

R Markdown : : CHEAT SHEET



R Markdownとは?



.Rmd ファイル・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。

Reproducible Research・ボタンをクリックするかコマンドを入力するとR Markdownファイル内での処理が再実行されたり、完成したレポートとしてエクスポートされたりします。

Dynamic Documents・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。

R Markdown

```

1 ---
2 title: "R Markdown" プレビュー コードチャンクの挿入
3 author: "RStudio" 先の設定
4 output: 公開
5   html_document:
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>.

```

> library(rmarkdown)
> render("report.Rmd", output_file = "report.html")

```

.rmdの構成

YAMLヘッダー
レンダラーのオプション部分(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、---に挟まれた行の間に記述します

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。

パラメータ

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)

- 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
- パラメータの呼び出し** ヘッダで設定した値は、params\$<name>という様に値を呼び出します
- 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

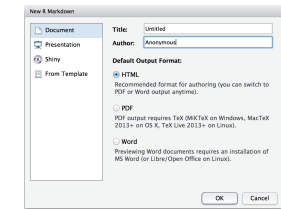
```

---
params:
  n: 100
  d: !r Sys.Date()
---
Today's date is `r params$d`

```

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

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ことができます
- 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
- 6 **ビルドログの確認** R Markdownコンソール内で出来ます
- 7 **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

レンダラー

rmarkdown::render() to render/knit at cmd line. Important args:

input - レンダリングするファイル	output_options - 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 - インプットファイルのエンコード
-----------------------------	---	---------------------------------	------------------------------	-------------------------------	-----------------------------------

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 getRversion()` → 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のアイコンでも挿入できます。

```

```{r echo=TRUE}
getRversion()
```

```

**グローバルオプション**  
knitr::opts\_chunk\$set()で設定。具体例は次の通り。

```

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

```

## 重要なチャンク

- 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="##")
- 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)

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,

## 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. インプ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RStudio IDEのRun Documentボタンをクリックしてレンダラーする

```

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

NOTE: レポートはShinyアプリとしてレンダラーされます。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セッションで実行されます。





Pandocマークダウン

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: * _ \
endash: --, emdash: ---
equation: $A = \pi * r^2$
equation block:

$$E = mc^2$$


```

block quote

Header1

Header 2

Header 3

Header 4


Header 5

Header 6

HTML ignored in pdfs

<http://www.rstudio.com>

Jump to [Header 1](#)



Caption

- unordered list
 - sub-item 1
 - sub-item 2
- item 2
 - sub-sub-item 1

Continued (indent 4 spaces)

- item 2
 - Continued (indent 4 spaces)
- ordered list
 - sub-item 1
 - A. sub-sub-item 1

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.

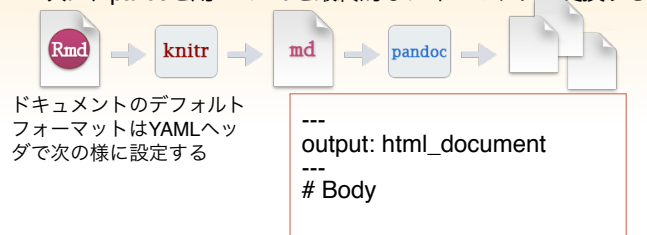
A footnote 1

- Here is the footnote.

YAMLを用いたレンダーオプションの設定

レンダリング時、R Markdownは次の様に処理されます。

- 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
- 次に、pandoを用いて.mdを最終的なフォーマットへ変換する



output設定値	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	Rich Text Format
md_document	Markdown
github_document	Github互換マークダウン
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_folding: hide
    toc_float: TRUE
# Body
  
```

スペース2つ分のインデント

スペース4つ分のインデント

html タブ

サブヘッダをタブにするにはcssの.tabsetクラスを使います

```

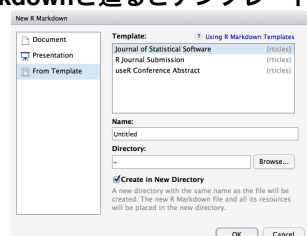
# Tabset {.tabset .tabset-fade .tabset-pills}
## Tab 1
text 1
## Tab 2
text 2
### End tabset
  
```

再利用可能なテンプレート作成

- 新しいパッケージの作成 パッケージをinst/rmarkdown/templatesへ置く
- 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
- パッケージをインストール
- 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
---
  
```



sub-option	description	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
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)	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
slide_level	ドキュメントに依存するものを埋め込む										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									

テーブル

Rのデータをテーブル化するいくつかの方法

```

data <- faithful[1:4, ]
knitr::kable(data, caption = "Table with kable")
---
{r results = "asis"}
print(xtable::xtable(data, caption = "Table with xtable"),
      type = "html", html.table.attributes = "border=0")
---
{r results = "asis"}
stargazer::stargazer(data, type = "html", title = "Table with stargazer")
---
  
```

ここでの例以外に stargazer, xtable, knitr等があります

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

- 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CSL1.0スタイルファイルを設定します
- 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.
- 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。

```

---
bibliography: refs.bib
cs1: style.csl
---
  
```

```

Smith cited (Joe Smith 2004).
Smith cited without author (2004).
Joe Smith (2004) cited in line.
  
```

