

R Markdown documentation : tables, graphs and equations

Jeanne, KC, Samba, Tamsir

2025-01-21

Présentations générale de Rmarkdown

RMarkdown est une extension du langage Markdown. Il permet d'intégrer dans un même fichier du texte explicatif, du code exécutable (en R ou dans d'autres langages comme Python) et les résultats d'analyse (tableaux, graphiques, etc.). RMarkdown est particulièrement utile pour produire des documents reproductibles, clairs et professionnels.

```
library(haven)
library(reshape2)
library(ggplot2)

## Warning: le package 'ggplot2' a été compilé avec la version R 4.4.2
library(kableExtra)

## Warning: le package 'kableExtra' a été compilé avec la version R 4.4.1
library(gtsummary)
library(stringr)

Base <- read_dta("Base_de_données.dta")

## Pour transformer en variables catégorielles celles qui ne le sont pas encore.

to_rec <- list()
for (i in names(Base)){
  if(is.null(attr(Base[[i]], "labels"))== 0){ # On met deux crochets pour préciser que c'est la colonne
    to_rec <- c(to_rec, i)
  }
}

# Création des variables recodées
for (i in to_rec ){
  label <- attr(Base[[i]], "label")

  Base[[i]] <- factor(Base[[i]], labels= str_to_title(names(attr(Base[[i]], "labels"))),
                       levels=unname(attr(Base[[i]], "labels")))
  attr(Base[[i]], "label") <- label

  Base[[i]] <- as.character(Base[[i]])

}
```

Structure générale

Un fichier RMarkdown est structuré en trois parties principales : l'en-tête YAML, le texte explicatif et les code chunks.

L'en-tête YAML contient les métadonnées et les options de configuration du document, comme le titre, l'auteur, la date et le format de sortie.

```
---
```

```
title: "Analyse des Données"
author: "Votre Nom"
date: "`r Sys.Date()`"
output: html_document
```

```
--
```

Figure 1: Le YALM

Le texte explicatif est écrit en Markdown, permettant de structurer le document avec des titres, sous-titres, listes et liens.

```
# Introduction
Ce rapport présente une analyse détaillée des données.

## Objectifs
- Comprendre les tendances principales.
- Identifier les relations entre les variables.

**Note :** Les analyses sont basées sur le jeu de données `mtcars`.
```

Figure 2: Texte du document

Les chunks permettent d'insérer et d'exécuter du code directement dans le document. Les résultats sont automatiquement intégrés.

```
```{r, echo=TRUE, message=FALSE}
Charger les données intégrées 'mtcars'
data("mtcars")
Afficher un résumé des données
summary(mtcars)
```

Figure 3: Code chunk

## Pourquoi utiliser RMarkdown ?

- Reproductibilité: Idéal pour documenter les analyses de manière reproductible.

- Gain de temps: Génère automatiquement des rapports sans copier-coller les résultats.
- Polyvalence: Produit des fichiers HTML interactifs, des PDF élégants ou des documents Word.
- Interopérabilité: Peut inclure du code en Python, SQL ou d'autres langages. Les options des chunks de code Chaque chunk peut être configuré pour personnaliser son exécution et son affichage.

Voici les options courantes :

- echo : Affiche ou cache le code source.
- eval : Exécute ou non le code.
- warning: Affiche ou supprime les avertissements.
- message: Affiche ou cache les messages.

Générer un document avec Knit Pour convertir un fichier “.Rmd” en document final :

1. Cliquez sur le bouton “Knit” dans RStudio.
2. Choisissez le format de sortie selon vos besoins (HTML, PDF ou Word).
3. Si nécessaire, installez les packages manquants (comme “rmarkdown”, “knitr” et “pandoc”).

## Faire des graphiques

### Paramètres de mise en forme du graphique

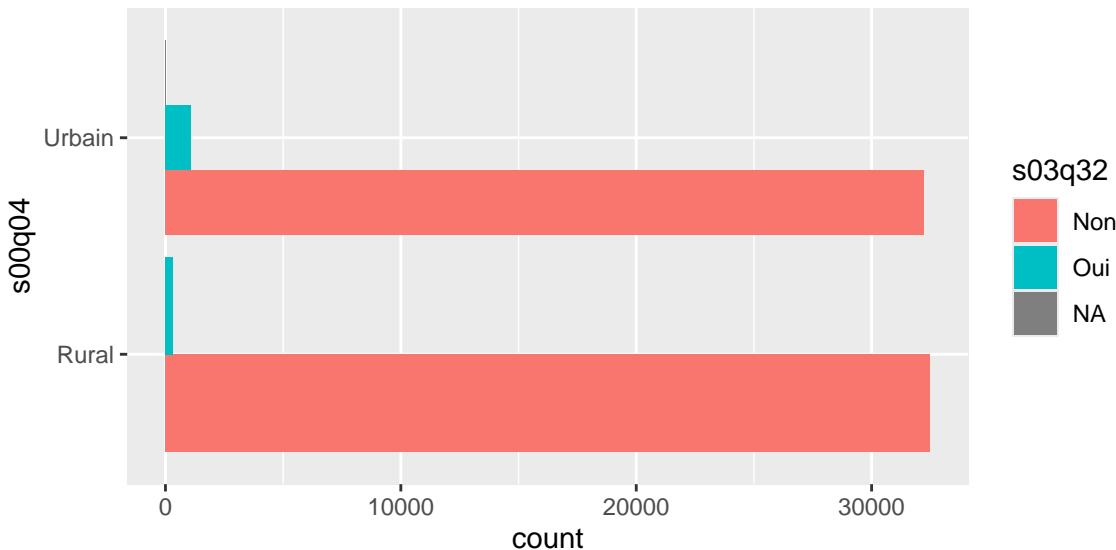
- Pour préciser la taille du graphique, on peut ajouter dans l'entête du chunk les paramètres suivants : (fig.height=4, fig.width=6) et ajuste la taille selon les besoins.
- Pour ne pas afficher les graphiques, on peut utiliser ce paramètre dans l'entête du chunk : (fig.show = “hide”).

A noter que ces options peuvent être déclarées au début pour tous les chunks, en paramètres généraux.

```
#'Avez-vous rencontré un problème de santé
#'durant les 12 derniers mois qui a entraîné une hospitalisation ?
```

```
#'Bénéficiez-vous d'une couverture maladie ?
```

```
ggplot(Base)+
 geom_bar(aes(x=s00q04, fill=s03q32), position="dodge")+
 coord_flip()
```



Un petit texte sur les heatmaps...

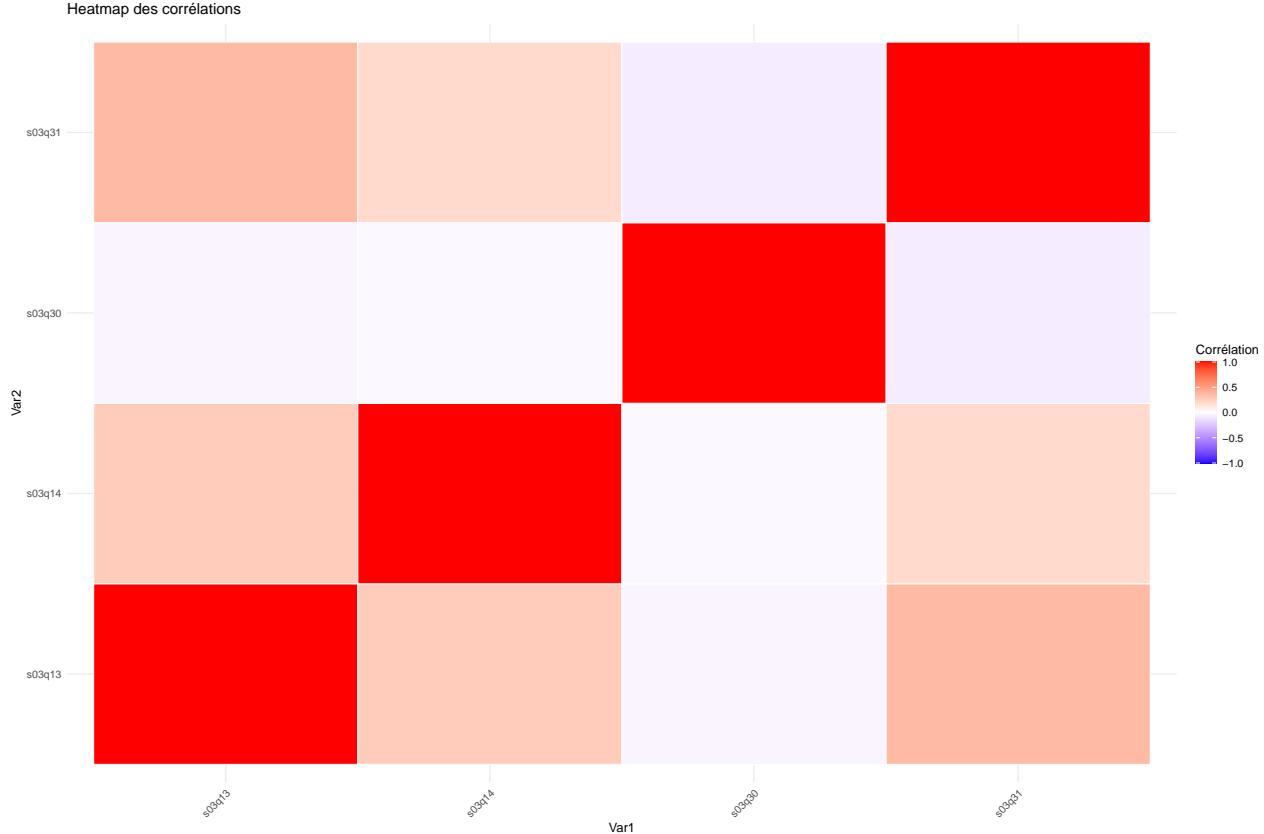
Les **heatmaps** (ou cartes de chaleur) sont des représentations graphiques utilisées pour visualiser des données sous forme de matrices, où chaque cellule est colorée en fonction de la valeur qu'elle représente. Ces visualisations sont particulièrement utiles pour identifier rapidement des motifs, des tendances, ou des anomalies dans des données complexes. Les heatmaps sont fréquemment utilisées en analyse de données, en bioinformatique (comme dans les études d'expression génique), en sciences sociales, et en machine learning pour des matrices de corrélation ou de confusion. Les nuances de couleurs utilisées, souvent issues de dégradés, aident à différencier les valeurs basses, moyennes et élevées, rendant les relations entre les variables plus intuitives et visuellement attrayantes.

```
variables_numeriques <- c("s03q13", "s03q14", "s03q30", "s03q30", "s03q31")

Calcul de la matrice de corrélation
cor_matrix <- cor(Base[variables_numeriques], use = "complete.obs")

Création de la heatmap
g <- ggplot(melt(cor_matrix), aes(Var1, Var2, fill = value)) +
 geom_tile(color = "white") +
 scale_fill_gradient2(low = "blue", high = "red", mid = "white",
 midpoint = 0, limit = c(-1, 1), space = "Lab",
 name = "Corrélation") +
 labs(title = "Heatmap des corrélations") +
 theme_minimal() +
 theme(axis.text.x = element_text(angle = 45, vjust = 1, hjust = 1))
```

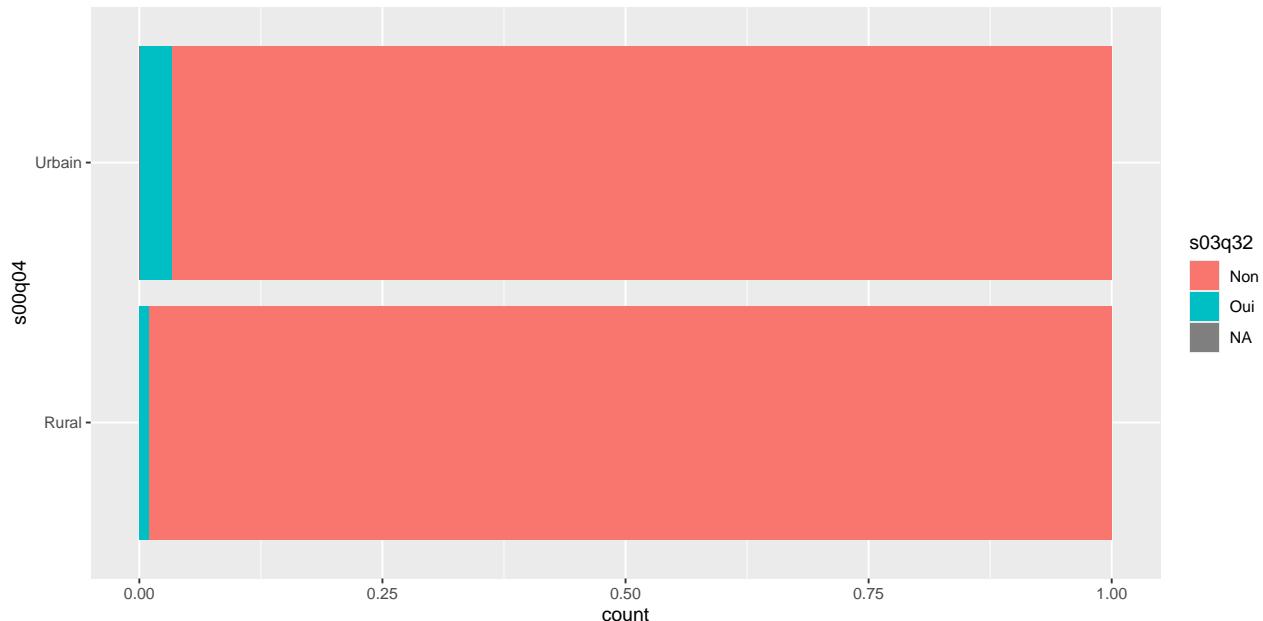
g



Quelques jolis graphiques....

```
#'Bénéficiez-vous d'une couverture maladie ?
```

```
ggplot(Base)+
 geom_bar(aes(x=s00q04, fill=s03q32), position="fill") +
 coord_flip()
```

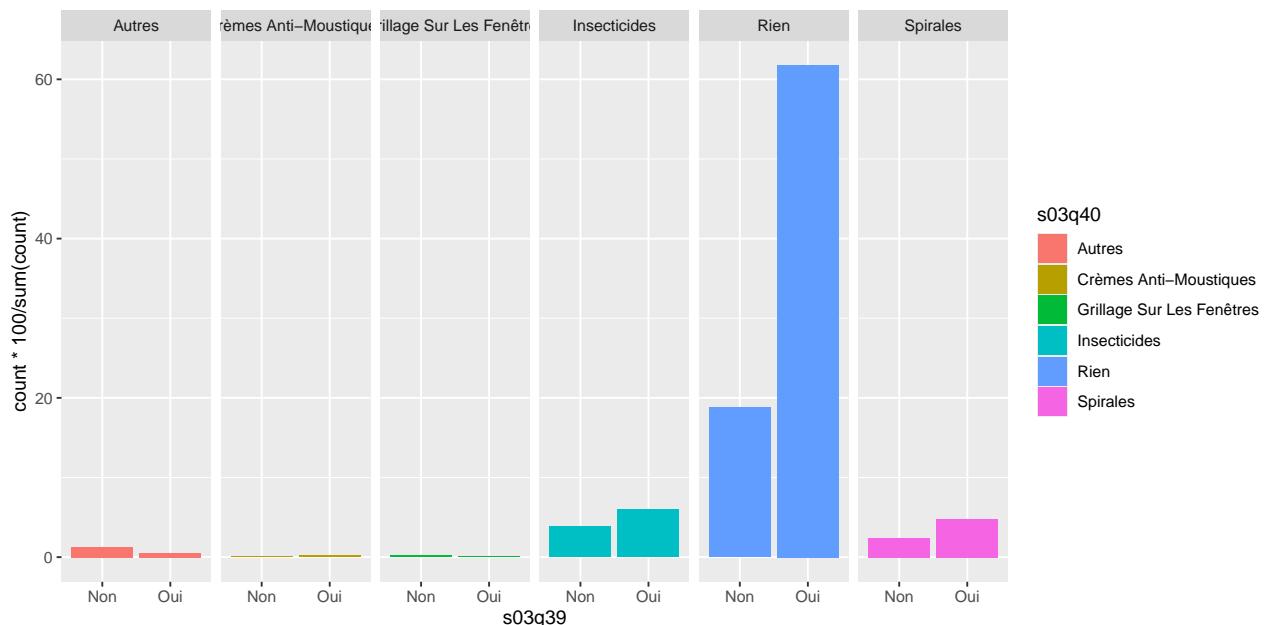


#### Code chunk

```
#'s03q40: Moyens de prévention du paludisme autres que moustiquaire,
#'suivant la tendance à utiliser une moustiquaire.
```

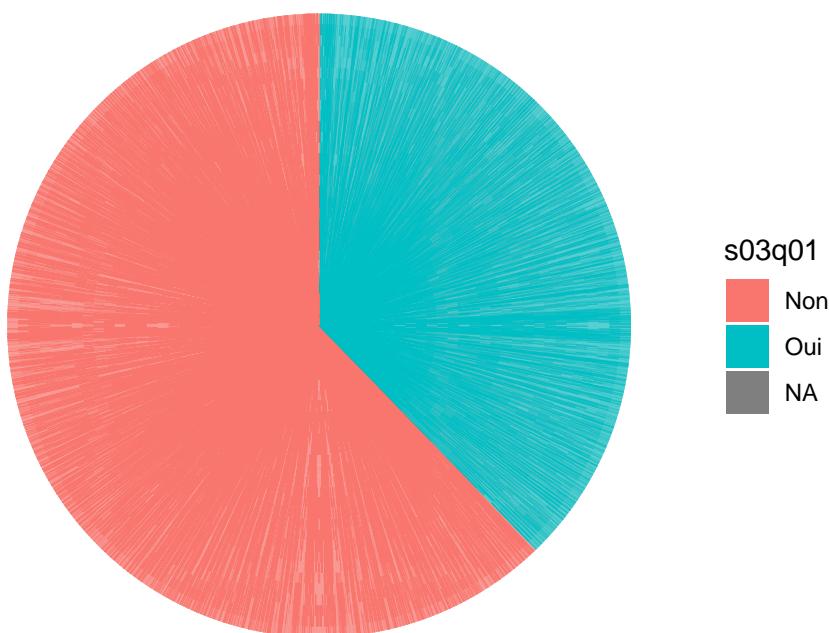
```
#'s03q38: Dormez-vous habituellement sous une moustiquaire ?
```

```
Base1 <- Base[!is.na(Base$s03q39),]
Base1 %>%
 #filter(!is.na(s03q39)) %>%
ggplot() +
 geom_bar(aes(x=s03q39, y=after_stat(count*100/sum(count))), fill=s03q40) +
 facet_grid(cols= vars(s03q40))
```



```
ggplot(Base) +
 geom_bar(aes(x = "", y = s03q01, fill = s03q01), width = 1, stat = "identity") +
 coord_polar("y", start = 0) +
 theme_void() +
 labs(title = "Répartition des individus suivant la rencontre ou non d'un problème de santé pendant la nuit")
```

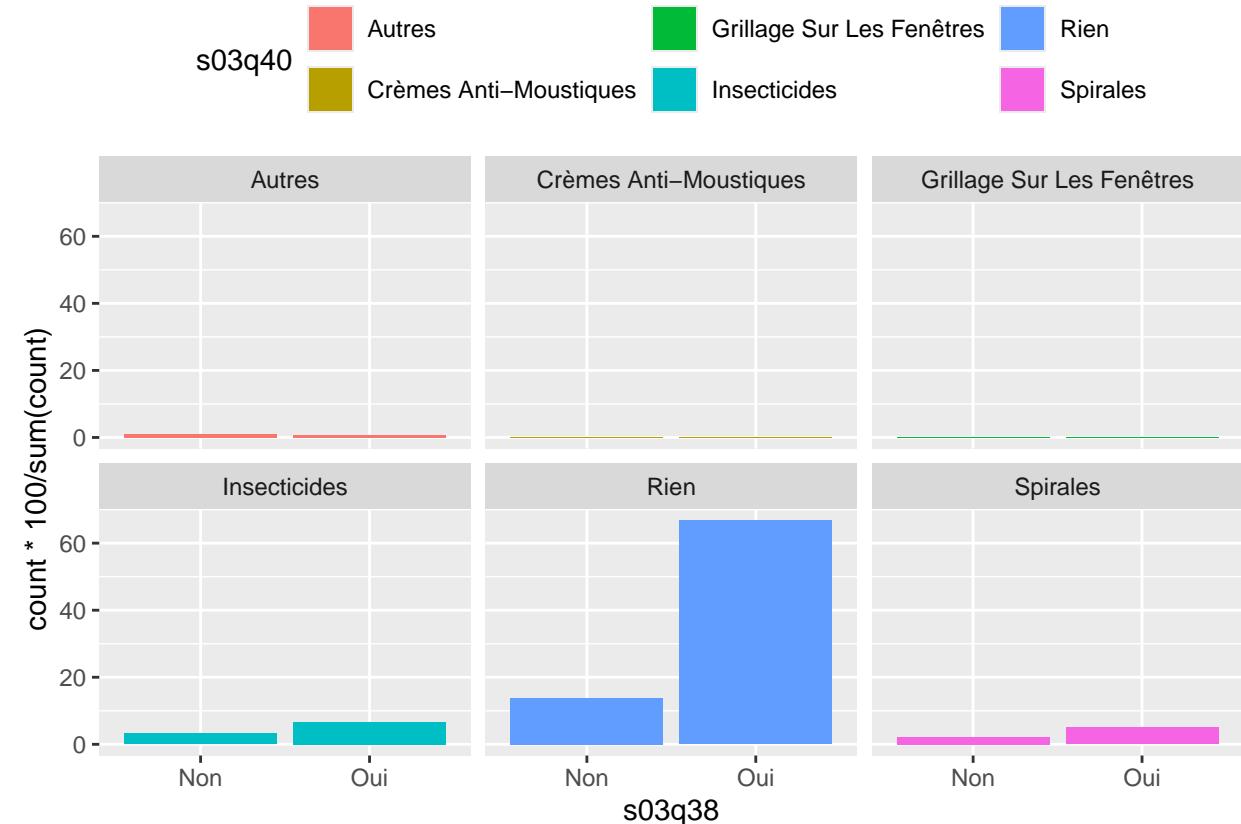
Répartition des individus suivant la rencontre ou non d'un problème de santé pendant la nuit



Source : ANSD, EHCVM 2018/2019, calculs propres

```
Utilisez-vous des protections anti-moustiques autres que les moustiquaires ?
```

```
Base[!is.na(Base$s03q38),] %>%
ggplot() +
 geom_bar(aes(x=s03q38,y=after_stat(count*100/sum(count)), fill=s03q40)) +
 facet_wrap(vars(s03q40), nrow=2) +
 theme(legend.position = "top")
```



## Faire des tableaux

On peut réaliser des tableaux avec sur Rmarkdown, avec la fonction kable() du package knitr ou en utilisant les packages kableExtra et gtsummary.

### kable() et le package kableExtra()

Faisons un petit tableau.

```
L'individu utilise t-il d'autres protections qu'une moustiquaire ?
t <- table(Base$s03q40)
t
```

```
##
Autres Crèmes Anti-Moustiques Grillage Sur Les Fenêtres
1176 197 214
Insecticides Rien Spirales
6509 53268 4738
```

Pas très attrayant... Et si on rendait cela plus regardable ?

Var1	Freq
Autres	1176
Crèmes Anti-Moustiques	197
Grillage Sur Les Fenêtres	214
Insecticides	6509
Rien	53268
Spirales	4738

```
kable(t)
```

Var1	Freq
Autres	1176
Crèmes Anti-Moustiques	197
Grillage Sur Les Fenêtres	214
Insecticides	6509
Rien	53268
Spirales	4738

Nous pouvons changer ajouter un titre et changer le style

```
t %>%
kable(format = "latex")
```

Var1	Freq
Autres	1176
Crèmes Anti-Moustiques	197
Grillage Sur Les Fenêtres	214
Insecticides	6509
Rien	53268
Spirales	4738

```
t %>%
kable(format = "latex", booktable = TRUE)
```

Var1	Freq
Autres	1176
Crèmes Anti-Moustiques	197
Grillage Sur Les Fenêtres	214
Insecticides	6509
Rien	53268
Spirales	4738

Avec kable\_style() de kableExtra, on peut aller plus loin. Notons que cette fonction s'applique à un tableau kable.

```
t %>%
kable("latex", booktabs = T)%>%
kable_styling(latex_options = "striped")
```

On peut en sorte que cela prenne toute la largeur de la page et on peut ajuster la taille de police.

```
t %>%
kable("latex", booktabs = T)%>%
kable_styling(full_width = TRUE, font_size = 6)
```

Var1	Freq
Autres	1176
Crèmes Anti-Moustiques	197
Grillage Sur Les Fenêtres	214
Insecticides	6509
Rien	53268
Spirales	4738

Et si les colonnes de notre dztaframe sont nombreuses, on peut s'arranger pour qu'elles s'affichent toutes avec l'option “scale\_down”. Comparez !

```
Base[1:10, 1:12] %>%
 kable("latex", booktabs = T)
```

vague	grappe	menage	s01q00a	s00q01	s00q04	s01q01	s03q00	s03q01	s03q02	s03q03	s03q04
1	1	1	1	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	1	1	2	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	1	2	1	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	1	2	2	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	1	3	1	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	1	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	2	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	3	1	3	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	4	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	5	Dakar	Urbain	Féminin	1	Non	NA	NA	NA

Et...

```
Base[1:10, 1:12] %>%
 kable() %>%
 kable_styling(latex_options = c("striped", "scale_down"))
```

```
Warning in styling_latex_scale(out, table_info, "down"): Longtable cannot be
resized.
```

vague	grappe	menage	s01q00a	s00q01	s00q04	s01q01	s03q00	s03q01	s03q02	s03q03	s03q04
1	1	1	1	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	1	1	2	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	1	2	1	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	1	2	2	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	1	3	1	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	1	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	2	Dakar	Urbain	Féminin	1	Non	NA	NA	NA
1	3	1	3	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	4	Dakar	Urbain	Masculin	1	Non	NA	NA	NA
1	3	1	5	Dakar	Urbain	Féminin	1	Non	NA	NA	NA

Aussi, il est possible de sectionner des lignes ou des colonnes spécifiques et de leur donner un style rien qu'à eux!

```
head(Base[1:10, 1:10]) %>%
 kable("latex", booktabs = TRUE) %>%
 kable_styling(latex_options = "scale_down") %>%
```

vague	grappe	menage	s01q00a	s00q01	s00q04	s01q01	s03q00	s03q01	s03q02
1	1	1	1	Dakar	Urbain	Féminin	1	Non	NA
1	1	1	2	Dakar	Urbain	Féminin	1	Non	NA
1	1	2	1	Dakar	Urbain	Masculin	1	Non	NA
1	1	2	2	Dakar	Urbain	Féminin	1	Non	NA
1	1	3	1	Dakar	Urbain	Masculin	1	Non	NA
1	3	1	1	Dakar	Urbain	Masculin	1	Non	NA

```
column_spec(1, bold = TRUE) %>%
column_spec(5, color = "red") %>%
row_spec(1, bold = T, color = "white", background = "black", italic = TRUE) %>%
row_spec(0, angle = 45, bold = TRUE)
```

## gtsummary

gtsummary permet de réaliser des tableaux statistiques combinant plusieurs variables, l'affichage des résultats pouvant dépendre du type de variables.

Par défaut, gtsummary considère qu'une variable est catégorielle s'il s'agit d'un facteur, d'une variable textuelle ou d'une variable numérique ayant moins de 10 valeurs différentes.

Une variable sera considérée comme dichotomique (variable catégorielle à seulement deux modalités) s'il s'agit d'un vecteur logique (TRUE/FALSE), d'une variable textuelle codée yes/no ou d'une variable numérique codée 0/1.

Dans les autres cas, une variable numérique sera considérée comme continue.

On peut réaliser des tableaux croisés intéressants...

```
tbl_summary(
 Base,
 include = c(s01q01, s03q38, s03q34), # sexe, moustiquaire, finacmt de l'assurance
 by = s00q04, # group
 missing = "no" # don't list missing data separately
) %>%
add_n() %>% # add column with total
modify_header(label = "**Variable**") %>% # column header
bold_labels()

Table printed with `knitr::kable()`, not {gt}. Learn why at
https://www.danielsjoberg.com/gtsummary/articles/rmarkdown.html
To suppress this message, include `message = FALSE` in code chunk header.
```

Variable	N	Rural, N = 32,798	Urbain, N = 33,322
s01q01	66,118		
Féminin		17,527 (53%)	17,789 (53%)
Masculin		15,271 (47%)	15,531 (47%)
s03q38	66,102		
Non		5,475 (17%)	8,131 (24%)
Oui		27,323 (83%)	25,173 (76%)
s03q34	1,414		

Variable	N	Rural, N = 32,798	Urbain, N = 33,322
Autre (À Préciser)		23 (7.3%)	122 (11%)
Employeur		28 (8.8%)	181 (16%)
Etat Employeur		28 (8.8%)	149 (14%)
Etat Et Privé		8 (2.5%)	14 (1.3%)
Etat/Programme		149 (47%)	447 (41%)
Privé (Individuel)		81 (26%)	184 (17%)

On peut faire cela, et d'autres choses encore...

## COMMENT FAIRE DES MATHS AVEC RMARKDOWN (EQUATIONS ET AUTRES) ?

### PARTIE 1 : Des formules qui apparaissent dans le sommeil...

Ramanujan EST UN GENIE DES MATHS HORS DU COMMUN. Ramanujan disait que les solutions mathématiques lui étaient **chuchotées par des divinités dans ses rêves**. Lorsqu'il se réveillait, il les notait immédiatement. Cette image illustre bien son génie:

#### Qui était Ramanujan?

Voici quelques points essentiels sur sa vie:

- **Nom complet** : Srinivasa Ramanujan (1887-1920).
- **Origine** : Né dans une famille pauvre en Inde, dans le village d'Erode.
- **Formation** : Appris les mathématiques avec seulement **deux livres** :
  - “A Synopsis of Elementary Results” de G.S. Carr.
  - Des carnets où il notait ses propres idées.
- **Découverte** : Ses travaux ont été reconnus par le mathématicien **G.H. Hardy**, qui l'invita à Cambridge, en Angleterre.
- **Décès prématué** : Mort à 32 ans, mais ses contributions continuent d'influencer les mathématiques modernes.

---

Srinivasa Ramanujan (1887-1920) est l'un des plus grands mathématiciens autodidactes de l'histoire. Né en Inde dans une famille modeste, il a appris les mathématiques presque **seul**, en étudiant **deux livres**: - “A Synopsis of Elementary Results” de G.S. Carr. - Ses propres carnets où il notait ses idées.

Ses travaux, bien qu'étonnantes et souvent sans démonstration, ont impressionné le célèbre mathématicien britannique **G.H. Hardy**, qui l'invita à Cambridge. En dépit de sa courte vie (il est mort à 32 ans), Ramanujan a laissé un héritage monumental.

---

#### Anecdotes fascinantes sur Ramanujan

- **Le mystère des carnets** : Ramanujan a laissé derrière lui des milliers de formules mathématiques dans des carnets. Certaines n'ont été prouvées que des **décennies après sa mort!**
- **Le taxi 1729** : Alors qu'il était à l'hôpital, Hardy lui rend visite et mentionne que le taxi qu'il a pris avait un numéro “ennuyeux” : **1729**. Ramanujan répond :  
“Non! C'est le plus petit nombre qui peut s'exprimer comme la somme de deux cubes de deux manières



Figure 4: Ramanujan réveille les mathématiques

*différentes.”*

$$1729 = 1^3 + 12^3 = 9^3 + 10^3$$

---

## Contributions mathématiques majeures

Ramanujan a marqué les mathématiques avec des formules à la fois incroyables et pratiques. Voici quelques-unes de ses contributions :

### 1. Formule pour calculer $\pi$

Cette formule est utilisée aujourd’hui pour calculer  $\pi$  avec une précision impressionnante :

$$\frac{1}{\pi} = \frac{2\sqrt{2}}{9801} \sum_{n=0}^{\infty} \frac{(4n)!(1103 + 26390n)}{(n!)^4 396^{4n}}$$

### 2. La somme infinie des entiers

Ramanujan a démontré que :

$$1 + 2 + 3 + 4 + \dots = -\frac{1}{12}$$

Bien que cela semble paradoxal, cette formule a des applications en physique théorique, notamment dans la théorie des cordes.

### 3. La fonction partition

Ramanujan a étudié la fonction **partition**  $p(n)$ , qui compte le nombre de façons de décomposer un entier en somme d’entiers positifs. Par exemple :

$$p(4) = 5 \quad \text{car } 4 = 4, 3 + 1, 2 + 2, 2 + 1 + 1, 1 + 1 + 1 + 1$$

Il a découvert des propriétés remarquables sur cette fonction, comme :

$$p(5) = \frac{1}{2} \left( \sqrt{5} + 1 \right)^5$$

### 4. Les séries hypergéométriques

Ramanujan a exploré les séries hypergéométriques, une classe d’équations infinies utiles en analyse complexe et en physique :

$$1 + \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \dots = \frac{\pi^2}{6}$$

### ### 5. Identité de Ramanujan pour les fonctions thêta

Ramanujan a étudié les **fonctions thêta**, qui jouent un rôle crucial en analyse complexe, en théorie des nombres et même en physique. Voici une de ses célèbres identités :

$$\left( \sum_{n=-\infty}^{\infty} q^{n^2} \right)^2 = \sum_{n=-\infty}^{\infty} q^{n(n+1)}$$

Cette équation relie deux séries infinies, où  $q$  est un paramètre complexe, avec  $|q| < 1$ . Ces identités apparaissent dans les formes modulaires, un domaine clé de la théorie des nombres.

---

### Exemple spécifique avec $q = e^{-\pi}$

Lorsque  $q = e^{-\pi}$ , une version particulière de cette identité devient :

$$(1 + 2q + 2q^4 + 2q^9 + \dots)^2 = 1 + 2q^2 + 2q^6 + 2q^{12} + \dots$$

Les fonctions thêta sont utilisées aujourd’hui en cryptographie, en théorie des partitions, et en physique théorique.

---

## PARTIE 2 : Premiers pas sur les maths avec RMarkdown

Maintenant que nous avons vu des exemples impressionnantes issus des travaux de Ramanujan, explorons comment écrire des mathématiques dans un fichier RMarkdown. Nous allons commencer par les bases.

RMarkdown utilise le langage **LaTeX** pour afficher des équations mathématiques. Voici les deux manières principales d’écrire des mathématiques :

1. **Équations en ligne** : Utilisez  $\$... \$$  pour insérer une équation dans une phrase.
  2. **Équations en bloc** : Utilisez  $\$\$... \$\$$  pour afficher une équation sur une ligne séparée et centrée.
- 

### Équations en ligne

Les équations en ligne sont utiles lorsque vous voulez insérer des mathématiques au milieu d’une phrase.

#### Exemple 1 : Texte avec mathématiques

““markdown La formule de Pythagore est donnée par  $a^2 + b^2 = c^2$ .

#### Exemple 1 : Formule d’énergie

Code LaTeX :

```
$$
E = mc^2
$$
```

Rendu mathématique :

$$E = mc^2$$

---

#### Exemple 2 : Fonction rationnelle

Code LaTeX :

```
$$
y = \frac{x^2 + 3x + 2}{x - 1}
$$
```

Rendu mathématique :

$$y = \frac{x^2 + 3x + 2}{x - 1}$$


---

### Exemple 3 : Somme, produit et intégrale

Code LaTeX :

```
$$
S = \sum_{i=1}^n i, \quad P = \prod_{i=1}^n x_i, \quad \int_0^1 x^2 \, dx
$$
```

Rendu mathématique :

$$S = \sum_{i=1}^n i, \quad P = \prod_{i=1}^n x_i, \quad \int_0^1 x^2 \, dx$$


---

### Exemple 4 : Fonction polynomiale

Code LaTeX :

```
$$
f(x) = x^2 + \text{terme constant}
$$
```

Rendu mathématique :

$$f(x) = x^2 + \text{terme constant}$$


---

### Exemple 5 : Matrice

Code LaTeX :

```
$$
\begin{matrix}
1 & 2 & 3 \\
4 & 5 & 6 \\
7 & 8 & 9
\end{matrix}
$$
```

Rendu mathématique :

$$\begin{matrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{matrix}$$


---

## Exemple 6 : Système d'équations

Code LaTeX :

```
$$
\begin{aligned}
x + y &= 10 \\
2x - y &= 5
\end{aligned}
$$
```

Rendu mathématique :

$$\begin{aligned} x + y &= 10 \\ 2x - y &= 5 \end{aligned}$$

## Tableaux récapitulatifs des commandes LaTeX

Voici des tableaux utiles pour apprendre et utiliser les commandes mathématiques dans RMarkdown avec LaTeX.

---

**Tableau 1 : Opérations mathématiques de base**

Commande	Description	Exemple LaTeX	Rendu
<code>+, -, \times, /</code>	Opérations de base	<code>\$a + b\$, \$a \times b\$</code>	$a + b, a \times b$
<code>\frac{num}{den}</code>	Fraction	<code>\$\frac{a}{b}\$</code>	$\frac{a}{b}$
<code>^</code>	Puissance	<code>\$x^2\$</code>	$x^2$
<code>-</code>	Indice	<code>\$x_1\$</code>	$x_1$
<code>\sqrt{}</code>	Racine carrée	<code>\$\sqrt{a+b}\$</code>	$\sqrt{a+b}$

---

**Tableau 2 : Sommes, produits et intégrales**

Commande	Description	Exemple LaTeX	Rendu
<code>\sum_{i=1}^n</code>	Somme	<code>\$\sum_{i=1}^n i\$</code>	$\sum_{i=1}^n i$
<code>\prod_{i=1}^n</code>	Produit	<code>\$\prod_{i=1}^n x_i\$</code>	$\prod_{i=1}^n x_i$
<code>\int_a^b</code>	Intégrale définie	<code>\$\int_a^b x^2 dx\$</code>	$\int_a^b x^2 dx$

---

**Tableau 3 : Commandes avancées pour matrices et alignements**

Commande	Description	Exemple LaTeX	Rendu
<code>\begin{matrix}</code>	Matrice simple	<code>\begin{matrix} 1 &amp; 2 \\ 3 &amp; 4 \end{matrix}</code>	$\begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix}$

Commande	Description	Exemple LaTeX	Rendu
\left(...\right)	Parenthèses autour d'une matrice	\left( \begin{matrix} 1 & 2 \\ 3 & 4 \end{matrix} \right)	$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}$
\begin{aligned}	Alignement des équations	\begin{aligned} x + y &= 10 \\ y &= 10 \\ 2x - y &= 5 \end{aligned}	$\begin{aligned} x + y &= 10 \\ y &= 10 \\ 2x - y &= 5 \end{aligned}$

Tableau 4 : Commandes pour inclure du texte dans les équations

Commande	Description	Exemple LaTeX	Rendu
\text{}	Texte dans une équation	\$f(x) = x^2 + \text{terme constant}\$	$f(x) = x^2 + \text{terme constant}$
\quad	Espacement dans une équation	\$x^2 + 3x \quad \text{et} \quad x - 1\$	$x^2 + 3x \quad \text{et} \quad x - 1$

## Équations et rendus des équations de RAMANUJAN

Code LaTeX :

```
$$
\frac{1}{\pi} = \frac{2\sqrt{2}}{9801} \sum_{n=0}^{\infty} \frac{(4n)!(1103 + 26390n)}{(n!)^4 396^{4n}}
$$
```

Rendu mathématique :

$$\frac{1}{\pi} = \frac{2\sqrt{2}}{9801} \sum_{n=0}^{\infty} \frac{(4n)!(1103 + 26390n)}{(n!)^4 396^{4n}}$$

Code LaTeX :

```
$$
1 + 2 + 3 + 4 + \dots = -\frac{1}{12}
$$
```

Rendu mathématique :

$$1 + 2 + 3 + 4 + \dots = -\frac{1}{12}$$

Code LaTeX :

```
$$
p(4) = 5 \quad \text{car } 4 = 4, \ 3+1, \ 2+2, \ 2+1+1, \ 1+1+1+1
$$
```

Rendu mathématique :

$$p(4) = 5 \quad \text{car } 4 = 4, 3 + 1, 2 + 2, 2 + 1 + 1, 1 + 1 + 1 + 1$$

---

Code LaTeX :

```
$$
p(5) = \frac{1}{2} \left(\sqrt{5} + 1 \right)^5
$$
```

Rendu mathématique :

$$p(5) = \frac{1}{2} \left( \sqrt{5} + 1 \right)^5$$

---

Code LaTeX :

```
$$
1 + \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \dots = \frac{\pi^2}{6}
$$
```

Rendu mathématique :

$$1 + \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \dots = \frac{\pi^2}{6}$$

---

Code LaTeX :

```
$$
\left(\sum_{n=-\infty}^{\infty} q^{n^2} \right)^2 = \sum_{n=-\infty}^{\infty} q^{n(n+1)}
$$
```

Rendu mathématique :

$$\left( \sum_{n=-\infty}^{\infty} q^{n^2} \right)^2 = \sum_{n=-\infty}^{\infty} q^{n(n+1)}$$

---

Code LaTeX :

```
$$
\left(1 + 2q + 2q^4 + 2q^9 + \dots \right)^2 = 1 + 2q^2 + 2q^6 + 2q^{12} + \dots
$$
```

Rendu mathématique :

$$(1 + 2q + 2q^4 + 2q^9 + \dots)^2 = 1 + 2q^2 + 2q^6 + 2q^{12} + \dots$$

## Définition des systèmes linéaires

Considérons les systèmes suivant :

$$(1) \begin{cases} 2x + y - z = 8 \\ -3x - y + 2z = -11 \\ -2x + y + 2z = -3 \end{cases}$$

$$(2) \begin{cases} 2x + y - z = 8 \\ -3x - y = -1 \\ y + 2z = 0 \end{cases}$$

$$(3) \begin{cases} x + y - z = 0 \\ -5x - y + 2z = -1 \\ -2x + y + 7z = 2 \end{cases}$$

## Exemple de résolution

La solution du système (1) est :

```
Définition du système linéaire (1)
A <- matrix(c(2, 1, -1,
 -3, -1, 2,
 -2, 1, 2),
 nrow = 3, byrow = TRUE) # Matrice des coefficients

b <- c(8, -11, -3) # Vecteur des termes constants

Résolution du système Ax = b
solution <- solve(A, b)

Affichage de la solution
names(solution) <- c("x", "y", "z")
solution

x y z
2 3 -1
```

## R markdown de façon générale

Dans cette partie, nous présentons quelques sorties ainsi que les fonctionnalités principales de R Markdown.

### Sorties de R Markdown

R Markdown offre plusieurs formats de sortie adaptés à différents besoins parmi lesquelles :

## HTML

Le format HTML permet de générer des documents interactifs consultables dans un navigateur. Ce type de sortie est idéal pour inclure des graphiques dynamiques, des animations ou des tableaux interactifs. Il est souvent utilisé pour partager des rapports en ligne ou pour créer des tableaux de bord via des frameworks comme **flexdashboard** ou **Shiny**.

## PDF

Le format PDF est utile pour produire des documents au rendu professionnel, adaptés à l'impression. Les rapports en PDF bénéficient de la mise en page précise offerte par LaTeX, nécessaire pour la génération de ce type de fichier. Ce format convient particulièrement aux rapports officiels et publications académiques.

## Word

La sortie au format Word génère un fichier **.docx** modifiable. Ce format est privilégié lorsque le document doit être collaboratif ou lorsque des modifications ultérieures par d'autres utilisateurs sont prévues. Les rapports Word sont simples à personnaliser et à partager.

## Texte et mise en forme

Pour ce qui est des mise en forme, nous allons explorer quelques fonctionnalités basiques.

- **Gras** : **\*\*texte\*\*** → **texte**
- **Italique** : **\*texte\*** → **texte**
- **Texte barré** : **~~texte~~** → **texte**
- **Code en ligne** : **`code`** → **code**

## Listes

Pour ce qui est des listes on a : **### Des listes à puces**

- Élément 1
- Élément 2
  - Sous-élément

## Des listes numérotée

1. Premier
2. Deuxième
3. Troisième

## Tables

### Tableau basique en Markdown

Colonne 1	Colonne 2
Valeur 1	Valeur 2
Valeur 3	Valeur 4

## Titres et table des matières

### Titre de niveau 1

Les titres de niveau 1 sont présentés comme suit : # Titre 1

## **Titre de niveau 2**

Pour les titres de niveau 2 on a : ## Titre 2

## **Titre de niveau 3**

Et les titres de niveau 3 : ### Titre 3

## **Options avancées**

### **Personnalisation avec CSS**

Incluez un fichier CSS personnalisé dans le YAML :

```
output:
 html_document:
 css: styles.css
```

### **Personnalisation pour PDF**

Ajoutez des commandes LaTeX ou un fichier .tex personnalisé :

```
output:
 pdf_document:
 includes:
 in_header: header.tex
```

### **Mode présentation**

Utilisez le format ioslides pour créer une présentation :

```
output: ioslides_presentation
```

---