

# Formation R Perfectionnement

Drees – 16-17 avril 2018



Martin CHEVALIER (Insee)

# Générer automatiquement des documents depuis R

## Pourquoi générer automatiquement des documents ?

- ▶ Exporter et documenter des **traitements** en vue d'une réutilisation future : statistiques pour une étude, traitements réalisés lors d'une réunion de travail, etc.

**Remarque** Utilisation analogue à celle permise par les instructions `ODS RTF` ou `ODS PDF` de SAS.

# Générer automatiquement des documents depuis R

## Pourquoi générer automatiquement des documents ?

- ▶ Exporter et documenter des **traitements** en vue d'une réutilisation future : statistiques pour une étude, traitements réalisés lors d'une réunion de travail, etc.

**Remarque** Utilisation analogue à celle permise par les instructions `ODS RTF` ou `ODS PDF` de SAS.

- ▶ Construire des **rapports complets et automatisés** pour des tâches répétitives : rapports d'utilisation, tests de la cohérence ou de la qualité de nouvelles données, etc.

# Générer automatiquement des documents depuis R

## Pourquoi générer automatiquement des documents ?

- ▶ Exporter et documenter des **traitements** en vue d'une réutilisation future : statistiques pour une étude, traitements réalisés lors d'une réunion de travail, etc.

**Remarque** Utilisation analogue à celle permise par les instructions `ODS RTF` ou `ODS PDF` de SAS.

- ▶ Construire des **rapports complets et automatisés** pour des tâches répétitives : rapports d'utilisation, tests de la cohérence ou de la qualité de nouvelles données, etc.
- ▶ Produire des publications **reproductibles** sur différents supports : notes, documentation, articles de revues, etc.

# Générer automatiquement des documents depuis R

## Principe de la génération automatique de documents

La génération automatique de documents complets repose sur deux éléments :

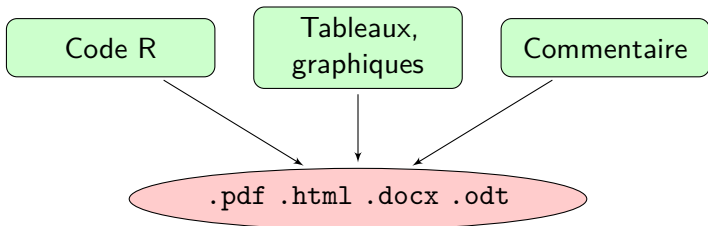
1. Articuler le code, les résultats et le commentaire dans un **même document** : garantir la cohérence et faciliter les mises à jour ;
2. Formater de façon standardisée le document vers **plusieurs sorties** : .html, .pdf, .docx, .odt.

# Générer automatiquement des documents depuis R

## Principe de la génération automatique de documents

La génération automatique de documents complets repose sur deux éléments :

1. Articuler le code, les résultats et le commentaire dans un **même document** : garantir la cohérence et faciliter les mises à jour ;
2. Formater de façon standardisée le document vers **plusieurs sorties** : .html, .pdf, .docx, .odt.



# Générer automatiquement des documents depuis R

## Étapes de la génération automatique de documents

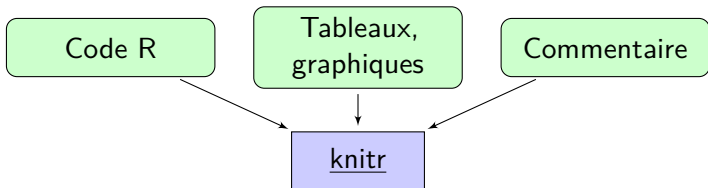
Code R

Tableaux,  
graphiques

Commentaire

# Générer automatiquement des documents depuis R

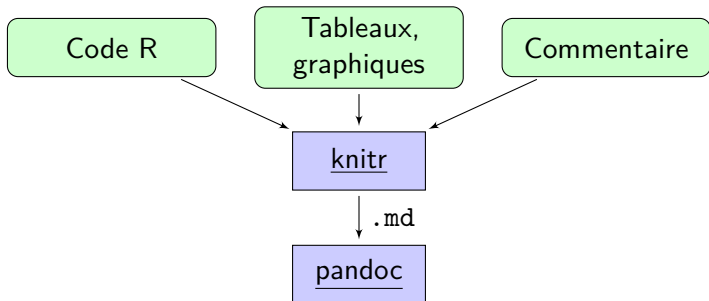
## Etapes de la génération automatique de documents





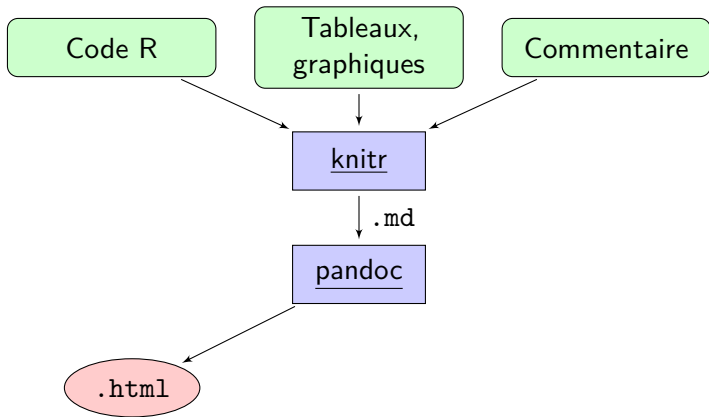
# Générer automatiquement des documents depuis R

## Etapes de la génération automatique de documents



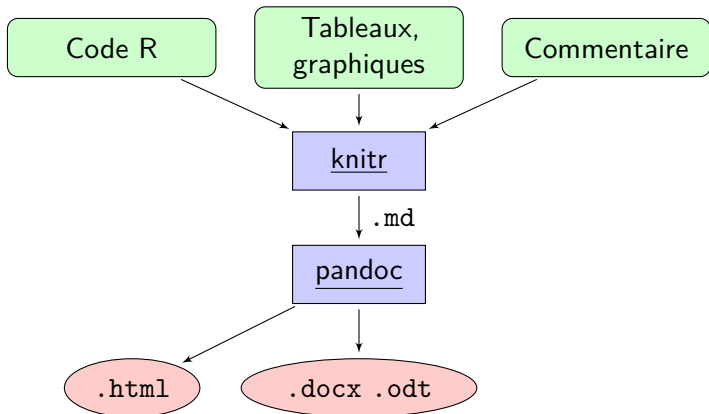
# Générer automatiquement des documents depuis R

## Etapes de la génération automatique de documents



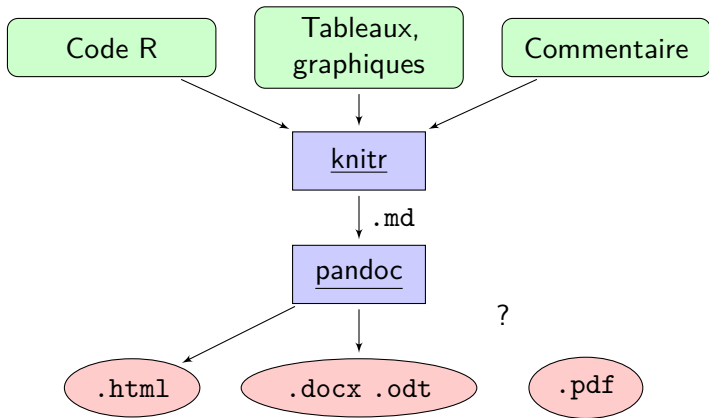
# Générer automatiquement des documents depuis R

## Etapes de la génération automatique de documents



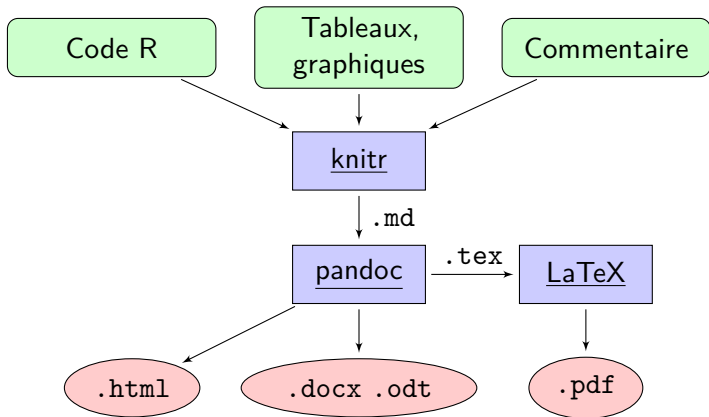
# Générer automatiquement des documents depuis R

## Etapes de la génération automatique de documents



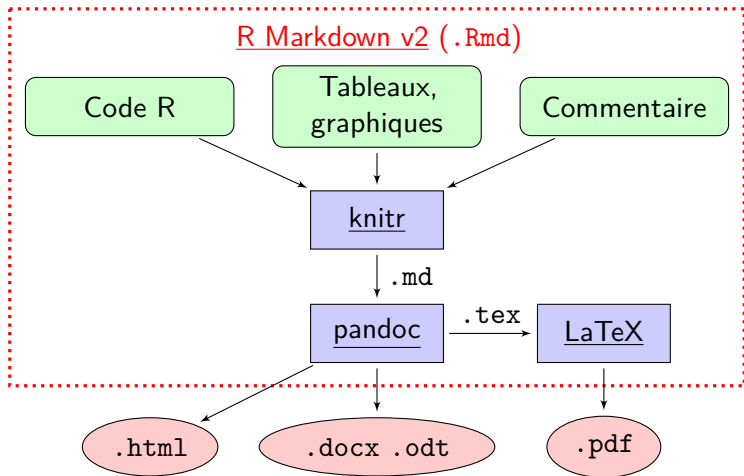
# Générer automatiquement des documents depuis R

## Etapes de la génération automatique de documents



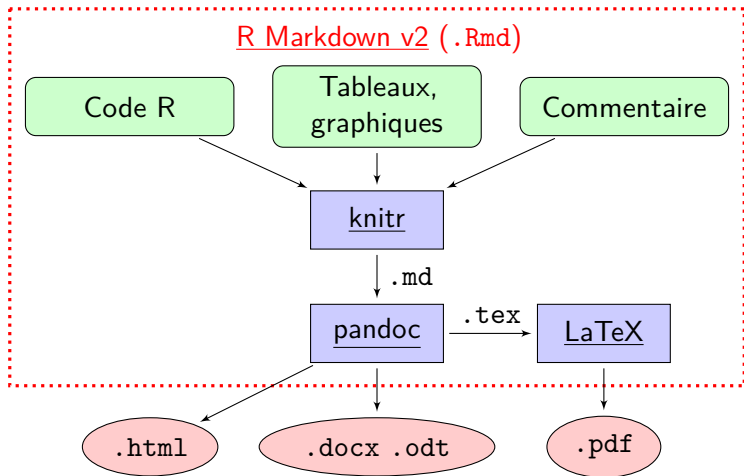
# Générer automatiquement des documents depuis R

## Etapes de la génération automatique de documents



# Générer automatiquement des documents depuis R

## Etapes de la génération automatique de documents



**Note** rmarkdown et knitr sont des *packages* R (avec plusieurs dépendances); pandoc et LaTeX sont des programmes autonomes.

# Générer automatiquement des documents depuis R

## Préparer et tester l'environnement de travail

### 1. Travailler sous RStudio

- ▶ RStudio facilite l'édition et la compilation de fichier `.Rmd` ;
- ▶ pandoc est embarqué par défaut dans RStudio.



# Générer automatiquement des documents depuis R

## Préparer et tester l'environnement de travail

### 1. Travailler sous RStudio

- ▶ RStudio facilite l'édition et la compilation de fichier `.Rmd` ;
- ▶ pandoc est embarqué par défaut dans RStudio.

### 2. Installer les *packages* nécessaires

- ▶ installer le *package* `rmarkdown` et ses dépendances ;
- ▶ installer le *package* `knitr` et ses dépendances.

# Générer automatiquement des documents depuis R

## Préparer et tester l'environnement de travail

### 1. Travailler sous RStudio

- ▶ RStudio facilite l'édition et la compilation de fichier `.Rmd` ;
- ▶ pandoc est embarqué par défaut dans RStudio.

### 2. Installer les *packages* nécessaires

- ▶ installer le *package* `rmarkdown` et ses dépendances ;
- ▶ installer le *package* `knitr` et ses dépendances.

### 3. Pour produire des fichiers `.pdf`, installer LaTeX (MiKTeX sous Windows) et s'assurer que ses programmes figurent dans le *path* de Windows.

# Générer automatiquement des documents depuis R

## Préparer et tester l'environnement de travail

### 1. Travailler sous RStudio

- ▶ RStudio facilite l'édition et la compilation de fichier `.Rmd` ;
- ▶ pandoc est embarqué par défaut dans RStudio.

### 2. Installer les *packages* nécessaires

- ▶ installer le *package* `rmarkdown` et ses dépendances ;
- ▶ installer le *package* `knitr` et ses dépendances.

### 3. Pour produire des fichiers `.pdf`, installer LaTeX (MiKTeX sous Windows) et s'assurer que ses programmes figurent dans le *path* de Windows.

### 4. Créer un nouveau fichier R Markdown (`.Rmd`), installer les *packages* complémentaires demandés, choisir le type de document et compiler le fichier d'exemple (`Ctrl + K`).

## Générer automatiquement des documents depuis R

### Ecrire du texte dans R Markdown

Pour écrire du texte dans un document R Markdown, il suffit de le **taper dans le fichier .Rmd** (sans le commenter ni l'échapper d'aucune manière).

# Générer automatiquement des documents depuis R

## Ecrire du texte dans R Markdown

Pour écrire du texte dans un document R Markdown, il suffit de le **taper dans le fichier** `.Rmd` (sans le commenter ni l'échapper d'aucune manière).

Des **balises** spéciales permettent de mettre en forme le document :

- ▶ les signes `*` et `_` permettent de mettre des mots en *\*italique\** ou en **\*\*gras\*\*** ;
- ▶ les six niveaux de titres sont préfixés par les signes `#` (premier niveau), `##` (deuxième niveau), etc.
- ▶ des listes sont automatiquement créées à partir de successions de `-` ou de séquences de nombres ou de lettres séparées par un retour à la ligne.

# Générer automatiquement des documents depuis R

## Ecrire du texte dans R Markdown

Pour écrire du texte dans un document R Markdown, il suffit de le **taper dans le fichier** `.Rmd` (sans le commenter ni l'échapper d'aucune manière).

Des **balises** spéciales permettent de mettre en forme le document :

- ▶ les signes `*` et `_` permettent de mettre des mots en *\*italique\** ou en **\*\*gras\*\*** ;
- ▶ les six niveaux de titres sont préfixés par les signes `#` (premier niveau), `##` (deuxième niveau), etc.
- ▶ des listes sont automatiquement créées à partir de successions de `-` ou de séquences de nombres ou de lettres séparées par un retour à la ligne.

**Note** Pour une présentation synthétique de R Markdown, se référer à l'aide-mémoire (*cheat sheet*) sur le site de RStudio.

## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Les blocs de code R sont intégrés dans R Markdown de la façon suivante :

```
```{r}``  
2 + 2  
```
```

## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Les blocs de code R sont intégrés dans R Markdown de la façon suivante :

```
```{r}  
2 + 2  
```
```

Par défaut **le code est évalué**, et **lui-même ainsi que ses résultats sont affichés** dans le document en sortie :

```
2 + 2
```

```
## [1] 4
```



## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Les **options** saisies en début de bloc permettent de préciser à `knitr` la manière de le prendre en compte, par exemple :

- ▶ `eval=FALSE` : le bloc n'est pas évalué ;
- ▶ `echo=FALSE` : le bloc n'est pas affiché ;
- ▶ `collapse=TRUE` : code et résultats sont affichés à la suite.

## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Les **options** saisies en début de bloc permettent de préciser à knitr la manière de le prendre en compte, par exemple :

- ▶ `eval=FALSE` : le bloc n'est pas évalué ;
- ▶ `echo=FALSE` : le bloc n'est pas affiché ;
- ▶ `collapse=TRUE` : code et résultats sont affichés à la suite.

```
```{r, echo=FALSE}  
2 + 2  
```
```

```
## [1] 4
```

## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Les **options** saisies en début de bloc permettent de préciser à knitr la manière de le prendre en compte, par exemple :

- ▶ `eval=FALSE` : le bloc n'est pas évalué ;
- ▶ `echo=FALSE` : le bloc n'est pas affiché ;
- ▶ `collapse=TRUE` : code et résultats sont affichés à la suite.

```
```{r, echo=FALSE}  
2 + 2  
```
```

```
## [1] 4
```

**Note** Toutes les options de knitr relatives aux blocs de code (*chunk options*) sont présentées sur la [page](#) du créateur du *package*, Yihui Xie.

## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Il est également possible d'intégrer le résultat d'un traitement R dans le corps d'un paragraphe avec la syntaxe :

```
`r`
```

## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Il est également possible d'intégrer le résultat d'un traitement R dans le corps d'un paragraphe avec la syntaxe :

```
`r      `
```

**Exemple** Pour intégrer dans le texte la date de compilation du document, utiliser

```
Document compilé le `r Sys.Date()``.
```

## Générer automatiquement des documents depuis R

### Ecrire du code dans R Markdown

Il est également possible d'intégrer le résultat d'un traitement R dans le corps d'un paragraphe avec la syntaxe :

```
`r      `
```

**Exemple** Pour intégrer dans le texte la date de compilation du document, utiliser

```
Document compilé le `r Sys.Date()``.
```

Document compilé le 2018-04-10.

## Générer automatiquement des documents depuis R

# Intégrer des graphiques dans R Markdown

Tous les graphiques produits par les blocs de code sont **automatiquement intégrés au fichier final**.

## Générer automatiquement des documents depuis R

# Intégrer des graphiques dans R Markdown

Tous les graphiques produits par les blocs de code sont **automatiquement intégrés au fichier final**.

Un **grand nombre d'options** sont consacrées au paramétrage des graphiques, notamment :

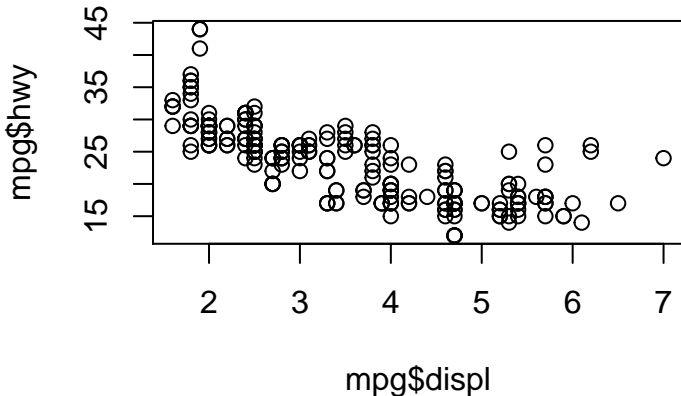
- ▶ `fig.width`, `fig.height` : largeur et hauteur utilisées pour produire le graphique, en pouces ;
- ▶ `fig.asp` : rapport hauteur/largeur (`fig.height` est neutralisé quand `fig.asp` est renseigné) ;
- ▶ `out.width`, `out.height` : largeur et hauteur du graphique dans la sortie finale ;
- ▶ `fig.align` : alignement du graphique ("`left`", "`right`" ou "`center`") ;
- ▶ `dpi` (72 par défaut) : résolution (utile uniquement pour HTML).



## Générer automatiquement des documents depuis R

### Intégrer des graphiques dans R Markdown

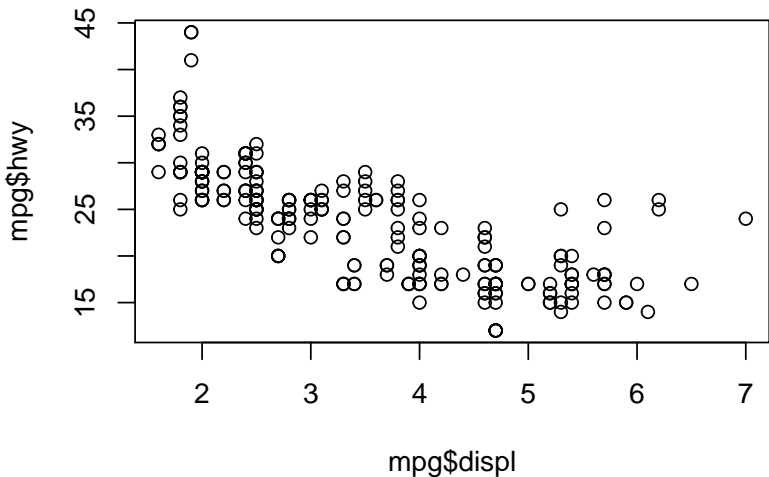
```
```{r, fig.asp = 3/4, fig.width = 4}  
plot(mpg$displ, mpg$hwy)  
```
```



## Générer automatiquement des documents depuis R

# Intégrer des graphiques dans R Markdown

```
```{r, fig.asp = 3/4, fig.width = 6, out.width = "4in"}  
plot(mpg$displ, mpg$hwy)  
```
```



## Générer automatiquement des documents depuis R

# Intégrer des tableaux dans R Markdown

Pour construire un tableau dans R Markdown, il suffit de le « dessiner » avec les signes – et | :

| Colonne 1 | Colonne 2 | Colonne 3 |
|-----------|-----------|-----------|
| 1         | a         | `TRUE`    |
| 2         | b         | `FALSE`   |

## Générer automatiquement des documents depuis R

### Intégrer des tableaux dans R Markdown

Pour construire un tableau dans R Markdown, il suffit de le « dessiner » avec les signes – et | :

```
Colonne 1	Colonne 2	Colonne 3
1          | a          | `TRUE`
2          | b          | `FALSE`
```

| Colonne 1 | Colonne 2 | Colonne 3 |
|-----------|-----------|-----------|
| 1         | a         | TRUE      |
| 2         | b         | FALSE     |

Les : permettent de spécifier l'alignement des colonnes.

# Générer automatiquement des documents depuis R

## Intégrer des tableaux dans R Markdown

En règle générale cependant, les tableaux à intégrer sont générés automatiquement à partir des données.

```
```{r}
resultat <- data.table(mpg)[
  , list(hwy=mean(hwy), cty=mean(cty)), by = drv
]
resultat
```
```

|       | drv | hwy      | cty     |
|-------|-----|----------|---------|
| ## 1: | f   | 28.16038 | 19.9717 |
| ## 2: | 4   | 19.17476 | 14.3301 |
| ## 3: | r   | 21.00000 | 14.0800 |

# Générer automatiquement des documents depuis R

## Intégrer des tableaux dans R Markdown

En règle générale cependant, les tableaux à intégrer sont générés automatiquement à partir des données.

```
```{r}
resultat <- data.table(mpg)[
  , list(hwy=mean(hwy), cty=mean(cty)), by = drv
]
resultat
```
```

| ##    | drv | hwy      | cty     |
|-------|-----|----------|---------|
| ## 1: | f   | 28.16038 | 19.9717 |
| ## 2: | 4   | 19.17476 | 14.3301 |
| ## 3: | r   | 21.00000 | 14.0800 |

La fonction `knitr::kable()` permet de **transformer un objet R en tableau formaté pour R Markdown**.

## Générer automatiquement des documents depuis R

# Intégrer des tableaux dans R Markdown

```
```{r, results = "asis"}  
knitr::kable(resultat)  
```
```

## Générer automatiquement des documents depuis R

### Intégrer des tableaux dans R Markdown

```
```{r, results = "asis"}  
knitr::kable(resultat)  
```
```

| drv | hwy      | cty     |
|-----|----------|---------|
| f   | 28.16038 | 19.9717 |
| 4   | 19.17476 | 14.3301 |
| r   | 21.00000 | 14.0800 |



# Générer automatiquement des documents depuis R

## Intégrer des tableaux dans R Markdown

```
```{r, results = "asis"}  
knitr::kable(resultat)  
```
```

```
drv	hwy	cty
f	28.16038	19.9717
4	19.17476	14.3301
r	21.00000	14.0800
```

Ce qui donne une fois formaté par R Markdown :

| drv | hwy      | cty     |
|-----|----------|---------|
| f   | 28.16038 | 19.9717 |
| 4   | 19.17476 | 14.3301 |
| r   | 21.00000 | 14.0800 |

# Générer automatiquement des documents depuis R

## Paramétrer un document R Markdown

La plupart des paramètres généraux du documents sont à indiquer dans son en-tête (désigné par l'acronyme YAML) :

```
---  
title: "Formation R Perfectionnement"  
author: "Martin Chevalier (Insee)"  
output:  
  html_document:  
    highlight: haddock  
    toc: yes  
    toc_depth: 2  
    toc_float: yes  
---
```

## Générer automatiquement des documents depuis R

### Paramétrer un document R Markdown

La plupart des paramètres généraux du documents sont à indiquer dans son en-tête (désigné par l'acronyme YAML) :

```
---
title: "Formation R Perfectionnement"
author: "Martin Chevalier (Insee)"
output:
  html_document:
    highlight: haddock
    toc: yes
    toc_depth: 2
    toc_float: yes
---
```

**Pour en savoir plus** Le site de RStudio documente le paramétrage de l'en-tête YAML selon les formats de sortie souhaités (html, pdf).