

Rmarkdown_practice

Kim Youngrae

2018 8 20

Contents

1	Introduction	1
2	Installation	1
3	Basic	2
3.1	Example. Basic calculation	2
3.2	Example. Plotting	2
4	Some useful syntax	3
4.1	text syntax	3
4.2	Chunk options	3
5	Tables	4
6	Global option	4
7	Exercise	5

1 Introduction

- R markdown은 Code와 Text를 동시에 포함한 문서를 작성하는 데 도움을 주는 Package이다.
- R studio에서도 쉽게 사용 가능하며, Code가 문서의 중간에 포함된 자료분석 레포트 등을 작성하는데 용이하다.
- R markdown을 이용해서 document 형식도 만들 수 있고, Beamer 형식으로도 만들 수 있다.
- 이들을 pdf 파일로 저장하기 위해선, latex 프로그램이 필요하며 Windows의 경우 MiKTeX (Complete version ; <http://miktex.org/2.9/setup>), MAC OS의 경우 Texlive (Full version ; <http://tug.org/mactex/>) 를 이용하면 된다.

2 Installation

```
install.packages("rmarkdown")  
library(rmarkdown)
```

- R markdown을 사용하기 위해서는 먼저 “rmarkdown” package를 설치해야 한다.

3 Basic

- Rstudio에서는 R markdown 파일을 쉽게 만들 수 있다. R markdown의 File 탭 - New file - R markdown을 선택하면, 원하는 파일의 형식과, 문서의 타입을 결정할 수 있다.
- R markdown 파일을 만들고 나면 흰색 바탕의 Text 부분과 회색 바탕의 Code chunk 부분 두가지 부분으로 나뉜다.
- 흰색 바탕의 Text 부분에는 글을 적고, 회색 바탕의 Code chunk 부분에는 실행하여 결과를 보고싶은 Code를 적는다. 여기서는 R을 이용하므로, R Code를 적으면 된다.
- Code chunk는 ‘ ‘ ‘{r}’ 줄과 ‘ ‘ ‘줄 사이에 Code를 적으며, {} 안에는 다양한 chunk option들이 들어가게 된다. (여기서 ‘는 Backtick입니다.)

3.1 Example. Basic calculation

```
x=3; y=5;  
print(x)
```

```
## [1] 3
```

```
y
```

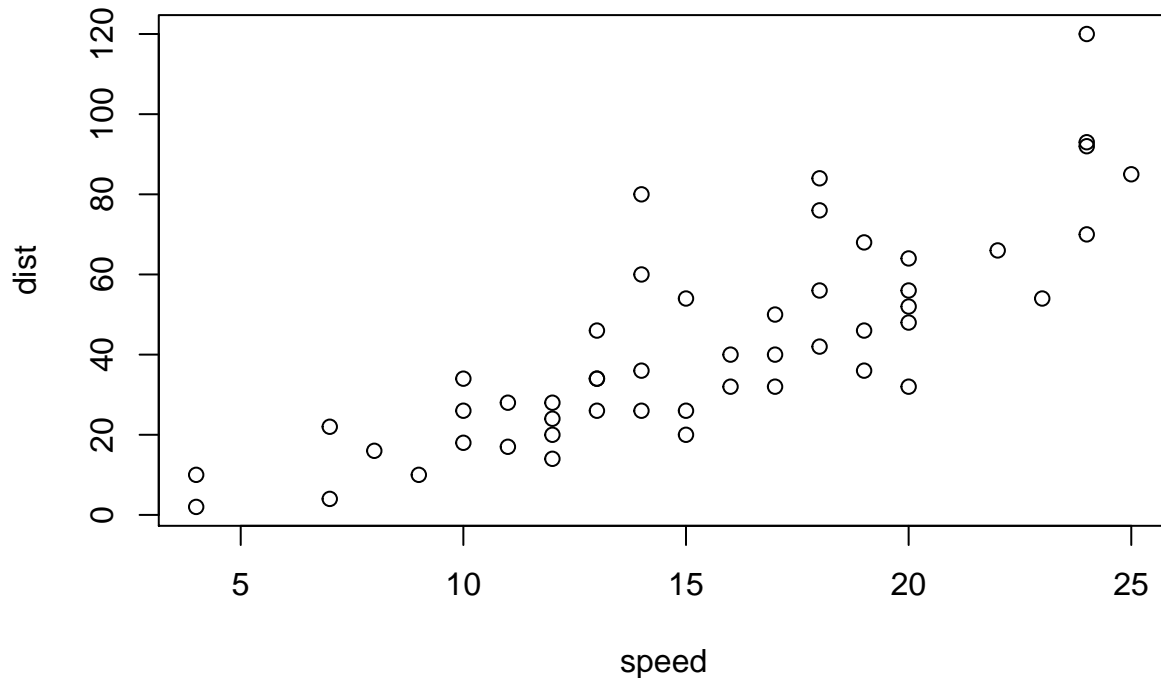
```
## [1] 5
```

```
x+y
```

```
## [1] 8
```

3.2 Example. Plotting

```
plot(cars)
```



4 Some useful syntax

4.1 text syntax

- `# ~ #####` : Header text
- `*,-` : 리스트() 생성. 띄어쓰기를 한 칸 해줘야 한다
- `1` : 숫자로 된 리스트 생성
- 공백 2칸 : 줄 바꿈
- `*text*`, `_text_` : 이탤릭체 (*ABC*, *ABC*)
- `*text*` `_text_` : 볼드체 (**ABC**, **ABC**)

4.2 Chunk options

Chunk option은 Code chunk가 시작하는 {}안에 넣을 수 있는 option들이다. 자주 사용하게 되는것들은 다음과 같다.

- `label` : Code chunk에 라벨을 사용할 수 있고, 코드 청크를 재사용할때 활용한다.
- `cache` : 앞에서 이미 계산된 값을 불러올 때 사용, 계산에 시간이 오래 걸리는 경우에 활용한다.
- `eval` : 해당 코드의 실행 여부를 결정한다. 코드만 보여주고 싶을때 사용한다.
- `include` : 해당 코드 청크를 보여줄지 말지 결정한다.
- `echo` : 소스코드를 보여줄지 말지 결정한다.
- `result` : 'hide' 옵션을 주면, 결과를 출력하지 않는다.
- figure 관련 option들
 - `fig.width` & `fig.height` : 그림의 크기를 결정하게 된다. (e.g. `fig.height=4`)
 - `fig.align` : 그림의 위치를 결정한다. (e.g. `fig.align='left'`)

- fig.cap : 그림의 캡션을 적는다.
- out.width & out.height : Output의 크기를 결정한다. (e.g. out.width = '80%')

5 Tables

문서를 작성하다 보면 Table을 그려야 하는 경우가 많은데, 그 중 knitr 패키지에 들어있는 kable이라는 함수를 이용하면 다음과 같다.

```
knitr::kable(iris[1:5, ], caption = 'A caption')
```

Table 1: A caption

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa

```
print(iris[1:5,])
```

```
##      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 Global option

R markdown 파일의 가장 위에, 파일의 형식, 제목등을 결정하는 부분이 있는데, 이곳에서 여러가지 조작을 통해, 조금 더 보기 편한 문서를 작성할 수 있다. 우선 여기서는 pdf 형식의 문서에서 쓸모있는 option들을 알아보자. 기본적으로 아래와 같이 되어있다.

```
---
title: "Rmarkdown_practice"
author: "Kim Youngrae"
date: '2018 8 20 '
output: pdf_document
---
```

이 문서에 사용된 option들은 다음과 같다.

```
---
title: "Rmarkdown_practice"
```

```

author: "Kim Youngrae"
date: '2018 8 20 '
output:
  pdf_document:
    toc: true
    toc_depth: 2
    number_sections: true
header-includes :
- \usepackage{kotex}
- \usepackage{amsmath}
fontsize: 11pt
---

```

하나씩 살펴보도록 하자.

- 목차를 만들기 위해선 toc라는 option이 필요하고, 이는 output : 아래 pdf_document: 아래 들어가게 된다. toc_depth는 어느 깊이까지 표현하는지에 대한 이야기로, #의 숫자마다 깊이가 결정된다.
- 한글을 사용하기 위해선 kotex 패키지가 필요하다. 이 뿐만 아니라, R markdown에서는 latex 패키지들을 사용할 수 있는데, 이를 사용하기 위해선 \usepackage{ name } 을 이용하고, header-includes : 아래 적으면 된다.
- fontsize도 여기서 결정할 수 있다.

수많은 option들이 있지만, 자주 사용되는 것으로는 위의 것들이 있다.

7 Exercise

위에서 배운것들을 이용하여 다음의 문제를 풀어 Markdown으로 작성해 보세요.

- ISLR package의 Auto data를 이용해서, mpg를 response variable, horsepower를 covariate으로 하는 단순선형회귀분석을 시행합니다. 이를 이용해 다음에 답하세요.
 - 얻어진 모형의 회귀계수와, standard error를 table로 print하세요.
 - Data의 산점도를 그리고, 그 위에 회귀직선을 붉은색으로 그려보세요. (불필요한 결과와 Code는 출력하지 않고 만들어보세요.)