



人大 Beamer 模板

作者

机构

2020 年 02 月 24 日



目录

准备工作

常用语法





准备工作

模板说明

- 简单来说，这个模板可以实现下面这件事：在 Rstudio 中使用简洁的 markdown 语法生成人大主题的 beamer。具体过程为：
 - ① 作者在 Rstudio 中编写 Rmarkdown 文件，点击 knit 按钮。
 - ② Pandoc 将 Rmd 文件编译成 tex 文件。
 - ③ LaTeX 编译引擎根据 tex 文件生成 beamer。
 - ④ 自动删除编译所需的 tex 文件以及相关文件，只保留 Rmd 文件以及最后的 beamer。
- 其中需要作者自己做的只有第一步，相比于使用 LaTeX 写 beamer，方便很多。



模板说明

- 几乎所有的操作（新建幻灯片、插图、插表等）都可以使用简洁的 markdown 语法实现。
- 当作者需要进行数学符号的输入时，也可以无缝对接 LaTeX 语法。
- 可以展示 R 代码及其运行结果。
- 此外：这个模板使用的是 ctex 宏包，可以实现较为和谐的中英文混编；背景是人大主题。



需要安装软件

- 要使用这一模板，需要安装以下软件：
 - Texlive (为了能编译出 beamer)
 - R
 - Rstudio (为了使用 Rmarkdown)
- 此外，在 R 中还需要安装以下包：
 - knitr (为了编译 Rmarkdown)
 - rticles (支持中文)
 - tinytex (轻量级的 LaTeX)





常用语法



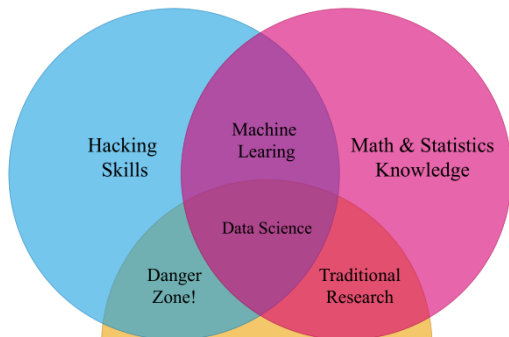
基础语法

- 基本可以使用 Markdown 语法来编写幻灯片：
 - 最常用的两个指令就是 # 和 ##。其中 # 用于开始一个新的章节，## 用于开始一个新的幻灯片。
 - 两段文字之间空一行即可分段。
 - 文字之间的空格不显示，但中文和英文之间会自动留出一个空格，所以不管书写的时候有没有写空格，最终输出结果中中英文之间还是会留白。按照个人书写习惯即可。



插图很简单，同样沿用 markdown 语法，比在 LaTeX 中简洁很多。插图指令为 ``，其中 `[]` 中填写图片名称，`()` 中填写图片路径。例如 `![韦恩图](figure/VD.png)` 表示在此处插入 `figure` 文件夹中的 `VD.png` 图片，并起名为韦恩图。

注意，因为 `figure` 跟这个 `Rmd` 文件是在同一个文件夹中，所以路径可以简化。如果是想要使用其它文件夹中的图片，需要写完整路径，如 `![韦恩图](D:/picture/VD.png)`。





上一张幻灯片的图片太大了。我们可以使用 `! [韦恩图](figure/VD.png){ width="40%" }` 将图片缩小为原来的 40%，如下所示。需要注意的是 `width=40%` 跟左右大括号之间必须有空格。

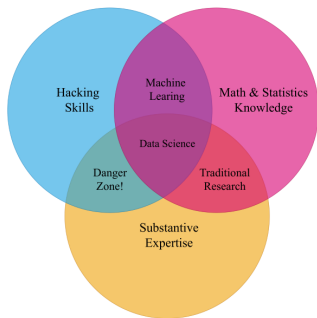


图 2: 韦恩图

表

在 markdown 中插入表格比在 LaTeX 中简单很多。比如在下面的表格中：第一行表示每一列的名字，用竖线隔开；第二行表示对齐方式，:---是左对齐，---:是右对齐，:---:是居中对齐。从第三行开始，是具体的数据，每一行表示一个人，每一列表示对应变量的具体取值。

姓名	性别	年龄
张三	男	21
小白	女	20
王五	男	45

如果表格太长，可以在 LaTeX Tables Online 网站上将 Excel 表格转成上述 markdown 格式，再复制粘贴即可。



超链接

要插入超链接，使用 `[]()`，其中 `()` 中是网址，`[]` 中是超链接的名字。比如 必应。在生成的 pdf 中点“必应”就可以跳转到对应网址。需要注意 `[]` 与左边的文字之间必须要有空格。



数学符号

跟 LaTeX 完全一致，如果需要在句子中插入数学符号或者式子，比如 x_1 ，就左右各用一个美元符号。如果是要空出一行单独显示式子，比如：

$$x_1 + x_2 = y$$

就空出一行，并且左右各用两个美元符号。



分级标题

在每一页幻灯片中，可以分别用 * + - 生成不同级别的 item，注意缩进，注意符号后要有一个空格，例如：

- 中国
 - 广东
 - 广州
 - 深圳
 - 湖南
 - 长沙



分级标题

如果需要枚举，用“数字.”代替上面的 * + - 即可，注意“.”之后要有空格，例如：

- 中国
 - 广东
 - ① 广州
 - ② 深圳
 - 湖南
 - ① 长沙



R 代码及结果

可以在 beamer 中插入 R 代码及其运行结果，例如：

```
a = 2
```

```
a
```

```
## [1] 2
```


R 代码及结果

值得注意的是，整份 Rmd 文件中的 R 代码是有前后顺序的。比如我们在上一页定义 a 取值为 2，在后面的代码中，只要这一变量没有被覆盖定义，这一结果依然有效。例如：

```
a+3 # a=2
```

```
## [1] 5
```

```
a^2 # a=2
```

```
## [1] 4
```

```
a = 5 # 重新定义  
a
```

```
## [1] 5
```



谢 谢!

