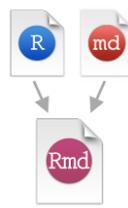


R Markdown cheat Sheet

learn more at rmarkdown.rstudio.com



.Rmdファイル

R Markdown (.Rmd)ファイルはあなたの研究の記録です。このファイルには科学者たちがあなたの研究を再現し、理解するのに必要なコードと解説が盛り込まれています。



再現可能な研究

ボタンをクリックするかコマンドを打ち込むとR Markdownファイル内でやった事が再実行されたり、完成したレポートとして結果をエクスポートしたりする事ができます。



動的なドキュメント

完成したレポートはhtml, pdf, MS Word, ODT, RTFまたはmarkdownドキュメントとして保存する事が出来ます。htmlやpdfベースのスライドショーとしても保存できます。

ワークフロー

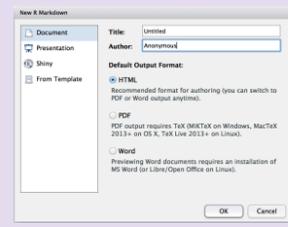
1 新しい.Rmdファイルを作成する File ▶ New File ▶ R Markdown から、予め準備されたテンプレートをウィザード形式で選ぶ事が出来ます。

2 ドキュメントを作成する テンプレートを元に編集していきます。

3 ドキュメントを編んで(knit)レポートを作成する knitボタンをクリックするか、render()を実行します。

4 プレビュー IDEウインドウ内で確認できます。

5 公開(オプション) ウェブやサーバへ公開できます。



ウィンドウで開く

保存

スペルチェック

置換

公開

アウトライン表示

プレビュー

コードチャ

コードチャンク

コードチャンクの

実行

チャンクの最初からこのチャンクまでの全チャンクを実行

現在のチャンクの

変更

実行

report.html

R Markdown

RStudio

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.

summary(cars)

| ## | 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 |

render()

markdown::render()を使うとコマンドラインでレンダー/knitが出来ます。重要な引数は...

input - レンダリングするファイル

output_format - レンダリングオプションのリスト

output_file

output_dir

params - 使用するパラメータリスト

envir - コードチャンクを評価する環境

encoding - インพุットファイルのエンコード

publishボタンは次のアカウントと同期できます。

- rpubs.com
- shinyapps.io
- RStudio Connect

ドキュメントのリロード

ドキュメント内検索

ドキュメントの出力先

ビルドログの確認

R Markdownコンソール内で出来ます。

アウトプットファイルの使用

.Rmdファイルと一緒に保存されます

.Rmdの構成

YAMLヘッダー
レンダりのオプション部分(例:pandoc)で設定対象:値の形式(YAML)で記述します。

- ファイルの最初に書く
- --- に挟まれた行の間

テキスト
markdownやコードチャンク形式でフォーマットした記述

コードチャンク
埋め込みしたコードのチャンクは、次の部分を指します。

- チャンクの開始は``{r}``
- チャンクの終わりは````

コードの実行結果は、ドキュメント内に追加されます。

.Rmdファイルの保存されている場所が作業ディレクトリとして使われます。

```

1 ---
2 title: "R Markdown"
3 author: "RStudio"
4 output:
5   html_document: 先の設定
6   toc: TRUE
7 ---
8
9 ```{r setup, include=FALSE}
10 knitr::opts_chunk$set(echo = TRUE)
11 ```
12
13 ## R Markdown
14
15 This is an R Markdown document.
16 Markdown is a simple formatting
17 syntax for authoring HTML, PDF,
18 and MS Word documents.
19
20 ```{r cars}
21 summary(cars)
22 ```
23
24 For more details on using R Markdown
25 see <http://rmarkdown.rstudio.com>.
  
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents.

summary(cars)

| ## | 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 |

For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

インタラクティブドキュメント

4ステップでレポートをインタラクティブなShinyドキュメントにしましょう。



- 1 YAMLヘッダーにruntime: shinyを追加
- 2 インพุットオブジェクトを埋め込むため、Shinyのinput関数を呼び出す
- 3 リアクティブな出力をさせるため、Shinyのrender関数を呼び出す
- 4 markdown::run()からRStudio IDEのRun Documentボタンをクリックしてレンダーする

```

---
output: html_document
runtime: shiny
---

```{r, echo = FALSE}
numericInput("n",
 "How many cars?", 5)

renderTable({
 head(cars, input$n)
}),

```

How many cars?

5

|   | speed | dist  |
|---|-------|-------|
| 1 | 4.00  | 2.00  |
| 2 | 4.00  | 10.00 |
| 3 | 7.00  | 4.00  |
| 4 | 7.00  | 22.00 |
| 5 | 8.00  | 16.00 |

shiny::shinyAppDir()でapp全体を埋め込むことができます

\* レポートはShinyアプリとしてレンダーされます。つまり、html\_documentのようにhtmlの出力フォーマットを必ず選択する必要があり、アクティブなRセッションで走らされます。

## コードの埋め込みとknitrシンタックス

### インラインコード

``r<code>``で結果のみを文中に出力させられます。

```
Built with
r getReversion() → Built with 3.2.3
```

### コードチャンク

````{r}``から````で囲んだ1行以上の部分です。{}の中のrの後にチャンクオプションを記述します。RStudioのCアイコンでも挿入できます。

```
```{r echo = TRUE}
getReversion()
```
## [1] '3.2.3'
```

グローバルオプション

knitr::opts_chunk\$set()で設定。例えば...

```
```{r include = FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```

重要なチャンクオプション

- cache** - knitする時に結果をキャッシュするかの設定 (デフォルト=FALSE)
 - cache.path** - キャッシュを保存するディレクトリ (デフォルト="cache/")
 - child** - knit時にインクルードするファイル (デフォルト=NULL)
 - collapse** - 全出力を1つのブロックに折りたたむ (デフォルト=FALSE)
 - comment** - 結果の各行のプレフィックス (デフォルト="##")
- 上記以外のオプション: R.options, aniopts, autodep, background, cache.comments, cache.lazy, cache.rebuild, cache.vars, dev, dev.args, dpi, engine.opts, engine.path, fig.asp, fig.env, fig.ext, fig.keep, fig.lp, fig.path, fig.pos, fig.process, fig.retina, fig.scap, fig.show, fig.showtext, fig.subcap, interval, out.extra, out.height, out.width, prompt,

- dependson** - キャッシュ時のチャンクの依存関係 (デフォルト=NULL)
- echo** - 出力ドキュメントにコードを表示する (デフォルト=TRUE)
- engine** - チャンク内でのデフォルトのコード言語 (デフォルト="R")
- error** - ドキュメント内にエラーメッセージを表示させる(TRUE)か、エラー発生時にレンダリング処理を中止する(FALSE) (デフォルト=FALSE)
- eval** - チャンク内でコードを実行する (デフォルト=TRUE)

- fig.align** - 'left', 'right', 'center'で並べ方を指定 (デフォルト='default')
- fig.cap** - 文字のキャプション (デフォルト=NULL)
- fig.height, fig.width** - プロットの縦横をインチで指定
- highlight** - ソースコードのシンタックスハイライトの設定 (デフォルト=TRUE)
- include** - 実行後チャンクをドキュメントに含めるかの設定 (デフォルト=TRUE)

- message** - message()関数で出力される文字を表示するかの設定 (デフォルト=TRUE)
- results** (デフォルト='markup')
 - 'asis' - 結果をそのまま
 - 'hide' - 結果は表示しない
 - 'hold' - 全コードの後に全結果を表示
- tidy** - 表示用にコードを整える (デフォルト=FALSE)
- warning** - 警告メッセージをドキュメントに含めるかの設定(デフォルト=TRUE)

パラメーター

データや値などを変えて再利用できるよう、ドキュメントをパラメータ化(例: データセット、値等)

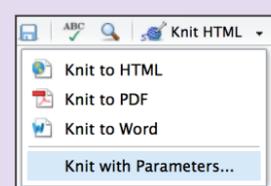
- 1 パラメータを追加
ヘッダ部分で定義できるparamsの一部としてパラメータを作成・設定
- 2 パラメータの呼び出し
ヘッダで設定した値は、params\$<name>というようにして値を呼び出します
- 3 パラメータの設定
Knit with Parameters...から値を設定するか、render()の引数で設定する

```

---
params:
  n: 100
  d: !r Sys.Date()
---
  
```

インデントはスペース2つ

Today's date is 'r params\$d'



render("doc.Rmd", params=list(n=1, d=as.Date("2015-01-01")))

.Rmdの構造

左側にあるようなシンタックスで記述すると、右側のように出力されます。
(レンダリング後)

```
Plain text
End a line with two spaces
to start a new paragraph.
*italics* and **bold**
`verbatim code`
sub/superscript2
strikethrough
escaped: \* \_ \\\
~~strikethrough~~
escaped: \* \_ \\\
endash: --, emdash: ---
equation: $A = \pi * r^2$
equation block:
$$E = mc^2$$
> block quote
# Header1 {#anchor}
## Header 2 {#css_id}
### Header 3 {.#css_class}
#### Header 4
##### Header 5
##### Header 6
<!--Text comment-->
\textbf{Tex ignored in HTML}
<em>HTML ignored in pdfs</em>
<http://www.rstudio.com>
[link](www.rstudio.com)
Jump to [Header 1](#anchor)
image:

* unordered list
+ sub-item 1
+ sub-item 2
- sub-sub-item 1
* item 2
Continued (indent 4 spaces)
1. ordered list
2. item 2
i) sub-item 1
A. sub-sub-item 1
(©) A list whose numbering
continues after
(©) an interruption
Term 1
: Definition 1
| Right | Left | Default | Center |
|:-----:|:-----:|:-----:|:-----:|
| 12 | 12 | 12 | 12 |
| 123 | 123 | 123 | 123 |
| 1 | 1 | 1 | 1 |
- slide bullet 1
- slide bullet 2
(>- to have bullets appear on click)
horizontal rule/slide break:
***
A footnote [^1]
[^1]: Here is the footnote.
```

```
Plain text
End a line with two spaces
to start a new paragraph.
italics and bold
verbatim code
sub/superscript2
strikethrough
escaped: \* \_ \\\
endash: --, emdash: ---
equation: A = \pi * r^2
equation block:
E = mc^2
```

block quote

Header 1

Header 2

Header 3

Header 4

Header 5

Header 6

HTML ignored in pdfs

<http://www.rstudio.com>

Jump to [Header 1](#)

image:



Caption

- unordered list
 - sub-item 1
 - sub-item 2
 - sub-sub-item 1
- item 2
 - Continued (indent 4 spaces)

1. ordered list
2. item 2
 - i. sub-item 1
 - A. sub-sub-item 1

- continues after
1. A list whose numbering continues after
 2. an interruption

Term 1

Definition 1

| Right | Left | Default | Center |
|-------|------|---------|--------|
| 12 | 12 | 12 | 12 |
| 123 | 123 | 123 | 123 |
| 1 | 1 | 1 | 1 |

- slide bullet 1
- slide bullet 2

horizontal rule/slide break:

A footnote ¹

1. Here is the footnote. →

YAMLを用いたレンダーオプションの設定

| サブオプション | 説明 | html | pdf | word | odt | rtf | md | github | ioslides | slidy | beamer |
|------------------------------|--|------|-----|------|-----|-----|----|--------|----------|-------|--------|
| citation_package | 引用を管理するためのLaTeXパッケージの指定 (natbib, biblatex, none) | | X | | | | X | | | | X |
| code_folding | Rコードの折りたたみ表示を設定 ("none", "hide", "show") | X | | | | | | | | | |
| colortheme | Beamerのカラーテーマを設定 | | | | | | | | | | X |
| css | ドキュメントの体裁を定義するCSSファイルの指定 | X | | | | | | | X | X | |
| dev | プロットの出力に使うグラフィックデバイスの指定 (例: "png") | X | X | | | | X | X | X | X | X |
| duration | スライドのフッターにカウントダウンタイマーを設置 (分単位) | | | | | | | | | | X |
| fig_caption | キャプション付きで図をレンダリングする設定 | X | X | X | X | | | | X | X | X |
| fig_height, fig_width | 図のデフォルトの高さ/幅を設定 (インチ単位) | X | X | X | X | X | X | X | X | X | X |
| highlight | シンタックスハイライトの設定 ("tango", "pygments", "kate", "zenburn", "texmate") | X | X | X | | | | | | | X |
| includes | ドキュメントに取り込むファイルの指定 (in_header, before_body, after_body) | X | X | | X | | X | X | X | X | X |
| incremental | 箇条書きを一度に表示するか、クリック毎に表示するかの設定 | | | | | | | | X | X | X |
| keep_md | knitrの処理で出力される.mdファイルを残す | X | | X | X | X | | | X | X | |
| keep_tex | knitrの処理で出力される.texファイルを残す | | X | | | | | | | | X |
| latex_engine | LaTeXをレンダーするためのエンジンを指定 ("pdflatex", "xelatex", "lualatex") | X | | | | | | | | | X |
| lib_dir | 依存関係にあるファイルがあるディレクトリの指定 (Bootstrap, MathJax等) | X | | | | | | | X | X | |
| mathjax | MathJaxレンダーの場所を指定(ローカル/リモートのURL両方指定可能) | X | | | | | | | X | X | |
| md_extensions | R Markdownへ追加/削除する拡張を定義 | X | X | X | X | X | X | X | X | X | X |
| number_sections | ヘッダに章番号を追加 | X | X | | | | | | | | |
| pandoc_args | Pandocへ渡すオプション引数の設定 | X | X | X | X | X | X | X | X | X | X |
| preserve_yaml | 最終的なドキュメントにYAMLフロントマターを残す | | | | | | | | X | | |
| reference_docx | docxを出力する時に形式をコピーするための元ファイル | | | X | | | | | | | |
| self_contained | ドキュメントに依存するものを埋め込む | X | | | | | | | | X | X |
| slide_level | 1ページに分割する見出しの最小レベルを設定 | | | | | | | | | | X |
| smaller | フォントの大きさが小さいものを使用する | | | | | | | | | | X |
| smart | クオートをカーリーにする、ダッシュをemダッシュにする等の処理を行う | X | | | | | | | | | X |
| template | レンダリングに使うPandocテンプレート | X | X | | X | | | | | | X |
| theme | BootswatchとBeamerで使うテーマ | X | | | | | | | | | X |
| toc | ドキュメントの始めに見出しを追加 | X | X | X | | X | X | X | | | X |
| toc_depth | 目次へ含める見出しのレベルを設定 | X | X | X | | X | X | X | | | |
| toc_float | メインコンテンツの右上に見出しのフロートテーブルを追加する | X | | | | | | | | | |

上記以外のオプション: extra_dependencies, fig_crop, fig_retina, font_adjustment, font_theme, footer, logo, html_preview, reference_odt, transition, variant, widescreen

- レンダリング時、R Markdownは次の様に処理されます。
1. Rコードを実行し、knitrで結果とテキストを.mdファイルへ埋め込む
 2. 次に、pandoを用いて.mdを最終的なフォーマットへ変換する



ドキュメントのデフォルトフォーマットはYAMLヘッダで次の様に設定する

```
---
output: html_document
---
```

| output設定する値 | 出力ファイル |
|------------------------------|------------------------|
| html_document | html |
| pdf_document | pdf (要TeX) |
| word_document | Microsoft Word (.docx) |
| odt_document | OpenDocument Text |
| rtf_document | リッチテキストフォーマット |
| md_document | マークダウン |
| github_document | github式マークダウン |
| ioslides_presentation | ioslides HTMLスライド |
| slidy_presentation | slidy HTMLスライド |
| beamer_presentation | beamer pdfスライド (要TeX) |

サブオプション(右の一覧)で出力をカスタマイズ

```
---
output:
  html_document:
    code_floating: hide
    toc_float: TRUE
---
```

htmlタブセット
サブヘッダをタブにするにはcssの.tabsetクラスを使います

```
# Tabset {.tabset .tabset-fade .tabset-pills}
## Tab 1
text 1

## Tab2
text 2

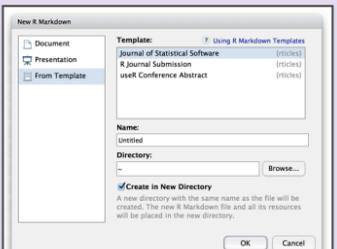
### End Tabset
```

End tabset

再利用可能なテンプレート作成

1. 新しいパッケージの作成
パッケージをinst/rmarkdown/templatesへ置く
2. ディレクトリ内で次のファイルを作成する
 - **template.yaml** (下記参照)
 - **skelton.Rmd** (テンプレート)
 - 必要なその他ファイル
3. パッケージをインストール
4. File ▶ New File ▶ R Markdownと辿るとテンプレートが出来上がっています

```
template.yaml
name: My Template
```



テーブル

Rのデータをテーブル化するいくつかの方法

| eruptions | waiting |
|-----------|---------|
| 3.600 | 79 |
| 1.800 | 54 |
| 3.333 | 74 |
| 2.283 | 62 |

| eruptions | waiting |
|-----------|------------|
| 1 | 3.60 79.00 |
| 2 | 1.80 54.00 |
| 3 | 3.33 74.00 |
| 4 | 2.28 62.00 |

| eruptions | waiting |
|-----------|----------|
| 1 | 3.600 79 |
| 2 | 1.800 54 |
| 3 | 3.333 74 |
| 4 | 2.283 62 |

```
data <- faithful[1:4, ]
```{r results = "asis"}
knitr::kable(data, caption = "kableでテーブル化")
```

```{r results = "asis"}
print(xtable::xtable(data, caption = "xtableでテーブル化"),
 type = "html", html.table.attributes = "border=0")
```

```{r results = "asis"}
stargazer::stargazer(data, type = "html",
 title = "stargazerでテーブル化")
```
```

ここでの例以外にも **stargazer**, **xtable** と **knitr** の使い方があり ます

参考・引用文献

- 参考・引用文献の管理は次の形式が対応しています
.bib, .bibtex, .copac, .enl, .json, .medline, .mods, .ris, .wos, .xml
1. 参考文献ファイルの設定
YAMLヘッダに記述します。必要であればCSL 1.0スタイルファイルも設定します。
- ```

bibliography: refs.bib
csl: style.csl

```
2. 引用キーをテキスト内で使う  
Smith cited [smith04].  
Smith cited without author [-smith04].  
[smith04] cited in line.
  3. レンダー  
参考文献はドキュメントの最後に追加されます。
- ```
Smith cited (Joe Smith 2004).
Smith cited without author (2004).
Joe Smith (2004) cited in line.
```