldheadrule\headrule
\renewcommand{\headrule}{\color{blue}\oldheadrule}% Add colour to \headrule
\renewcommand{\footrulewidth}{1.5pt}
\let\oldfootrule\footrule
\renewcommand{\footrule}{\color{blue}\oldfootrule}% Add colour to \headrule
\pagestyle{fancy}% Default page style
\lhead{\scriptsize[R-Markdown]}% left head entête gauche
\chead{}% entête Au centre
\rhead{\textcolor{black}{2024}}% right head entête à droite
\lfoot{\textcolor{black}{ENSAE}}% pieds de page gauche
\rfoot{Groupe 5}% Au centre du pied de page
%
%\section*{Introduction}\addcontentsline{toc}{section}{Introduction}
% \l'etoile '*' permet de ne pas compter une section comme un titre

```

NB :

Toutes ces mises en formes ne seront visibles que pour les sorties sous le format pdf.

Voici une listes de balises LaTeX que vous pouvez utiliser pour vos documents R markdown et leur fonctionnement respectifs.

Catégorie	Balise LaTeX	Description
Texte	<code>\textbf{}</code>	Texte en gras
	<code>\textit{}</code>	Texte en italique
	<code>\underline{}</code>	Texte souligné
	<code>\texttt{}</code>	Texte en police à chasse fixe
Mathématiques	<code>\$... \$</code>	Formule en ligne
	<code>\$\$... \$\$</code>	Formule sur une ligne séparée
	<code>\begin{equation} ... \end{equation}</code>	Formule numérotée
	<code>\frac{num}{den}</code>	Fraction
	<code>\sqrt{}</code>	Racine carrée
	<code>\sum</code>	Somme
	<code>\int</code>	Intégrale
	<code>\lim</code>	Limite
Symboles	<code>\alpha, \beta, \gamma, \ldots</code>	Lettres grecques
	<code>\pm, \mp, \times, \div</code>	Symboles mathématiques
	<code>\infty</code>	Infini
	<code>\partial</code>	Dérivée partielle
	<code>\nabla</code>	Nabla (gradient)
Tableaux	<code>\begin{tabular} ... \end{tabular}</code>	Environnement de tableau
	<code>\hline</code>	Ligne horizontale
	<code>\multicolumn{n}{c}{texte}</code>	Fusionner des colonnes
Alignement	<code>\begin{align} ... \end{align}</code>	Alignement des équations
	<code>\\</code>	Saut de ligne dans les équations alignées
Fonctions et Textes	<code>\mathrm{}</code>	Texte en mode mathématique
	<code>\mathbf{}</code>	Texte en gras en mode mathématique
	<code>\mathit{}</code>	Texte en italique en mode mathématique
Matrices	<code>\begin{matrix} ... \end{matrix}</code>	Matrice
Délimiteurs	<code>\begin{pmatrix} ... \end{pmatrix}</code>	Matrice entre parenthèses
	<code>\left(... \right)</code>	Parenthèses automatiques
	<code>\left[... \right]</code>	Crochets automatiques
	<code>\left\{ ... \right\}</code>	Accolades automatiques
Flèches	<code>\rightarrow, \leftarrow</code>	Flèches droite, gauche
	<code>\leftrightarrow</code>	Flèche double
	<code>\Rrightarrow, \Lleftarrow</code>	Double flèches droite, gauche
Ensembles	<code>\mathbb{R}, \mathbb{N}, \mathbb{Z}</code>	Ensembles des réels, naturels, entiers
Accolades et Barres	<code>\overline{ }, \underline{ }</code>	Ligne au-dessus, ligne en-dessous
	<code>\overbrace{ }, \underbrace{ }</code>	Accolade au-dessus, accolade en-dessous
Accents	<code>\hat{ }, \tilde{ }, \bar{ }</code>	Chapeau, tilde, barre
Fonctions Trigonométriques	<code>\sin, \cos, \tan, \cot</code>	Fonctions trigonométriques
	<code>\arcsin, \arccos, \arctan</code>	Fonctions trigonométriques inverses
Logique	<code>\forall, \exists, \neg</code>	Quantificateurs et négation
	<code>\wedge, \vee, \implies</code>	Et, ou, implique

FIGURE 5 – Balises LaTeX

VI R Markdown, HTML et CSS

Si vous souhaitez avec des documents interactifs, il est conseillé de privilégier les sorties HTML qui supportent les graphiques et tableaux dynamiques, ce qui n'est pas le cas des sorties pdf par exemple.

Vous pouvez utiliser du code CSS pour faire toutes mises en formes de votre document HTML ; vous pourrez même ainsi créer vos propres templates.

Par exemple, à l'aide du code CSS suivant :

```
<style>
  body {
    font-family: 'Arial', sans-serif;
  }
  h1{
    color: #2E8B57;
  }
  h2 {
    color: blue
  }
  h3 {
    color: red
  }
  p {
    font-size: 16px;
  }
  .custom-table {
    width: 100%;
    border-collapse: collapse;
  }
  .custom-table th, .custom-table td {
    border: 1px solid #ddd;
    padding: 8px;
  }
  .custom-table th {
    background-color: #f2f2f2;
    text-align: left;
  }
</style>
```

On obtient la sortie HTML suivante :

Introduction

Ce document est un exemple de fichier R Markdown embelli avec du CSS et HTML. Il contient des graphiques dynamiques et des tableaux dynamiques.

Graphique Dynamique

Code et sortie

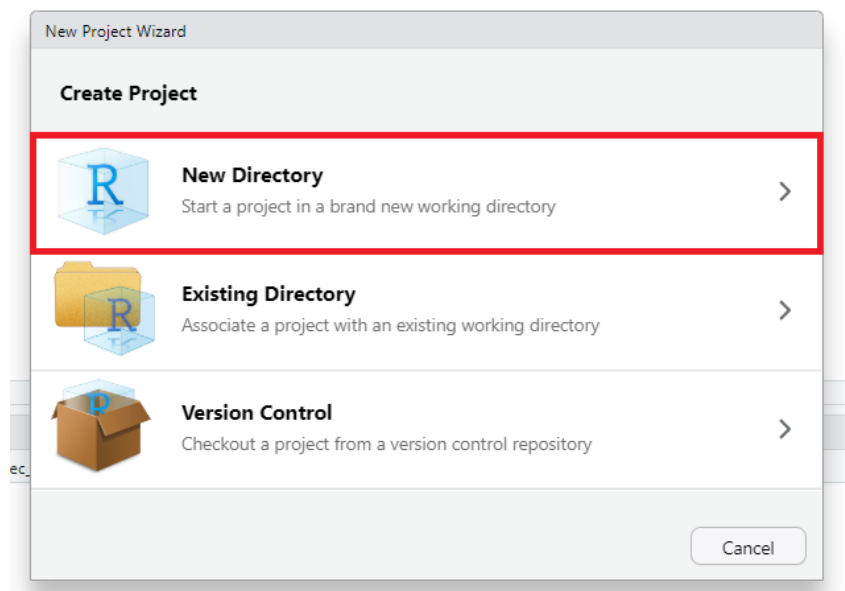
NB : Les graphiques et tableaux dynamiques ne sont pas supportés sous le format pdf. Si vous effectuez une telle sortie R vous renverra des erreurs. Pour corriger ceux-ci, vous devez ajouter l'option `always_allow_html: true` dans l'entête YAML.

VII Creation de livre avec bookdown de R markdown

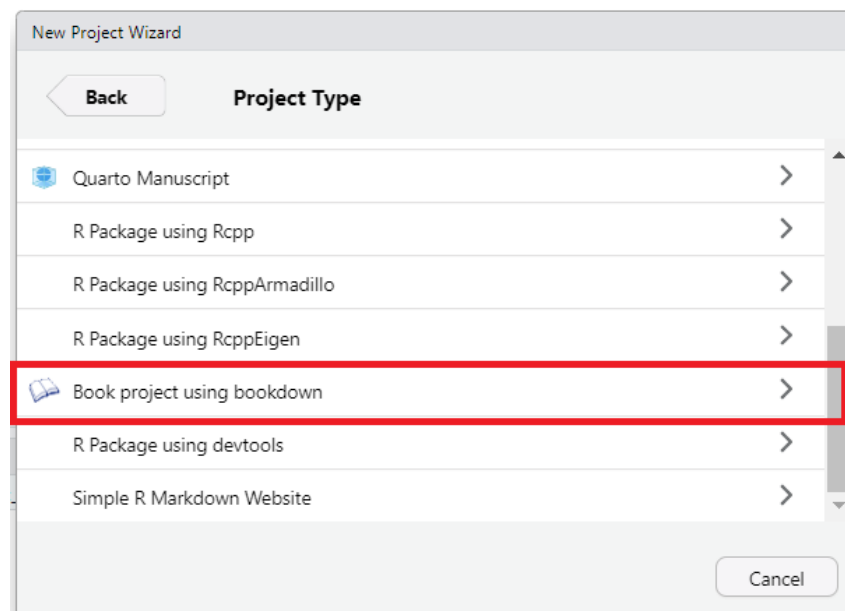
Pour la création de livre avec R markdown, rassurez-vous tout d'abord d'installer le package bookdown avec par exemple la commande `install.packages("bookdown")`.

Ensuite suivez les étapes suivantes :

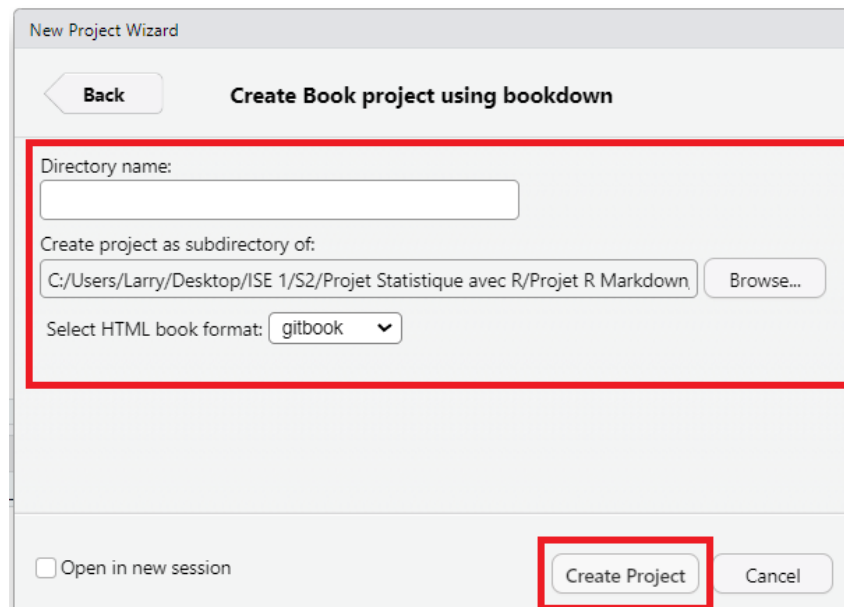
1. Allez dans File -> New Project.
2. Sélectionnez New Directory.



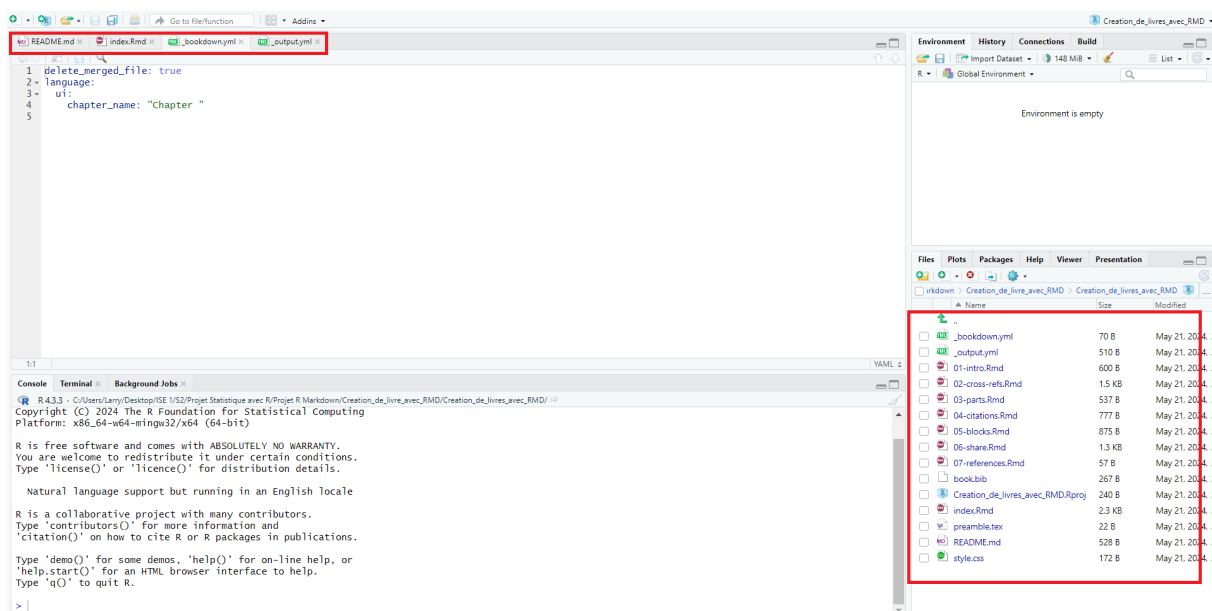
3. Choisissez Book Project using bookdown.



4. Donnez un nom à votre projet et sélectionnez un dossier pour le sauvegarder et Cliquez sur Create Project.



Une structure de répertoire sera générée par défaut pour votre livre.



La structure de base générée inclut les fichiers suivants :

- `_bookdown.yml` : Fichier de configuration pour Bookdown.
- `index.Rmd` : Page de titre et de préface de votre livre.
- `01-intro.Rmd`, `02-chap.Rmd`, etc. : Chapitres de votre livre.
- `book.bib` : Fichier de références bibliographiques (si nécessaire).
- `output.yml` : Configuration des formats de sortie (PDF, HTML, ePub).

Vous pouvez ensuite configurer le `_bookdown.yml` de façon à organiser votre livre comme vous le souhaitez.

retour à introduction

Compilation du livre

Pour compiler le livre, vous pouvez utiliser :

- `bookdown::render_book("index.Rmd", "bookdown::pdf_book")` pour compiler en **pdf**

- `bookdown::render_book("index.Rmd", "bookdown::epub_book")` pour compiler en **epub**



larrysandjo337@gmail.com

@Copyright 2024

SUPERVISE PAR :

Aboubacar HEMA
Data Analysis