

R Markdown で日本語 beamer プレゼンテーション (LuaLaTeX) 版

ill-identified

2020-09-08

Нужны новые формы. Новые формы нужны, а если их нет, то лучше ничего не нужно.

新しいフォーマットが必要なんですよ. 新しいフォーマットが. それがないというなら, いっそ何もないほうがいい.

— A. チェーホフ 『かもめ』

イントロダクション

使い方 / 用例

用例: 図表の挿入

外部資料の引用方法

基本的なカスタマイズ

トラブルシューティング

まとめ

補足: 細かい技術的な話

イントロダクション

このスライドは何?

- ・ あまり情報が流れていない, R Markdown と beamer で日本語を含むスライドを作るためのテンプレート兼用例集
- ・ reveal.js など HTML 媒体は他の資料を参照
 - ・ [ここ](#)や[ここ](#)を見よ
- ・ もともとは自分用に作ったテンプレだったものを万人向けに修正

想定される用途

- ・ Tokyo.R など R を使った話を発表する際の資料作成
- ・ 技術・アカデミック寄りの話題を想定
- ・ 具体的に要求されるもの
 - ・ 日本語表示
 - ・ ラスタまたはベクタ画像の挿入
 - ・ 表の挿入
 - ・ R コードを見やすく表示
 - ・ 参考文献の相互参照 / リスト自動生成
 - ・ LyX や overleaf より簡単であること
 - ・ なんかナウでオサレな感じは求めてない
 - ・ 自由すぎるデザインは不可

先行事例の紹介

- ・ 伊東『R Markdown と Beamer でプレゼンテーション資料作成』
 - ・ Lua \LaTeX を使って日本語で Beamer スライド作成
- ・ Atusy『R Markdown + XeLaTeX で日本語含め好きなフォントを使って PDF を出力する』
- ・ 先行事例との違い:
 - ・ 使用者が設定を書く負担削減のためテンプレート化
 - ・ フォントやテーマなどデザインに関する大まかな選択の余地
 - ・ X₃ \LaTeX / Lua \LaTeX 両方に対応
 - ・ 日本語文献 bib ファイル・bst ファイルに対応
 - ・ 充実したスライド作例

reveal.js じゃダメなの?

- ・ 個人的にデザインとかあまり好きじゃない
- ・ 上下左右に動いて空間識失調になる
 - ・ (個人の体験です)
 - ・ 上下のみにもできる
- ・ html よりも不変な媒体にしたい
 - ・ pdf が明確に優れているかは怪しい
- ・ ~~Q: お前が使いこなせてないだけじゃないの?~~
 - ・ ~~A: うるさい~~

パワーポイントじゃダメなの？

- ・ 私は持っていない
- ・ シンタックスハイライトが面倒
 - ・ 注: パワポの場合は VSCode か `reprex` でコピペ
- ・ ドラッグ&ドロップで位置調整は便利
- ・ しかしポンチ絵芸術になりがち
- ・ 極力シンプルにして視線誘導の負担をなくすべき
 - ・ 徹底するかは好みの問題

使い方 / 用例

セットアップ

1. パッケージのインストール

```
remotes::install_github(  
  "Gedevan-Aleksizde/my_latex_templates",  
  subdir = "rmdja")
```

2. TeXLive (>= 2020) のインストール

- ・ 分からなければ [TeX wiki](#) のページを参考に
- ・ Debian/Ubuntu のユーザは apt より公式ダウンローダの方が良いかも

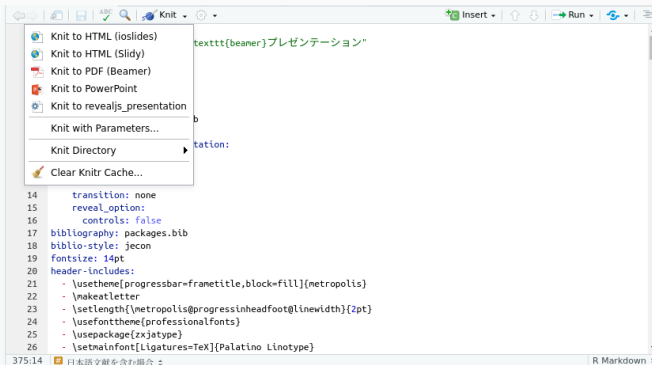
基本 (1/2)

1. yaml ヘッダに以下を書く

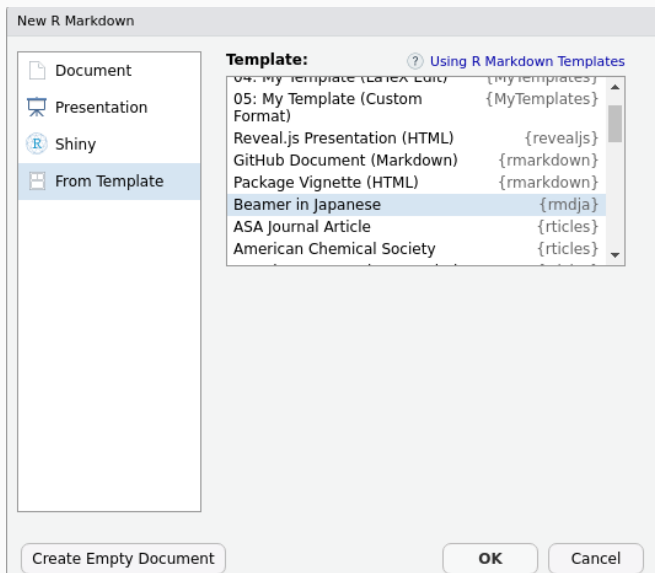
- ・ 詳しくは example/ のテンプレートを参考に

output: rmdja::beamer_presentation_ja

2. RStudio のツールバーの “Knit” を押す



- ・ または テンプレートから選択



最低限必要な設定

- ・ フォントが OS ごとに異なるのが問題
- ・ 以下は Linux / Win / Mac のいずれでも動く

```
output:
```

```
  rmdja::beamer_presentation_ja:
```

```
    latex_engine: lualatex
```

```
mainfont: "DejaVu Serif"
```

```
mainfontoptions:
```

```
sansfont: "DejaVu Sans"
```

```
monofont: 'Courier New'
```

基本構文 (1/2)

- ・ markdown 的な書き方でできる
- ・ “## タイトル” でスライドの開始

節見出し

タイトル 1

- ****太字**** `**bold**`
- 強調 `_emph_`
- **タイプライタ体** ``mono``
- ~~取り消し線~~

- ・ 太字 `bold`
- ・ 強調 `emph`
- ・ タイプライタ体 `mono`
- ・ 取り消し線

- ・ \LaTeX コマンドも挿入可能
 - ・ うまく行かない場合は $\backslash\ldots\{\text{=latex}\}$ で囲む
- ・ \LaTeX 使用例: \LaTeX で文書にルビも打てる
- ・ ルビ表示は `pxrubrica` の構文を参考に

ソースコードの表示

- ・ 以下で対応言語一覧がわかる

```
names(knitr::knit_engines$get())
```

```
[1] "awk"          "bash"          "coffee"        "gawk"           "gr"
[7] "lein"         "mysql"         "node"          "octave"         "perl"
[13] "Rscript"      "ruby"          "sas"           "scala"          "se"
[19] "stata"        "zsh"           "highlight"     "Rcpp"           "tex"
[25] "c"            "cc"            "fortran"       "fortran95"     "as"
[31] "asis"         "stan"          "block"         "block2"         "js"
[37] "sql"          "go"            "python"        "julia"          "sa"
```

- ・ 伊東 『R Markdown と Beamer でプレゼンテーション資料作成』 (Lua \LaTeX 使用)
- ・ 松田 『Beamer 読本-講演用スライド作成のために-』
- ・ Kazutan 『R Markdown によるスライド生成』 『R Markdown 入門』
- ・ Atusy 『R Markdown + XeLaTeX で日本語含め好きなフォントを使って PDF を出力する』
- ・ R Markdown 2.0 チートシートの日本語訳, Takahashi, M. 訳

もう少しくわしいやつ

- Atusy『R Markdown **ユーザーのための** Pandoc's Markdown』
- 謝益輝 (yihui) “knitr - Elegant, flexible, and fast dynamic report generation with R” (開発者本人)
- Xie, Yihui & C. Dervieux “R Markdown Cookbook”

今回使うパッケージ

- ・ この用例作成には以下パッケージを使用

```
01 require(conflicted)      # パッケージの競合防止用
02 require(tidyverse)       # 全般
03 require(ggthemes)        # ggplot2 のデザイン変更
04 require(ggdag)           # ネットワーク図の用例に
05 require(DiagrammeR)      # DOT 言語でのネットワーク図例
06 require(kableExtra)      # 表の出力オプション
07 require(stargazer)       # 複雑な LaTeX の表を扱う例
```

- ・ 以下はインストールのみ / 読み込む必要なし

- ・ citr: 引用文献の挿入を GUI で
- ・ bookdown: 数式を GUI で

ソースコードの表示: 基本事項

- ・ `echo=T` でチャンク内コードを表示
 - ・ デフォでは非表示
 - ・ 自動でシンタックスハイライト
- ・ はみ出す場合は `tidy=F` して手動改行
 - ・ 日本語等で折り返し地点がうまく行かない
- ・ `class.source = "numberLines, LineAnchors"` で行番号表示 (参考)

ソースコードの表示: 出力例

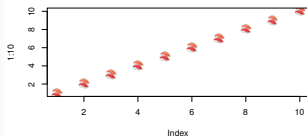
```
```{r, echo=T, class.source = "numberLines, LineAnchors"}  
require(conflicted)
require(tidyverse)
require(ggthemes)
```
```

```
01 require(conflicted)  
02 require(tidyverse)  
03 require(ggthemes)
```

カラー絵文字

- ・ `BXcoloremoji` をインストール
 - ・ `\coloremoji{}` で絵文字表示: 🍣
 - ・ 実際には画像に置き換えている
- ・ グラフ描画には特に設定必要なし
 - ・ ソースコード上のものは文字化けする

```
plot(1:10, pch = "?")
```



数式の挿入: 行内(インライン)

- markdown 風の LaTeX コード埋め込み
- \LaTeX の数式を\$で挟む
- 例: らんま $\frac{1}{2}$
 - 出力: らんま $\frac{1}{2}$
 - 注: 行内で分数はスラッシュ使ったほうが見やすい
- 数式にはセリフフォント使用
 - スライドはサンセリフが良いとされる
 - しかし数式の統一感がない
 - (個人の好み?)

数式の挿入: 独立行

- ・ \$\$で挟んだ範囲に \LaTeX 構文

```
$$\begin{aligned}
& \sin^2(x) + \cos^2(x) = 1\\
& f(x) = \frac{1}{(2\pi)^2} \int_{\mathbb{R}^n} \hat{f}(\omega) \exp(i\omega x) d\omega
\end{aligned}$$
```

$$\sin^2(x) + \cos^2(x) = 1$$

$$f(x) = \frac{1}{(2\pi)^2} \int_{\mathbb{R}^n} \hat{f}(\omega) \exp(i\omega x) d\omega$$

数式の挿入: bookdown パッケージのアドインで補完

1. RStudio のツールバー “Addins”
2. “Input LaTeX Math”

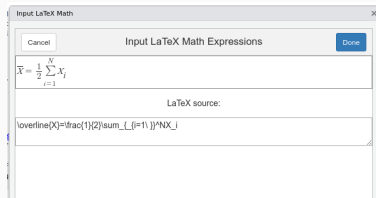


図 1: bookdown の数式入力機能

- ・ 一部対応してない記号もある?
 - ・ `\mathbb{}`とか `\hat{}`とか
- ・ 数式のみで `\aligned` 等環境の入力は不可

用例: 図表の挿入

図の挿入: 画像ファイル貼り付け

- ・ `out.width=/out.height=` でサイズ調整
- ・ jpeg, png, eps, pdf に対応
 - ・ \LaTeX の制約
- ・ デフォルトでは縦に並べる
 - ・ 横並びにしたい場合は `fig.show="hold"`

```
knitr::include_graphics(file.path(file_loc,  
  c("img/tiger.eps", "img/tiger.pdf", "img/tiger.png")))
```



図 2: いつもの虎 (TeXLive より)

図の挿入: markdown 構文で貼り付け

- ・ `out.width=/out.height=`が適用されない
- ・ pandoc 構文でサイズ指定

```
![The Tiger](img/tiger.pdf){ height=30% }
```



図 3: The Tiger

図の挿入: ggplot2 のグラフ

- ・ `fig.cap=`でキャプションを設定可能. `labs(title =)`と違い自動相互参照あり

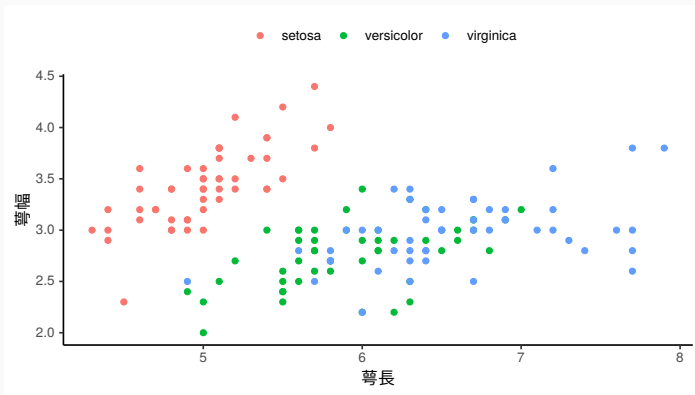


図 4: ggplot2 の出力例: iris データ

図の挿入: 文字の大きさをそろえるには

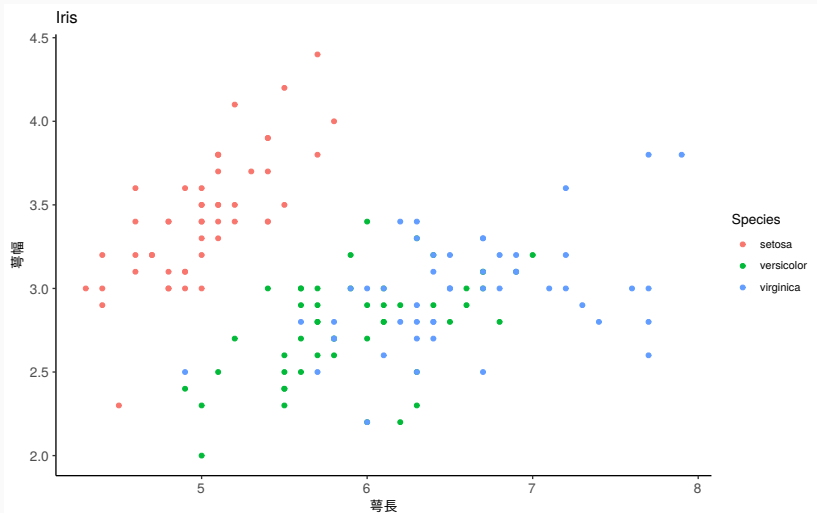
- ・ 出力された画像ファイルの文字が小さい!
- ・ その原因は
 1. 自動縮小されるため
 - ・ 込み入った話なので次のスライドへ
 2. 単位が違うため
 - ・ beamer は主に pt 単位
 - ・ ggplot2 は `aannotate()` のみ mm 単位
 - ・ 補足
 - ・ `cairo_pdf()` の `pointsize` はビルトインデバイスにのみ影響
 - ・ 『ggplot2 の size が意味するもの』

図の挿入: 画像サイズの基本ルール

- ・ R が作図したファイルを一旦保存し, 拡大縮小して貼り付けられる
 - ・ `fig.width/fig.height` は保存時のサイズ
 - ・ `out.width/out.height` は表示するサイズ
- ・ R の保存サイズと beamer スライドのサイズのデフォルトは違う
 - ・ スライドは 5.04 x 3.78 in (128 x 96 mm)(4:3)
 - ・ `ggsave()` は 9.11 x 5.77 in で保存
- ・ RStudio のビューアは文字の大きさ固定でサイズを画面に合わせる
 - ・ 違和感の正体 (?)

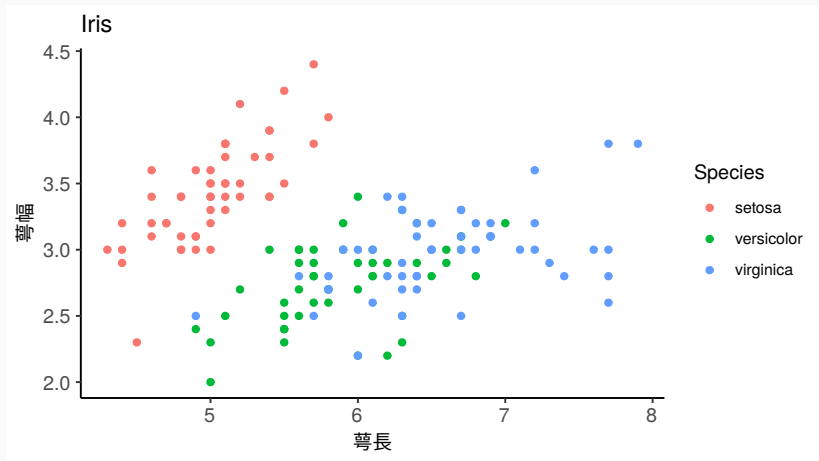
図の挿入: 幅 100%で出力

- ・ 注: `out.width="100%"` はスライドサイズではなく本文領域の相対サイズ



図の挿入: beamer サイズで保存, 幅 100%で出力

- ・ 相対的に文字が大きくなった



図の挿入: 字の大きさをなるべく揃える

- ・ 基準を beamer に合わせる方法
 1. 保存時サイズを beamer の画面サイズと同じにする
 2. `theme_*`() で `base_size` を beamer の文字サイズと同じにする
- ・ `out.width="100%"` のとき, グラフタイトルと本文のサイズが一致
- ・ 拡大縮小に合わせて文字の大きさを調整する
- ・ 横長のグラフなら `fig.width=` を調整する
- ・ ユーザは `theme_*`() の文字サイズのみ手動で書く
 - ・ `theme_set(base_size =)` で統一すると楽

図の挿入: 再現可能なポンチ絵

- ・ 概念図とかの図示はどうするか
 - ・ NOT データの視覚化 (ビジュアライゼーション)
 - ・ ggplot2 の本来の使い方ではない
- ・ ggdag はネットワーク図に使える
 - ・ 因果ダイアグラム, 遷移図, グラフィカルモデル等
- ・ ggforce はベン図の描画に応用可能
 - ・ 世間的にはグラフの部分拡大用パッケージ?
- ・ 詳しくは個別のマニュアル参照
- ・ 霞が関流ポンチ絵は専門外

図の挿入: ポンチ絵の例 1

- ・ 以前作ったやつ²の修正

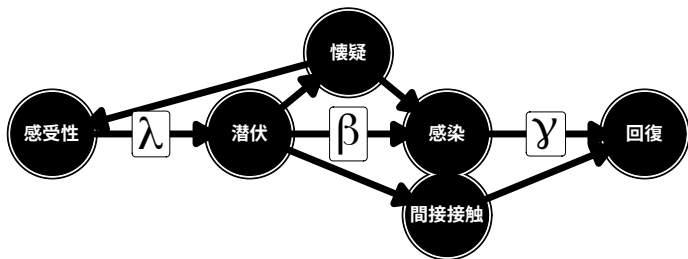


図 5: ggdag で作った YJ-SEIR モデルの遷移図

図の挿入: ポンチ絵の例 2

- ・ `ggforce::geom_circle()` を利用
 - ・ 参考: [How to Plot Venn Diagrams Using R, ggplot2 and ggforce](#)

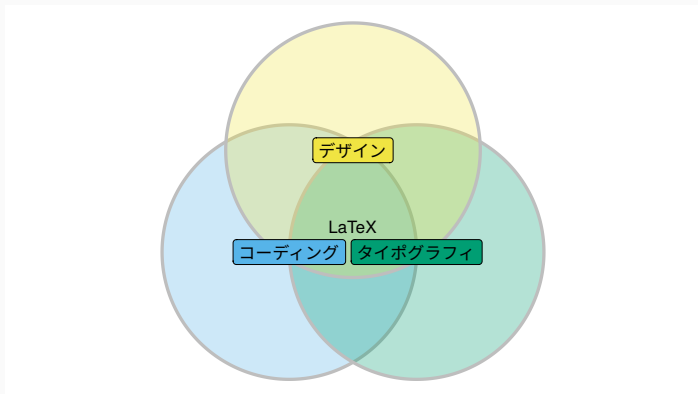


図 6: ベン図の例

図の挿入: DiagrammeR で DOT 言語で書く

- DiagrammeR::grViz() で DOT 言語によるグラフィカルモデル描画
 - 注: fig.show="hold"にすると正しく出力できない

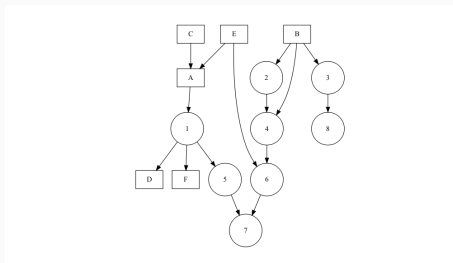


図 7: DiagrammeR による作図

図の挿入: R 以外のデバイス

- \LaTeX の tikz を使用可能
 - tikz を知らない人は[ここ](#)や [TeX Wiki](#) を読む
 - 現時点では日本語表示が面倒 ([参考](#))
 - そこまでやるなら全部 \LaTeX で書いたほうがいいのではないかな?

表の挿入: データフレーム

- ・ デフォルトの表示

```
data(iris)
print(head(iris))
```

| | Sepal.Length | Sepal.Width | Petal.Length | Petal.Width | Species |
|---|--------------|-------------|--------------|-------------|---------|
| 1 | 5.1 | 3.5 | 1.4 | 0.2 | setosa |
| 2 | 4.9 | 3.0 | 1.4 | 0.2 | setosa |
| 3 | 4.7 | 3.2 | 1.3 | 0.2 | setosa |
| 4 | 4.6 | 3.1 | 1.5 | 0.2 | setosa |
| 5 | 5.0 | 3.6 | 1.4 | 0.2 | setosa |
| 6 | 5.4 | 3.9 | 1.7 | 0.4 | setosa |

表の挿入: データフレームを `kable()` で表示

- \LaTeX 風の表になる
 - 詳しくは `knitr::kable()` や `kableExtra` のマニュアル

```
data(iris)
kable(head(iris[, 1:3]),
       caption="kable() による表示")
```

表 1: `kable()` による表示

| Sepal.Length | Sepal.Width | Petal.Length |
|--------------|-------------|--------------|
| 5.1 | 3.5 | 1.4 |
| 4.9 | 3.0 | 1.4 |
| 4.7 | 3.2 | 1.3 |
| 4.6 | 3.1 | 1.5 |
| 5.0 | 3.6 | 1.4 |

表の挿入: 外部の.tex ファイル

- ・ \LaTeX でかかれた表を貼り付けて掲載
 - ・ `\input{tab.tex}` でコピーなしで貼り付け可
 - ・ リサイズは手動で
- ・ 以下, 表を一旦.tex ファイルで出力してから読み込む
 - ・ R 上で生成した TeX コードなら直接出力可 (2 枚後のスライド参照)

表の挿入: .tex で書かれた表を掲載

| | Sepal.Length | Sepal.Width | Petal.Length | Petal.Width | Species |
|---|--------------|-------------|--------------|-------------|---------|
| 1 | 5.10 | 3.50 | 1.40 | 0.20 | setosa |
| 2 | 4.90 | 3.00 | 1.40 | 0.20 | setosa |
| 3 | 4.70 | 3.20 | 1.30 | 0.20 | setosa |
| 4 | 4.60 | 3.10 | 1.50 | 0.20 | setosa |
| 5 | 5.00 | 3.60 | 1.40 | 0.20 | setosa |
| 6 | 5.40 | 3.90 | 1.70 | 0.40 | setosa |

表の挿入: stargazer の表示

- ・ `{r, results="asis"}` で出力 tex コードを直接表示
- ・ stargazer の使い方は[矢内氏の解説](#)や[私のブログ](#)参照

```
fit1 <- lm(Sepal.Length ~ Petal.Width, data = iris)
fit2 <- lm(Sepal.Length ~ Petal.Width + Petal.Length,
           data = iris)
stargazer(fit1, fit2,
          header = F, type = "latex",
          digits = 2, digits.extra = 0, align = T,
          ...)
```

表の挿入: stargazer の出力結果

表 2: 回帰分析の結果

| | モデル 1 | |
|-------------------------|-------------------|-------------------|
| | 萼長 | |
| | (1) | (2) |
| 定数項 | 4.78***
(0.07) | 4.19***
(0.10) |
| 花弁幅 | 0.89***
(0.05) | -0.32**
(0.16) |
| 花弁長 | | 0.54***
(0.07) |
| Observations | 150 | 150 |
| Adjusted R ² | 0.67 | 0.76 |
| F Statistic | 299.17*** | 240.95*** |

表の挿入: markdown 構文

Table: 得点一覧


| クラス | 科目 | 平均 |
|-----|----|--------|
| A | 算数 | \$90\$ |
| B | 算数 | \$95\$ |

表 3: 得点一覧

| クラス | 科目 | 平均 |
|-----|----|----|
| A | 算数 | 90 |
| B | 算数 | 95 |

外部資料の引用方法

ハイパーリンクの挿入

- ・ url は自動でリンク
 - ・ <https://rstudio.com/>
- ・ markdown 方式のリンク
 - ・ [RStudio](https://rstudio.com/)
 - ・ [RStudio](https://rstudio.com/)
- ・ 画像にハイパーリンク  RStudio[®] を貼ることも可

- ・ `[@ref]` で番号引用: `\citep{ref} ([1])` に対応
- ・ `@ref` で著者名引用: `\citet{ref} (hoge hoge et al.)` に対応
- ・ `[@ref1; @ref1]` で連番引用 `[1, 2]`
- ・ 以下引用テスト

`[@R-tidyverse; @R-rmarkdown; @rmarkdown2018; @R-bookdown]`

`[@R-citr; @wickham2016Data; @Okumura2017LaTeX]`

`[3, 1, 6, 5] [2, 4, 7]`

文献引用の補助: 引用子の補完

- ・ 重複・書き間違えの防止
- ・ citr パッケージを使うと楽
 - ・ ツールバーの Addins から選択
 - ・ zotero 連携機能あり

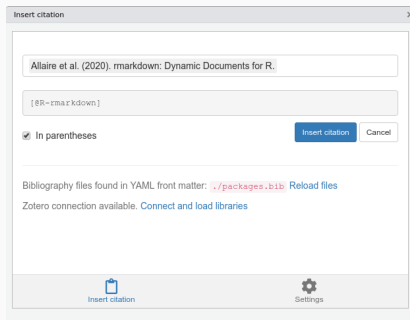


図 8: citr パッケージの GUI

- Mendeley, Zotero, ReabCube の3つが多い?
- 私は Zotero を使っている
 - 多言語対応, 連携機能の充実, 料金などの理由
 - 参考: 『Mendeley Exodus Mendeley から Zotero への移行の手引き~』
- RefManageR パッケージ
 - R で bib ファイルをパースしたりする
 - 文献管理用には既存ソフトで十分?

基本的なカスタマイズ

- ・ 欧文 / 和文それぞれ 3 種類指定できる
- ・ 欧文フォントは `fontspec` で制御
 - ・ `yaml` トップレベルで指定
 - ・ `beamer` なので `main` という名に反してサンセリフが主に使われる
- ・ それぞれに `*options` というオプションパラメータ指定が可能
 - ・ 相対文字サイズの手動調整などに使う

- ・ 設定例

```
mainfont: <欧文フォント>
mainfontoptions:
  - Scale=1.1
  - Ligatures=TeX
sansfont: <欧文サンセリフ体フォント>
monofont: <等幅フォント>
```

- ・ 和文フォントはフォーマット関数の下で指定
- ・ こちらもゴシック (サンセリフ) がメイン
- ・ こちらも *options がある

jmainfont: <和文フォント>

jmainfontoptions: <オプション>

jsansfont: <和文ゴシックフォント>

jmonofont: <和文等幅フォント>

フォント変更: 和文プリセット

- ・ 和文フォント指定はプリセットを使うと楽.
- ・ 対応フォント: Noto, IPA, 原ノ味, ヒラギノ, 游, モリサワ, 小塚, MS など
 - ・ プリセットにないものは手動設定
 - ・ $\text{Lua}\text{\LaTeX}$ は[ここ](#)や[公式ドキュメント](#)を参考に
 - ・ $\text{X}\text{\LaTeX}$ は[ここ](#)や[公式ドキュメント](#)を参考に
- ・ \LaTeX に詳しくないなら $\text{Lua}\text{\LaTeX}$ を使うと無難
- ・ 個別設定とプリセットではプリセットが優先される

フォント変更: 和文プリセットの設定例

```
jfontpreset: noto-otf
```

```
jfontpresetoptions:
```

- match
- deluxe
- no-math

フォント変更: 和文フォントプリセット

- ・ 詳細: $\text{X}\text{\LaTeX}$, $\text{Lua}\text{\LaTeX}$

表 4: 主なプリセット名対照表

| フォント | $\text{X}\text{\LaTeX}$ | $\text{Lua}\text{\LaTeX}$ |
|------|-------------------------|---------------------------|
| ヒラギノ | hiragino | hiragino-pro |
| MS | ms | ms |
| Noto | noto/noto-jp | noto-otf/noto-otc |

- ・ 游書体は OS バージョンごとに異なる
 - ・ ym-win, yu-win10, yu-osx
- ・ Ubuntu で Noto を選ぶ場合は noto で OK
- ・ 上記以外にも多数

インラインでのフォント変更

- ・ 明朝 / ローマンにしたいなら `\textmc/\textrm`
- ・ 本文中の一部だけフォントを変更したい時は `\jfontspec{}` を使う
 - ・ 欧文なら `\fontspec{}`
 - ・ Lua^AT_EX では和文欧文の同時一括切り替え不可
- ・ 詳しくは [luatex-ja のマニュアル](#)

ここはいつものフォント.

```
\textmc{ここだけ明朝 \textrm{Mincho}}`{=latex}
```

ここはいつものフォント. ここだけ明朝 Mincho.

スライドのテーマ変更

- ・ 指定できる名前一覧は[ここ](#)を参照
 - ・ metropolis テーマはあまりカラーバリエーションがない
 - ・ 数式をサンセリフにしたい場合は以下のように
 - ・ rownumber_chunk=デフォルトで行番号を付けるかどうか

output:

```
rmdja::beamer_presentation_ja:  
  fonttheme: professionalfonts  
  rownumber_chunk: true
```

シンタックスハイライトのテーマ変更

- ・ テーマは以下が用意されている
 - ・ default, tango, pygments, kate, monochrome, espresso, zenburn, haddock, breezedark, textmate
 - ・ 参考 [Xie Yihui のドキュメント](#)

output:

```
rmdja::beamer_presentation_ja:
```

```
highlight: tango
```

ハイパーリンクの色の変更

- ・ YAML ヘッダのトップレベルに記述する
- ・ linkcolor スライド内リンク
- ・ citecolor 参考文献リストへのリンク
- ・ urlcolor url リンク
- ・ デフォルトで可以使用できる色名は[ここ](#)を参照

```
linkcolor: blue
citecolor: green
urlcolor: red
```

アスペクト比の変更

- ・ 1610(16:10), 149(14:9), 54(5:4), 43(4:3), 32(3:2) から選べる
- ・ 160 mm x 90 mm にする例
 - ・ 出力画像も合わせたほうが調整しやすい

output:

`rmdja::beamer_presentation_ja:`

`fig_width: 6.29921`

`fig_height: 3.54331`

`aspectratio: 169`

引用形式の変更

- ・ 3 種類の出力方法
- ・ natbib 以外で良いなら TeXLive 不要
- ・ natbib
 - ・ `jecon.bst` が使える
 - ・ TeXLive が必要な原因
- ・ biblatex(+biber)
 - ・ 有力な日本語フォーマットがない?
 - ・ TeXLive 不要
- ・ citeproc: pandoc-citeproc の機能
 - ・ `cs1` ファイルで参考文献リストの体裁指定
 - ・ TeXLive 不要

引用形式の変更例

- ・ 今回は natbib パッケージを使用
- ・ natbib で「著者 (出版年)」表示にしたい場合は以下.
 - ・ その他のオプションは [natnotes.pdf](#) を参照

output:

```
rmdja::beamer_presentation_ja:  
  citation_package: natbib  
  citation_options: authoryear
```

参考文献リストの変更

- ・ .bib, .bst は以下にファイルパスを指定する
- ・ .bst は TeX 側が認識していればフルパス・相対パスである必要なし

```
bibliography: examples.bib  
biblio-style: jecon
```

「図」「表」の表示

- ・ 図や表を掲載するとキャプションの先頭に「図 X」「表 Y」などと表示される
 - ・ “Fig.”, “Tab.” などと表示したい場合は以下のように変更
- ・ 参考文献リストを載せる場合, `biblio-title` で見出しを変更できる

output:

```
rmdja::beamer_presentation_ja:
```

```
figurename: Fig.
```

```
tablename: Tab.
```

```
biblio-title: Further Readings
```

トラブルシューティング

Q 1: オプションが反映されない

- ・ A1. PDF の生成に失敗しただけで, 前回の PDF から更新されていないかも
- ・ A2. 書く場所を間違っている
 - ・ yaml ヘッダの入れ子には意味がある.
 - ・ すこしややこしいので解説

```
?rmdja::beamer_presentation_ja
```

- ・ pandoc 本来の引数と紛らわしい名前があるので注意
- ・ A3. 実際バグかも

Q 2: エラーの原因がよくわからない

- ・ A1: キャッシュ削除すると良くなることもある
 - ・ 前回失敗した際のキャッシュが悪さしてることは結構ある
 - ・ {ファイル名}_cache, {ファイル名}_files を消す
 - ・ cache = F
 - ・ エラーメッセージが実態と矛盾してるときはまず試す
- ・ A2: rmarkdown/knitr と \LaTeX どちらのエラーか確認
 - ・ output file: {ファイル名}.md と出れば pandoc まで機能している
 - ・ pandoc の変換が意図したものでない可能性はある

まとめ

結果どうなったか

- ・ 改善したこと
 - ・ `lstlisting.sty` より見やすいシンタックスハイライト
 - ・ R の画像や数値出力をコピーしなくて済む
 - ・ 一画面に収めるための構成だけ考えれば済むように
- ・ 悪化したこと
 - ・ 数式のリアルタイム補完は LyX が依然優秀
 - ・ python 作業中 (jupyter への) 不満高まり
 - ・ ポンチ絵も ggplot2 で作らねばという強迫症状
 - ・ 以前より組版に神経質になった

改良・機能追加したいところ

- ・ 手動インストール作業の削減
 - ・ TeXLive を入れなくても動かせるようにしたい
 - ・ たぶん tinytex と rmarkdown 両方がネック
- ・ 細かいレイアウト修正
 - ・ 例: キャプションが上か下かで統一されていない
 - ・ XeLaTeX と LuaLaTeX で微妙に文字サイズが違う
- ・ 他の言語のシンタックスハイライト
- ・ ggplot2 以外で描かれたグラフの対応
 - ・ 埋め込みはできるがフォントの調整が困難
 - ・ igraph みたいなのか...
- ・ [issues](#) に詳細

補足: 細かい技術的な話

- ・ 単に使いたいだけの人を見る必要なし
 - ・ 内部処理知りたい人向け

1. $\text{X}\text{\LaTeX}$ で取り消し線を付ける場合の問題
 - ・ 和文に取り消し線を付けるとタイプセットエラーが発生した
 - ・ `zxjatype` と `ulem` の競合と思われる
 - ・ `xeCJKnftf` を読み込むとなんか解決した
 - ・ 詳細: [TeX フォーラム](#)
2. $\text{Lua}\text{\LaTeX}$ と $\text{X}\text{\LaTeX}$ で文字サイズが変わってしまう

技術的に厄介だったところ

- ・ html と pdf(L^AT_EX) とで微妙に違う挙動
 - ・ ネット上の情報は html 前提が多い
 - ・ pandoc チョットワカル必要
- ・ 日本語を含む参考文献リスト
 - ・ upBibT_EX の適用
 - ・ 細かいオプション, 特に metropolis 特有の仕様
- ・ RStudio Cloud で動くかは未確認
 - ・ 日本語表示がおかしい説あり

- ・ 初期バージョンでは R 側で設定を書いていた
- ・ pandoc のテンプレートでかなり代替できると気づく
- ・ 結果だいぶシンプルな仕様に

日本語文献にどう対応しているか

- ・ `jecon.bst` を使いたい
 - ・ マルチバイト文字未対応 の BibTeX
 - ・ 日本語は `upBibTeX` 必要
 - ・ `biblatex` ではフォーマットに不満
- ・ `rmarkdown/tinytex` は日本語書誌情報処理未対応
 - ・ 内部では自前の設定で `Tex Live + latexmk` を呼び出し
 - ・ 呼び出しているラッパにオプションなし
 - ・ 積極的に改修の気配なし (参考)
- ・ 自前の設定を使用する (参考)
 - ・ `tinytex.latexmk.emulation = F`
 - ・ [ここ](#)を参考に`.latexmkrc` 設定
 - ・ Rmd と同じディレクトリに上記を置く

- ・ これを作るにあたって大いに参考になった資料
 - ・ Kazutan: 『R Markdown の内部とテンプレート開発』
 - ・ Atusy: 『R Markdown のオリジナルフォーマットを作ろう』
- ・ 文句言ったら光の速さで PR 出してくれた Atusy 氏
- ・ TeX Forum で質問に答えてくれた方々
- ・ 今風のデザインのヒントを与えてくれたところ
 - ・ pecorarista/sakuratheme
 - ・ ナウい Beamer スライド@Dentoo.LT #23

参考文献

- [1] JJ Allaire, Yihui Xie, Jonathan McPherson, Javier Luraschi, Kevin Ushey, Aron Atkins, Hadley Wickham, Joe Cheng, Winston Chang, and Richard Iannone. *rmarkdown: Dynamic Documents for R*, 2020. R package version 2.3.
- [2] Frederik Aust. *citr: RStudio Add-in to Insert Markdown Citations*, 2019. R package version 0.3.2.
- [3] Hadley Wickham. *tidyverse: Easily Install and Load the 'Tidyverse'*, 2019. R package version 1.3.0.
- [4] Hadley Wickham and Garrett Golemund. *R for Data Science: Import, Tidy, Transform, Visualize, and Model Data*. O'Reilly, Sebastopol, CA, first edition edition, 2016.
- [5] Yihui Xie. *bookdown: Authoring Books and Technical Documents with R Markdown*. Chapman & Hall, 2020.
- [6] Yihui Xie, JJ. Allaire, and Garrett Golemund. *R Markdown: The Definitive Guide*. Chapman and Hall/CRC, Boca Raton, Florida, 2018. ISBN 9781138359338.

- [7] 晴彦奥村, 裕介黒木. LATEX2 ϵ 美文書作成入門. 技術評論社, 東京, 第 7 版, 2017.