

# Bulletin de la Dialyse à Domicile

## Initiation au Logiciel de statistiques R : Initiation à Rmarkdown

Claire Della Vedova

1087 chemin de Sainte Roustagne, 04100 MANOSQUE, France

### Résumé

NDLR : Le RDPLF a pour but principal d'être une aide pour permettre aux équipes de dialyse à domicile d'évaluer leurs pratiques cliniques et également conduire des études à partir d'exports anonymisés des données qu'elles saisissent. A cette fin, depuis juin 2019, un article de formation à l'utilisation du logiciel Libre R est publié trimestriellement à chaque parution du Bulletin de la Dialyse à Domicile. Le but est de permettre à toutes les équipes de réaliser des statistiques de bases et visualiser rapidement leur données.

Le premier article de cette série d'initiation était consacré au téléchargement et à l'installation du logiciel R sur les ordinateurs Macintosh et PC : <https://doi.org/10.25796/bdd.v2i2.20513>.

Le second article était consacré à la visualisation graphiques des données statistiques avec le package Esquisse, simple d'utilisation et nécessitant peu d'apprentissage : <https://doi.org/10.25796/bdd.v2i3.21313>.

Le troisième article était consacré à la visualisation

La formation totale se fait sur 15 mois, au rythme d'un article par trimestre à chaque parution du BDD. Cela laissera largement le temps d'assimiler et tester les connaissances acquises entre chaque article. Pour ceux qui souhaiteront aller plus vite, ils pourront aller sur le blog (<https://statistique-et-logiciel-r.com/>).

Dates des prochaines parutions :

- article 5 (Juin 2020) : la manipulation de données (avec dplyr, notamment les fonction `group_by` et `summarise`)
- article 6 (Septembre 2020) : la réalisation d'analyses descriptives (paramètres statistiques et graphs) sous forme de dashboard avec le package flexboard

Mots clés : biostatistique, épidémiologie, logiciel R, RDPLF

graphiques avec le Package ggplot2 :

<https://doi.org/10.25796/bdd.v2i4.52303>

Ce quatrième article explique comment générer un rapport d'analyse automatiquement, et de façon dynamique, en utilisant "Rmarkdown". Il est expliqué ce qu'est Rmarkdown, son fonctionnement, et surtout, comment faire un rapport d'analyses statistiques automatisé, en pas à pas.

Comme dans les numéros précédent un fichier exemple, tiré de la base de données du RDPLF sera utilisé.

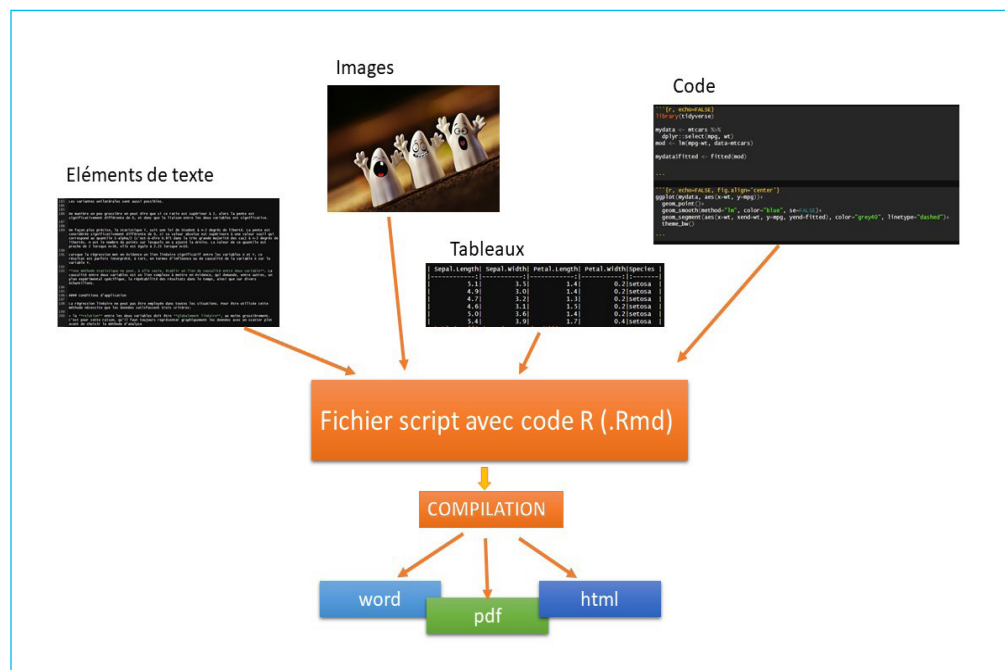
Claire Della Vedova est Ingénieure en biostatistique / data analyste, Elle utilise quotidiennement le logiciel R pour analyser des données. Elle a travaillé pendant plus de 15 ans dans les domaines de l'environnement et de la santé, et a formé de nombreux étudiants et chercheurs à l'utilisation de R. Elle anime depuis novembre 2017 le blog Statistique et Logiciel R dont le but est d'aider les débutants à mieux appréhender les méthodes statistiques classiques et à utiliser le logiciel R plus efficacement, notamment au travers de tutoriels : <https://statistique-et-logiciel-r.com/>.

Faire des analyses statistiques c'est bien, mais pouvoir générer un rapport d'analyse automatiquement, et de façon dynamique, c'est encore mieux ! Et la bonne nouvelle c'est que cela est très facile sous R, en utilisant "Rmarkdown". Dans cet article, je vous explique qu'est ce que c'est Rmarkdown, comment ça fonctionne, et surtout, comment faire votre premier rapport d'analyses statistiques automatisé, en pas à pas.

## 1. C'est quoi RMarkdown?

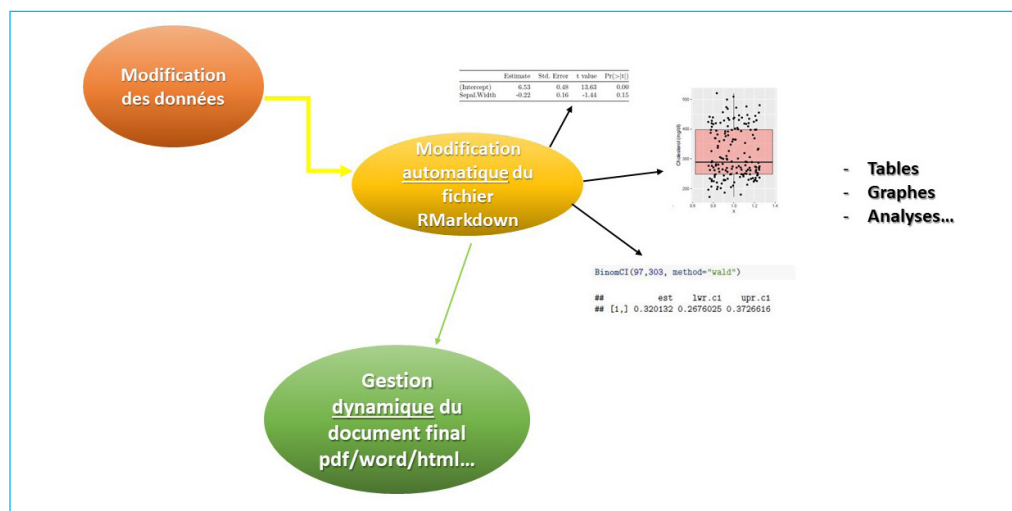
RMarkdown est un format de script R, un peu différent du format classique (en .R), qui utilise des balises, un peu comme le format html, mais de façon simplifiée. Ce format Rmarkdown est noté .Rmd.

Le format Rmarkdown a deux grands avantages. Tout d'abord, il est possible d'inclure plus d'éléments que du code R. Nous pouvons insérer du texte, du code R, des images, des liens hypertexte, etc.... Ensuite, le format Rmarkdown permet de générer un rendu, c'est-à-dire un document en html, word ou pdf, qui contiendra lui aussi le texte, les images, les liens hypertextes, le code R et le résultat de ce code R.



J'utilise généralement un rendu au format Word pour le travail en collaboration, car tout le monde peut, sans connaissance particulière, ajouter un commentaire, une image ou faire une modification. J'emploie le format HTML pour la communication, lorsque que je veux présenter une analyse ou un résultat, et le format pdf pour rendre, par exemple, un rapport d'analyse, dans sa version finale.

Un des grands intérêts de RMarkdown réside aussi dans le caractère dynamique des document générés. Cela signifie que si un jeu de données est modifié, cette modification sera prise en compte dans le reste du script et donc de l'analyse.



Ce format a, en outre, l'avantage de pouvoir rendre accessible toutes les informations, les données et les calculs qui ont été nécessaires à une analyse de données. Autrement dit d'inscrire son travail dans une démarche reproductible. Cela permet, par exemple, de conserver une trace du cheminement de sa pensée, ou des hypothèses testées, et des résultats obtenus. Ou encore de toujours garder le lien entre les données utilisées et les figures ou les tableaux descriptifs réalisés (pour une publication par exemple).



J'espère vous avoir convaincu de son utilité !

## 2. Avant de commencer...

### 2.1 Installez et chargez les packages nécessaires

Avant de démarrer, il est préférable d'avoir installé des versions récentes de R et R Studio, voir les dernières versions (version 3.6.3 au 29/02/2020, pour R et version 1.2.5033 au 08/01/2020 pour RStudio).

Les packages nécessaires sont :

- rmarkdown,
- markdown
- knitr
- tinytex (nécessaire à la génération de pdf)

Vous pouvez les installer avec l'outil d'installation (fenêtre en bas à gauche dans R studio), ou en recopiant les lignes de commande suivantes dans la console :

```
install.packages(<<knitr>>)
install.packages(<<rmarkdown>>)
```

```
install.packages(<<markdown>>)
```

```
install.packages(<<tinytex>>)
```

```
library(knitr)
```

```
library(rmarkdown)
```

```
library(markdown)
```

```
library(tinytex)
```

Pour utiliser le package tinytex, il est également nécessaire d'employer la commande suivante (toujours dans la console) :

```
tinytex::install_tinytex()
```

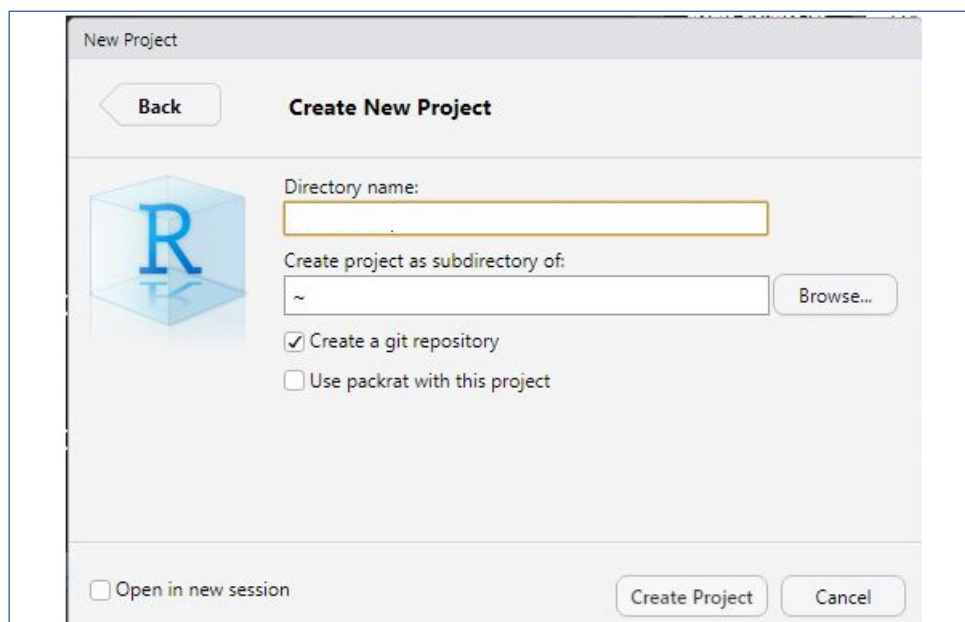
## 2.2 Organisez votre travail sous la forme d'un projet R :

Lorsque vous réalisez un projet d'analyse de données sous R, je vous recommande de le rendre indépendant de vos autres projets, en l'associant à un projet R spécifique. Pour cela, créez un dossier sous windows à l'emplacement souhaité, puis dans R Studio : File → New Project → Existing Directory → Indiquez l'emplacement → Create Project.

Vous pouvez retrouver la démarche en pas à pas dans mon premier article (<https://www.bdd.rdpf.org/index.php/bdd/article/view/20513/19163>), Après avoir suivi ces étapes, vous avez donc un dossier de travail et un nouveau projet R qui lui est associé.

Remarque : Vous pouvez aussi créer directement ce dossier à partir de R Studio, ce qui permettra de l'associer à un projet R automatiquement :

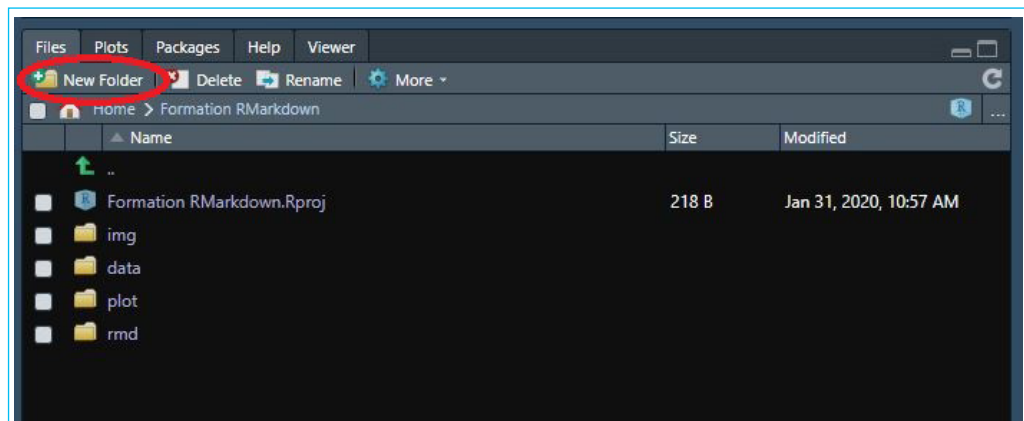
File → New Project → New Directory → New Project



Je vous conseille encore, avant de continuer, de créer quatre sous-répertoires :

- img, qui contiendra les images
- data, qui contiendra vos fichiers csv
- plot, dans lequel vous pourrez sauvegarder des graphiques
- rmd, qui contiendra vos scripts en Rmarkdown

Ceci se fait à partir de la fenêtre en bas à droite de RStudio:

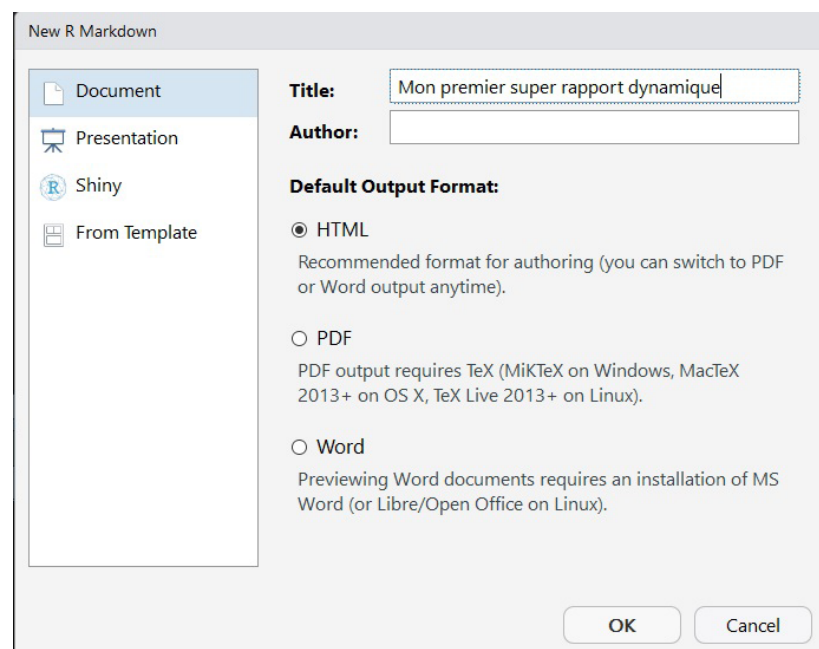


### 3. Générez votre premier rapport dynamique

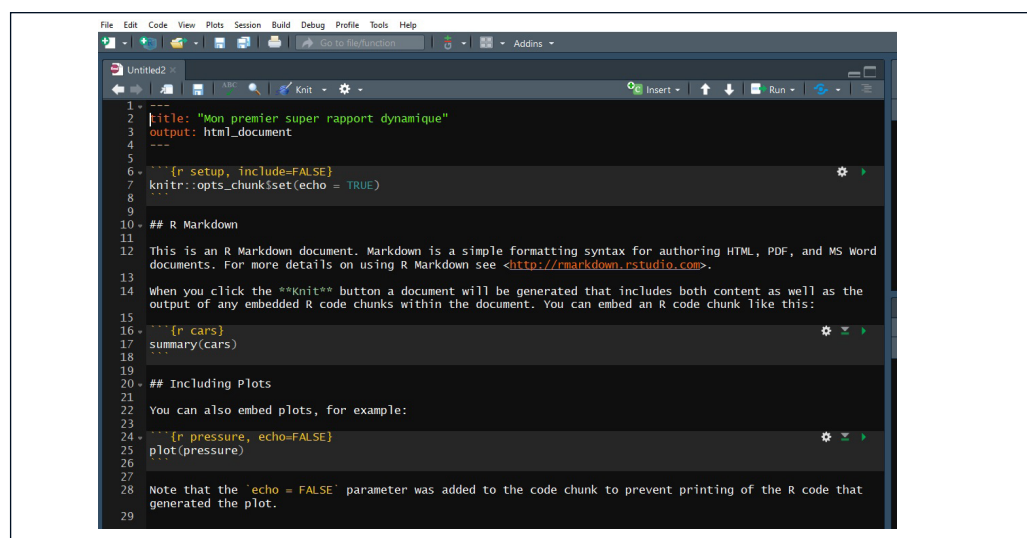
#### 3.1 Ouvrez un nouveau fichier R markdown et enregistrez le :

File → New File → Rmarkdown

Laissez toutes les options par défaut (document et html), et remplissez le champs “Title”, ce titre correspondra au titre du document, il peut être long.



Après avoir appuyé sur “OK”, un fichier Rmd apparaît dans la fenêtre d’édition (en haut à gauche). Ce fichier est un exemple, il comporte du code et du texte, ainsi que le titre que vous lui avez donné.

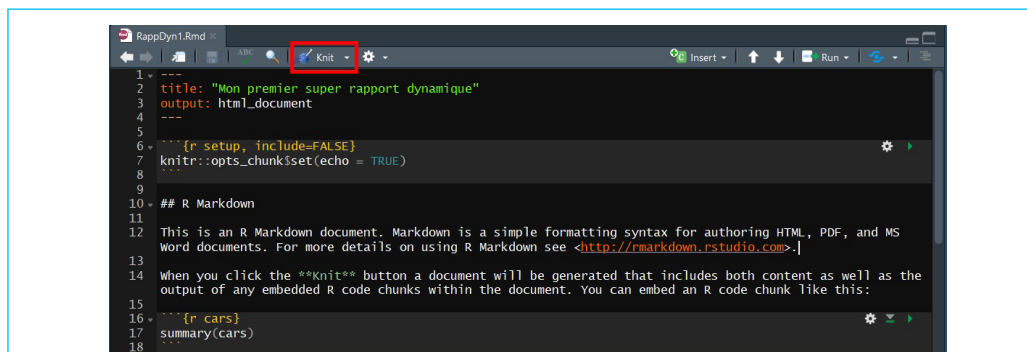


Enregistrez le dans le dossier rmd créé, en lui donnant un nom (un nom de fichier cette fois, “RappDyn1” par exemple) :

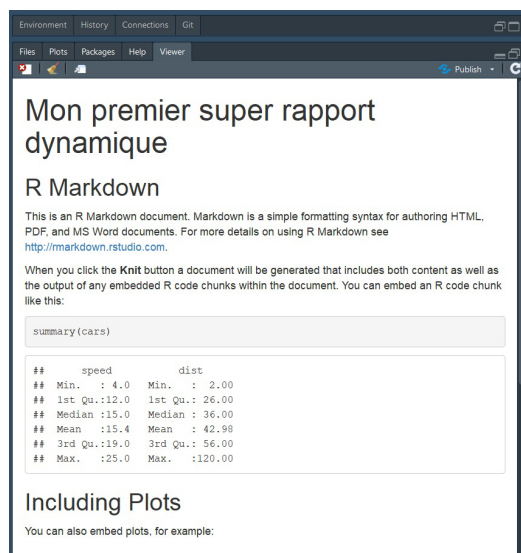
File → Save ou CTRL + S

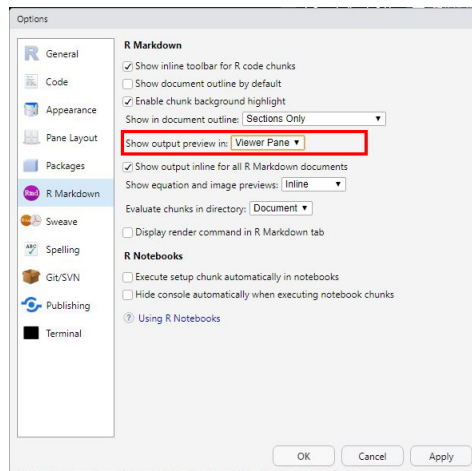
### 3.2 Lancer la génération

Pour cela, c’est très simple, cliquez sur le bouton avec la pelote de laine et les aiguilles :



Un fichier html portant le même nom que le fichier.Rmd est alors créé dans le working directory (là où vous avez créé votre projet R), et en même temps le rapport généré s’ouvre dans le viewer de R Studio :

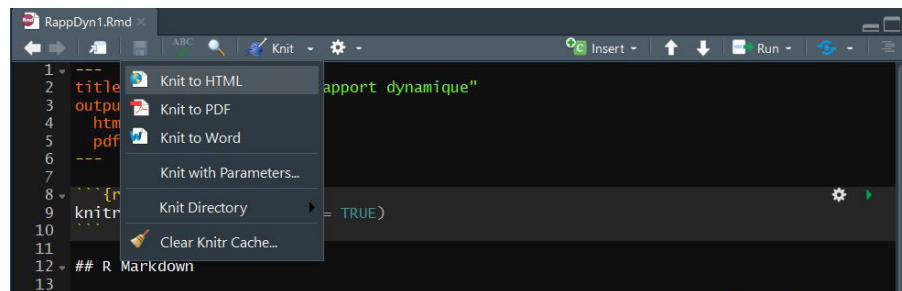




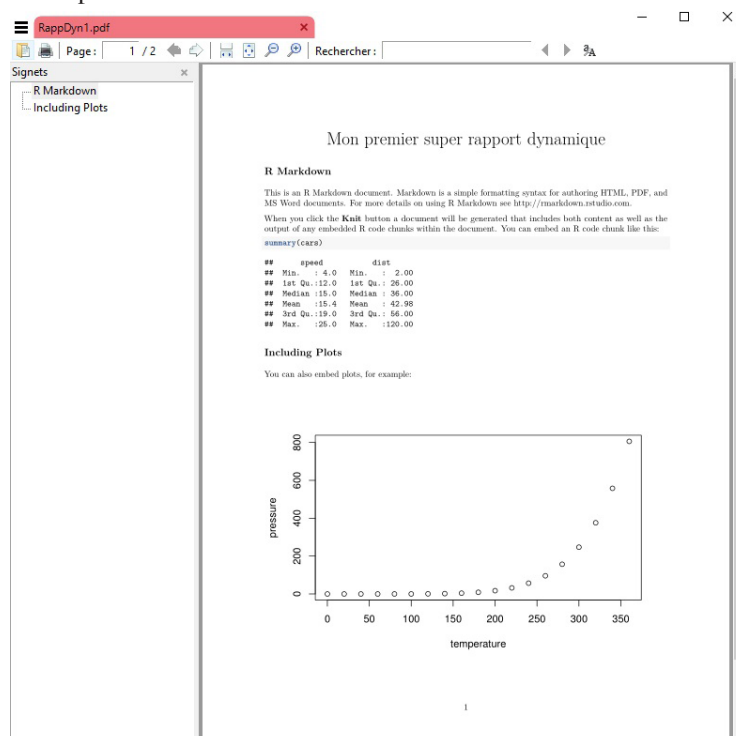
Si le document s'ouvre dans une fenêtre, allez dans Tools → Global Options → Rmarkdown

### 3.3 Choisissez votre format de sortie : html, word, pdf

En cliquant sur la flèche vers le bas, du bouton knit vous pourrez changer le format de sortie du rapport dynamique.



Le format par défaut est celui choisi à l'étape précédente, c'est à dire html, ici. Vous pourrez choisir Word ou pdf.



Maintenant que vous avez vu le rendu, nous allons regarder de plus près le fichier Rmd.

## 4. Organisation du fichier Rm

### 4.1 L'en-tête

L'en-tête est contenu entre deux séries de trois pointillés. Par défaut, il contient deux types d'éléments : le titre du document, et son format de sortie. Il est possible d'ajouter d'autres élé-

```
1 ---
2 title: "Mon premier super rapport dynamique"
3 output:
4   html_document: default
5   pdf_document: default
6 ---
```

ments, comme l'auteur, et la date de création, par exemple.

Il est encore possible d'ajouter une table des matières, ou encore un lien vers un fichier de références bibliographiques. Pour plus d'informations, vous pouvez consulter l'article "Com-

```
1 ---
2 title: "Mon premier super rapport dynamique"
3 output:
4   html_document: default
5   pdf_document: default
6   author : Claire Della Vedova (blog_SLR)
7   date : "08/08/18"
8 ---
```

ment insérer des références bibliographiques dans un document Rmarkdown ?" (<https://wp.me/p93iR1-fY>)

### 4.2 Les chunks contenant le code R

Les parties de code R sont contenues dans des blocs, appelés chunks». Ces chunks commencent par la balise `{r }` et finissent par la balise `}`.

```
18 {r cars}
19 summary(cars)
20 }
```

C'est dans ces accolades que nous allons pouvoir ajouter des options pour choisir de faire apparaître, ou non, le code dans le rapport dynamique, ainsi que les résultats, ou encore pour définir la taille des plots. Ces chunks peuvent être nommés à l'intérieur de l'accolade. Dans l'exemple précédent, le nom est «cars».

### 4.3 Le chunk de set up

```
7
8 {r setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE)
10
11 }
```

Ce chunk se trouve en dessous de l'en-tête, il permet de régler les options par défaut de tous les chunks. Par exemple, nous allons pouvoir indiquer que nous ne souhaitons pas faire apparaître les messages et les warnings qui pourraient être générés lors de l'exécution des chunk. Au lieu de le faire pour tous les chunks, nous pouvons le faire une seule fois ici.

```
8 {r setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE, message=FALSE, warning=FALSE)
10 }
```



Et si, pour un chunk donné, nous voulons faire apparaître les warnings et les messages, nous utiliserons le message=TRUE et warnings=TRUE dans l'accolade du chunk concerné.

#### 4.4 Les parties texte.

Elles peuvent être insérées partout en dehors des chunks

```
## R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:
```

Il est possible de mettre en gras, ou en italique, certaines parties du texte.

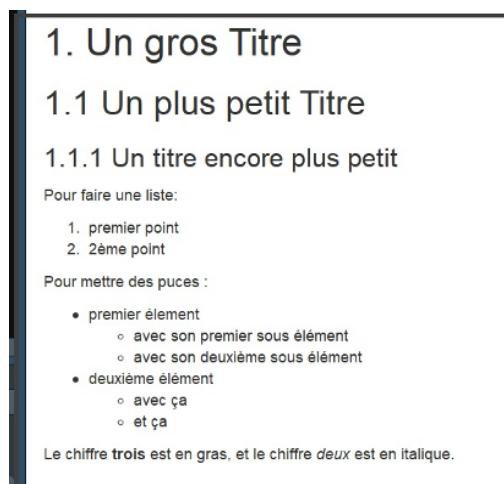
### 5. Quelques éléments pour débiter

#### 5.1 Titre, liste, gras, italique

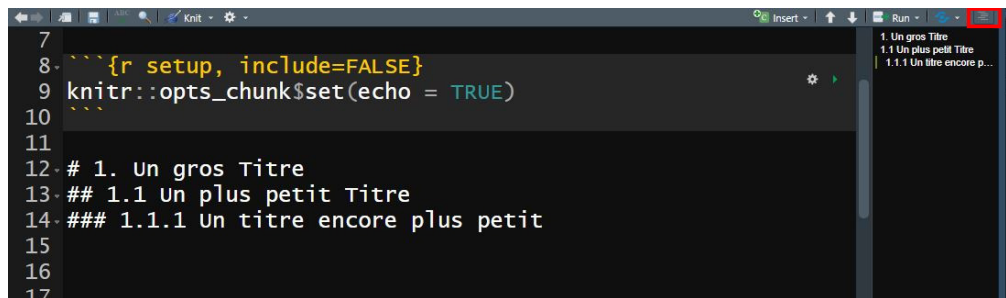
Voici comment faire des titres, des listes, des puces, mettre en gras et en italique:

```
52 # 1. Un gros titre
53 ## 1.1 Un plus petit titre
54 # 1.1.1 Un titre encore plus petit
55
56 Pour faire une liste:
57
58 1. premier point
59 2. 2ème point
60
61 Pour mettre des puces:
62 |
63 * premier élément
64 + avec son premier sous élément
65 + avec son deuxième sous élément
66 * deuxième élément
67 + avec ça
68 + et ça
69
70 Le chiffre **trois** est un gras, et le chiffre *deux* est un italique
71
```

Le rendu :



La table des matières du document peut être visualisée en cliquant sur le bouton suivant :



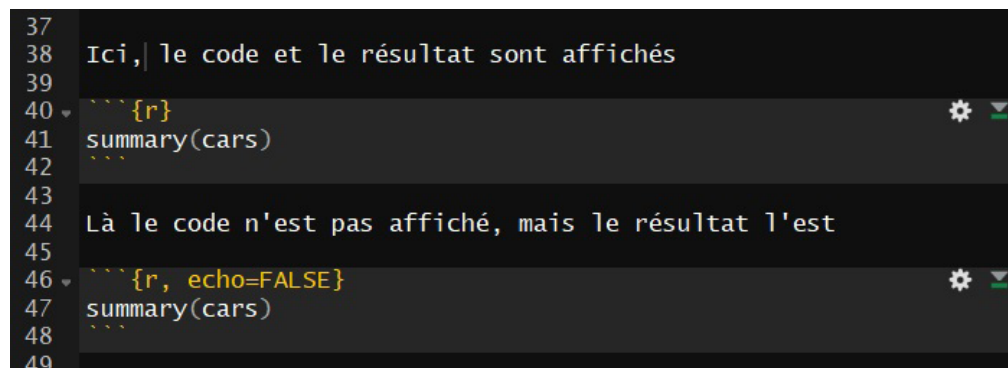
```
7
8 {r setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE)
10
11
12 # 1. Un gros Titre
13 ## 1.1 Un plus petit Titre
14 ### 1.1.1 Un titre encore plus petit
15
16
17
```

1 Un gros Titre  
1.1 Un plus petit Titre  
1.1.1 Un titre encore plus petit

## 5.2 Gestion du code

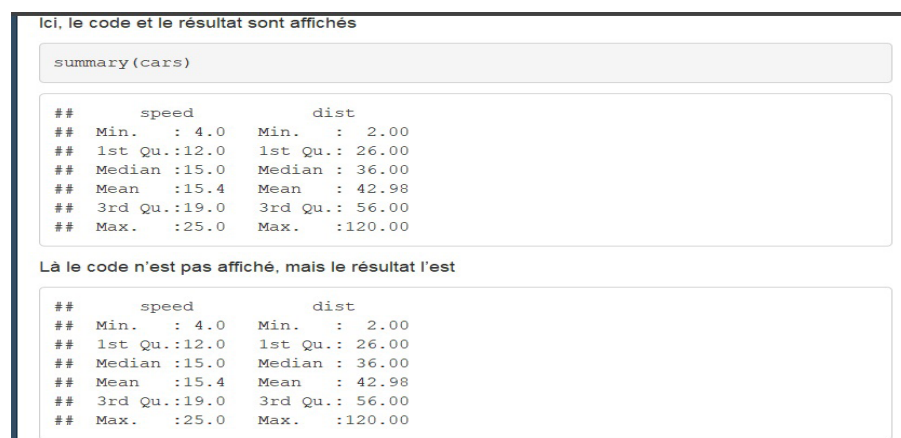
### 5.2.1 Affichage du code

Par défaut, le code R est affiché lors de la génération du rapport, cela correspond à l'option `echo=TRUE`. Pour supprimer l'affichage du code, il faut utiliser l'argument `echo=FALSE`.



```
37
38 Ici, le code et le résultat sont affichés
39
40 {r}
41 summary(cars)
42
43
44 Là le code n'est pas affiché, mais le résultat l'est
45
46 {r, echo=FALSE}
47 summary(cars)
48
49
```

Le rendu :



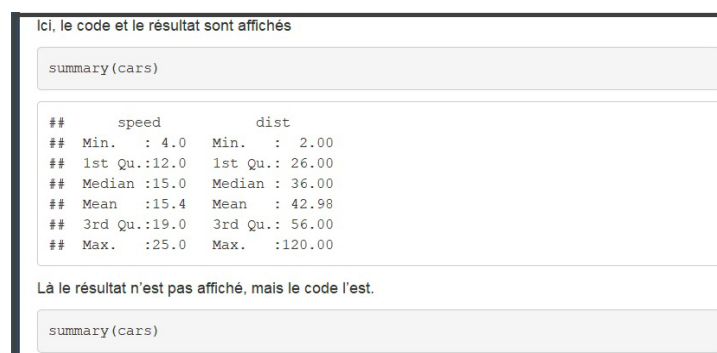
Ici, le code et le résultat sont affichés	
<pre>summary(cars)</pre>	
<pre>##      speed      dist ## Min.   : 4.0    Min.   : 2.00 ## 1st Qu.:12.0    1st Qu.: 26.00 ## Median :15.0    Median : 36.00 ## Mean   :15.4    Mean   : 42.98 ## 3rd Qu.:19.0    3rd Qu.: 56.00 ## Max.   :25.0    Max.   :120.00</pre>	
Là le code n'est pas affiché, mais le résultat l'est	
<pre>##      speed      dist ## Min.   : 4.0    Min.   : 2.00 ## 1st Qu.:12.0    1st Qu.: 26.00 ## Median :15.0    Median : 36.00 ## Mean   :15.4    Mean   : 42.98 ## 3rd Qu.:19.0    3rd Qu.: 56.00 ## Max.   :25.0    Max.   :120.00</pre>	

### 5.2.2 Affichage du résultat du code

Par défaut, le résultat du code est affiché lors de la génération du rapport, cela correspond à l'option `results=»markup»`. Pour le supprimer, il faut utiliser l'argument `results=»hide»`.

```
38 Ici, le code et le résultat sont affichés
39
40 {r}
41 summary(cars)
42
43
44 Là le résultat n'est pas affiché, mais le code l'est.
45
46 {r, results="hide"}
47 summary(cars)
48
49
```

Le rendu :



### 5.2.3 Gérer la taille des figures

A titre d'exemple, nous allons, ici, utiliser les mêmes données que dans le premier article, elles sont téléchargeables au format csv, à cette adresse : <https://www.rdpf.org/exempleR/FichierExempleStat.csv>

Une fois téléchargé, placez le fichier csv dans le dossier "data" de votre projet R , puis utilisez la commande suivante pour les importer dans R :

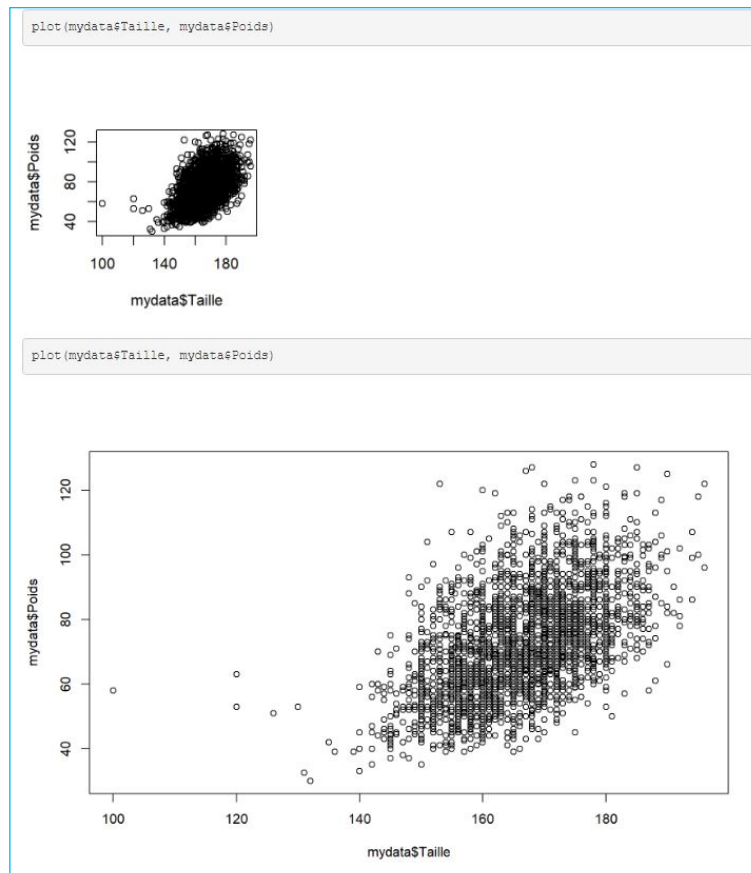
```
mydata <- read.csv2(<<../data/FichierExempleStat.csv>>)
```

Pour contrôler la taille des images, nous utilisons les options fig.width et fig.height :

```
{r, fig.width=3, fig.height=3}
plot(mydata$Taille, mydata$Poids)
...

{r, fig.width=9, fig.height=6}
plot(mydata$Taille, mydata$Poids)
...
```

Le rendu



### 5.3 Insérer une image

Pour insérer une image, il suffit de placer l'image dans le dossier "img" (par exemple une image nommée "logo\_journal.jpg") et d'utiliser, en dehors d'un chunk, la commande suivante :

```
337  
338   
339
```

Le rendu :



## 5.4 Créer une table de données

Pour créer une table, nous pouvons utiliser, dans un chunk, la fonction `kable()` du package `knitr`. Ici, le code et la sortie sans la fonction `kable()` :

```
349  
350- ````{r, eval=FALSE, echo=FALSE}  
351 mydata <- read.csv2("../data/FichierExempleStat.csv")  
352 head(mydata)  
353  
354 ````  
355
```

```
## code.post PAYS sexe Age.1ere.DP Charlson Charlson_modif Type.de.DP  
## 1 56100 FRANCE F 20.13 3 3 DPCA  
## 2 1006 TUNISIE F 50.74 3 2 DPCA  
## 3 2540 LUXEMBOURG M 18.86 5 5 DPA quotidienne  
## 4 3200 FRANCE F 50.86 6 5 DPA quotidienne  
## 5 31603 FRANCE F 54.07 6 5 DPCA  
## 6 74374 FRANCE F 53.89 4 3 DPCA  
## Taille Poids  
## 1 132 30.0  
## 2 131 32.5  
## 3 140 33.0  
## 4 142 35.0  
## 5 150 35.0  
## 6 145 37.0
```

Ici le code avec la sortie obtenue avec `kable`:

```
368  
369- ````{r}  
370 mydata <- read.csv2("../data/FichierExempleStat.csv")  
371 library(knitr)  
372 kable(head(mydata))  
373  
374
```

code.post	PAYS	sexe	Age.1ere.DP	Charlson	Charlson_modif	Type.de.DP	Taille	Poids
56100	FRANCE	F	20.13	3	3	DPCA	132	30.0
1006	TUNISIE	F	50.74	3	2	DPCA	131	32.5
2540	LUXEMBOURG	M	18.86	5	5	DPA quotidienne	140	33.0
3200	FRANCE	F	50.86	6	5	DPA quotidienne	142	35.0
31603	FRANCE	F	54.07	6	5	DPCA	150	35.0
74374	FRANCE	F	53.89	4	3	DPCA	145	37.0

## 5.5 Insérer un lien hypertexte

Pour insérer un lien hypertexte, il suffit de placer l'adresse entre les symboles `<` et `>`, par exemple :

```
388  
389 L'adresse de mon blog est :  
390 `<https://statistique-et-logiciel-r.com>`.
```

Le rendu :

L'adresse de mon blog est : <https://statistique-et-logiciel-r.com>.

## 5.6 Insérer un nouveau chunk

Sous windows, un nouveau chunk peut être insérer en utilisant les touches Ctrl+Alt+I.

## 6. Mais comment ça marche ?

La production des rapports se réalise en deux étapes :

1. Le fichier .Rmd qui contient des blocs de code R (que l'on appelle chunk) et du texte est fournit au package knitr qui va exécuter les codes R afin d'obtenir les sorties, et créer un document au format markdown (.md). Ce document contient alors les codes R, les résultats (ou les sorties), et les textes de commentaires.
2. Ce fichier .md est ensuite converti vers le format souhaité, comme "html", "docx" ou encore "pdf", par le package markdown basé sur pandoc (c'est un outil de conversion de document). (<https://pandoc.org/>).

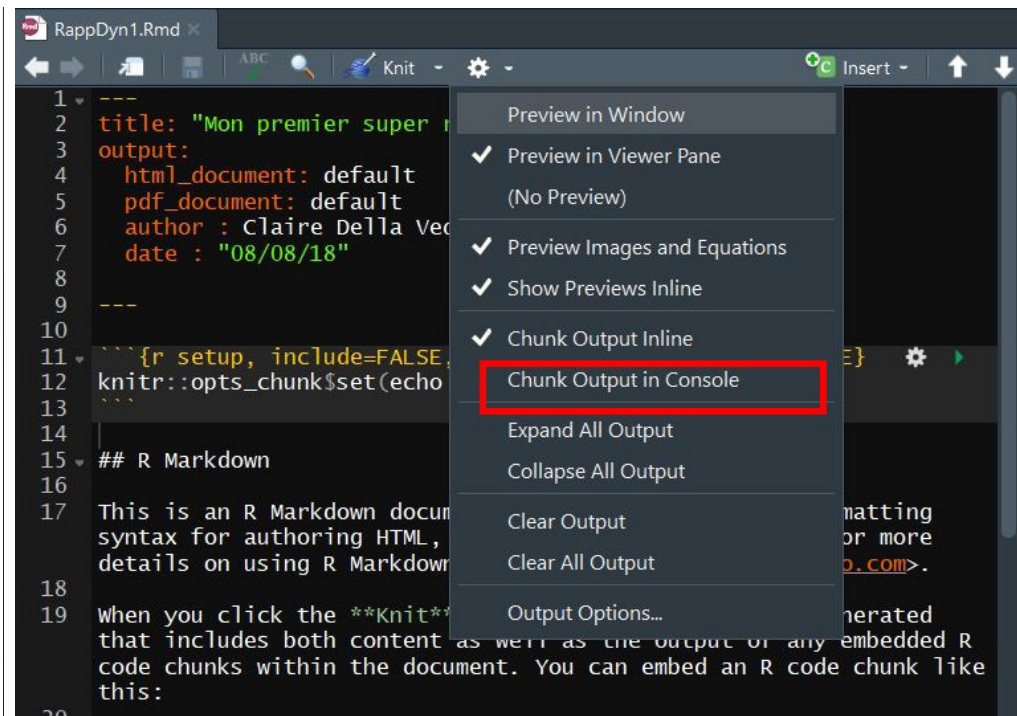


## 7. Remarques

### 7.1. Exécuter les codes des chunks.

Les codes contenus dans les chunks d'un document.Rmd peuvent être exécutés comme les codes contenus dans un script R en.R. Vous pouvez exécuter l'ensemble en faisant CTRL + A pour tout sélectionner et CTRL + ENTER pour exécuter. Vous pouvez aussi exécuter le code chunk par chunk en plaçant votre souris au début d'un chunk puis en utilisant CTRL + ENTER. D'autres options sont disponibles dans le menu déroulant du bouton run ( tout près de celui permettant d'afficher la table des matières)

Lorsque le code est exécuté, par défaut les résultats (sorties), et les plots sont affichés dans le document .Rmd. Pour afficher les sorties dans la console et les plots dans la fenêtre graphique, il faut choisir l'option "Chunk Output in Console"



## 8. Conclusion

Pour ceux qui ne travaillent pas encore en R markdown, j'espère que cet article vous convaincra qu'il est facile de franchir le pas. Cela demande un investissement minime pour un gain maximal. En passant en R markdown vous gagnerez en temps, en productivité, et en qualité. N'hésitez plus !

## 9. Pour aller plus loin :

- 10 astuces pour améliorer vos compte rendus d'analyse statistique <https://statistique-et-logiciel-r.com/10-astuces-pour-ameliorer-vos-documents-en-r-markdown/>
- R Markdown Cheat Sheet file:///C:/Users/clair/Downloads/rmarkdown-2.0.pdf
- R Markdown Reference Guide (<https://www.rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf>)

Vous pouvez les télécharger directement depuis l'onglet Help ==> Cheatsheet

- le livre "R Markdown: The Definitive Guide" : <https://bookdown.org/yihui/rmarkdown/>
- Pimp my RMD : a few tips for R Markdown : <https://holtzy.github.io/Pimp-my-rmd/>
- Tutoriel : personnalisez les rendus Word de R markdown <https://statistique-et-logiciel-r.com/tutoriel-personnalisez-word-rmarkdown/>

- Créer un livre-document avec Bookdown <https://statistique-et-logiciel-r.com/creer-un-livre-document-avec-bookdown/>
- Comment insérer des références bibliographiques dans un document Rmarkdown ? [https://statistique-et-logiciel-r.com/comment-inserer-des-references-bibliographiques-ge-rees-sous-mendeley-dans-un-document-rmarkdown\\_v2/](https://statistique-et-logiciel-r.com/comment-inserer-des-references-bibliographiques-ge-rees-sous-mendeley-dans-un-document-rmarkdown_v2/)
- Comment faire un dashboard avec R ? <https://statistique-et-logiciel-r.com/dashboard-r/>
- Comment faire un CV élégant avec R et le package pagedown? <https://statistique-et-logiciel-r.com/cv-elegant-r-package-pagedown/>
- l'add on remedy, développé par ThinkR, qui s'intègre à R Studio pour simplifier l'écriture des fichiers Rmd : <https://github.com/ThinkR-open/remedy>

*Open Access : cet article est sous licence Creative commons CC BY 4.0 : <https://creativecommons.org/licenses/by/4.0/deed.fr>*

*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.*

*Cette licence est acceptable pour des œuvres culturelles libres.*

*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. <http://creativecommons.org/licenses/by/4.0/>.*