

RMarkdown を使って日本語の文書を作成する

福田俊介

2019 年 11 月 14 日

目次

1	概要	2
2	PDF 出力に必要なもののインストール	2
2.1	TeX のインストール	2
2.2	IPAex フォントのインストール	3
3	日本語表示用の R Markdown ファイルのテンプレート	4
4	あとがき	5
4.1	カスタマイズを行う場所	5
4.2	その他	6

1 概要

RStudio では、R Markdown 書式の文書を kniter で処理することにより、簡単に HTML 等の体裁の整った文書に変換することが可能ですが、PDF 文書やこれに日本語の文字を含んだ文書を作成する場合には、追加の設定を上手く行わないと、PDF 文書の作成を行うことが出来ません。そこで、まず、R Markdown 書式から日本語の PDF 文書を作成するために必要となる追加のインストールすべきものの紹介と、日本語利用のための設定テンプレートを紹介します。

2 PDF 出力に必要なもののインストール

本来、R のシステム上で R Markdown を利用する場合には、rmakrdown パッケージをはじめ、それが内部で呼び出す、pandoc と呼ばれるプログラム等のインストールも必要となりますが、RStudio を使う場合、RStudio インストール時にそれらの必須プログラムと一緒にインストールされているので、rmarkdown や pandoc のインストールは必要ありません。

しかし、それ以外で、R Markdown 形式の文書を PDF 出力に変換する場合に必須となるのが、次のインストールです。

- $\text{T}_{\text{E}}\text{X}$ のインストール
- IPAex フォントのインストール

2.1 $\text{T}_{\text{E}}\text{X}$ のインストール

次に、 $\text{T}_{\text{E}}\text{X}$ のシステムをパソコンにインストールします。

$\text{T}_{\text{E}}\text{X}$ に関する日本語の情報は $\text{T}_{\text{E}}\text{X}$ Wiki がお勧めです。

- <https://texwiki.texjp.org/>

この Wiki に目を通して興味を持った人は、このページに紹介されている $\text{T}_{\text{E}}\text{X}$ のインストールを行ってもらって構いません。但し、先に述べた通り、インストールに数時間を要するので時間に余裕がある時に行いましょう。

しかしここでは、「Tiny $\text{T}_{\text{E}}\text{X}$ 」という $\text{T}_{\text{E}}\text{X}$ ディストリビューションを紹介します。これを使うと、RStudio 上から、簡単に $\text{T}_{\text{E}}\text{X}$ のインストールが行えます。また、インストール時間も格段に短く済みます。

2.1.1 Tiny $\text{T}_{\text{E}}\text{X}$

Tiny $\text{T}_{\text{E}}\text{X}$ は R のパッケージなので、まずはパッケージをインストールします。

```
install.packages("tinytex")
```

パッケージのインストールが完了したら、コンソールから以下のコマンドを入力して実行します。このコマンドによって、 $\text{T}_{\text{E}}\text{X}$ システムがパソコンにインストールされます。

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

このインストール中に、幾つかのエラーダイアログ（dll ファイルの認識に再起動を促すメッセージ）がでますが、OK を選択して、進めて問題ありません。

2.2 IPAex フォントのインストール

IPAex フォントについては、以下を参照してください。

- <https://ipafont.ipa.go.jp/old/ipaexfont/download.html>

T_EX 環境で使われる、フォントである IPA フォントをインストールします。tinytex パッケージから簡単にインストールできるので、以下のコマンドで、IPAex フォントをインストールしましょう。

```
tinytex::tlmgr_install("ipaex")
```

3 日本語表示用の R Markdown ファイルのテンプレート

以下に rmarkdown ファイルのテンプレートを紹介します。

```
---
title: "日本語用テンプレート"
author: ""
date: "2019年11月14日"
output:
  pdf_document:
    latex_engine: xelatex
    number_sections: true
    toc: true
    toc_depth: 2
documentclass: bxjsarticle
classoption: xelatex,ja=standard
indent: TRUE
geometry: no
---

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

if (knitr::opts_knit$get("rmarkdown.pandoc.to") %in% c("beamer", "latex")) {

 options(device = function(file, width = 7, height = 7, ...) {
 cairo_pdf(tempfile(), width = width, height = height, ...)
 })

 knitr::opts_chunk$set(dev="cairo_pdf", dev.args=list(family="ipa"))
}

```

この内容を `template.rmd` ファイルとして保存しておきます。rmd ファイルを作成するときには、準備したテンプレートファイルをコピーして利用することでいつでも日本語の入った markdown から PDF ファイルを作成することが可能になります。

## 4 あとがき

ここで紹介している RStudio での文書作成は、幾つかのツールを組み合わせで少し複雑な処理が行われています。

まず、**R Markdown** というのは、**Markdown** という文書記述の形式に加えて、文書中に記述されたコードを見やすく表示すると共に、その R コードを実行した計算結果やグラフを文書に埋め込むことも出来るように拡張したもののなのです。そこで、この R のコードが埋め込まれた部分について **knitter** というプログラムが処理することで、その部分をただの Markdown 書式の文書に変換することができます。そして、処理されて出来上がった Markdown 文書は、**pandoc** というプログラムによって、html や pdf の形式に変換することが出来ます。

### 4.1 カスタマイズを行う場所

このような流れにある rmarkdown による文書作成のカスタマイズについては、上記のそれぞれの段階で処理されるときに一緒に渡される引数を設定することで行います。

ここで、大まかに分けて、rmarkdown は、R の処理系から pandoc を呼び出すための入口の役目を果たしています。そこで、結局は処理をするおおもとの pandoc 側と同じ設定事項を設定して、それを pandoc に引き渡しているものが多いです。一方で、knitter は、文書内の R チャンクをどのように処理して出力するかの設定です。

#### 4.1.1 R Markdown と pandoc

ファイル先頭の「`---`」で囲まれた部分を、yaml フロントマターと呼びますが、R Markdown や pandoc にかかわる設定はこの部分で行います。

```

title: " 日本語用テンプレート"
author: " 福田俊介"
date: "2019 年 11 月 14 日"
output:
 pdf_document:
 latex_engine: xelatex
 number_sections: true
 toc: true
 toc_depth: 2
documentclass: bxjsarticle
classoption: xelatex,ja=standard
indent: TRUE
geometry: no

```

この部分の解説については、まず、R Markdown: The Definitive Guide において R Markdown の開発に関わった人が書いている文書が公開されています。この文書のうち、pdf ドキュメントに関しては特に次の項目にリファレンスがあります。

- <https://bookdown.org/yihui/rmarkdown/pdf-document.html>

この文書においてメタデータの変数等についての詳細は、直接 pandoc のマニュアルを見るように書かれています。pandoc のマニュアルは以下で公開されています。

- Pandoc User's Guide(<https://pandoc.org/MANUAL.html>)

当該マニュアルの中の、「Options」項目にある「Metadata variables」辺りをみると yml の中で設定できる変数名とその内容についての詳細を知ることが出来ます。一方、rmarkdown 側も、rmarkdown パッケージのリファレンスをみると、pdf\_document() 関数のリファレンスにその引数についての詳細が記されており、これが yml での設定項目とリンクしています。rmarkdown のリファレンスは以下にあります。

- <https://rmarkdown.rstudio.com/docs/index.html>
- [https://rmarkdown.rstudio.com/docs/reference/pdf\\_document.html](https://rmarkdown.rstudio.com/docs/reference/pdf_document.html)

#### 4.1.2 knitr

rmarkdown 内のチャンクの書き方そのものについては、先に紹介した R Markdown: The Definitive Guide にも記載されています。

しかし、より詳細については、以下に示す、knitr 開発者が記している文書が参考になります。

- <https://yihui.org/knitr/>

特に、Option 項目にある Plots の部分が、グラフを表示する時のグラフィックデバイスの扱いについての設定項目で、日本語等の文字を扱う場合に、設定が必要な部分になります。

## 4.2 その他

pdf 出力文書を行う場合、元の rmarkdown 文書に、 $\text{\TeX}$  の書式を埋め込むことが出来ます。このため、markdown を pandoc で  $\text{\TeX}$  に変換するだけでは、表現できないと思う場合、その部分は、 $\text{\TeX}$  で記述することが出来ます。(pandoc で色々な形式に出力できるという利便性は失われる)

そして、 $\text{\TeX}$  の知識を十分に得たならば、pandoc が PDF 出力する時に利用するデフォルトのテンプレートをカスタマイズすることで、文書の体裁をカスタマイズすることも可能となります。