

A Sample R Markdown Template

Vivek Patil

9/15/2015

Abstract

The objective of this document's template is to demonstrate some basics of the combination of R and markdown and how they can be knitted together using the knitr package (via the RStudio IDE) to produce beautiful docs/reports.

Contents

1	Lists	2
1.1	Numbered list	2
1.2	Bulleted (unordered) list	2
2	How can headers be defined?	2
3	Header 1	2
3.1	Header 2	2
3.1.1	Header 3	2
4	Images	2
5	Tables	3
6	HTML Tags	3
7	Equation	3
8	Blockquotes	3
9	Manual Line Breaks	3
10	Miscellaneous	3
11	Let's talk about mixing R code with markdown	3
11.1	Getting to know your dataset	3

The options mentioned in the header of this Rmarkdown file (with an extension of .Rmd) above for a pdf document can be modified as well as supplemented by options for other document formats.(Please see the “Output Options” section of RStudio’s reference site at <http://rmarkdown.rstudio.com/> for more information.)

Did you notice that this thing was in bold and that the previous thing was a web link?

1 Lists

1.1 Numbered list

Material for this document has been heavily borrowed from several documents.(Notice how a numbered list is being created below.)

1. Markdown cheatsheet at <http://warpedvisions.org/projects/markdown-cheat-sheet.md>.
2. R markdown reference from R Studio's site at <http://rmarkdown.rstudio.com/>
3. http://rmarkdown.rstudio.com/authoring_basics.html
 - Item 3a
 - Item 3b
4. Source 4

1.2 Bulleted (unordered) list

As previously mentioned, the objectives of this document are two fold.

- Learn the basics of markdown
- Learn how R code and output can be interspersed with markdown to create reports/documents.
- Another one
 - Item 3a
 - Item 3b
- And yet another one

2 How can headers be defined?

The header used above can be created using a different approach, using a single *hashtag* (wasn't that in *italics*?) to the left of the title of the header.

3 Header 1

3.1 Header 2

3.1.1 Header 3

3.1.1.1 Header 4

3.1.1.1.1 Header 5

4 Images

You can insert images in the document as well. For instance, if you have an image of a real apple in your working directory, then you can insert it in the document in the following manner.

If you wanted to show an images from a website, then that can also be done.



Figure 1: This is a picture of a honey crisp apple

5 Tables

Header	Header	Right
Cell	Cell	\$10
Cell	Cell	\$20

- Outer pipes on tables are optional (if you are not concerned with aesthetics, just drop)
- Colon used for alignment (right versus left)

6 HTML Tags

You can use html tags as well in markdown documents. For example, you could've used an `` tag to insert images. Super basics of html can be [found here](#)

7 Equation

Standard deviation, $s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2}$

8 Blockquotes

A friend once said:

It's always better to give than to receive.

9 Manual Line Breaks

End a line with two or more spaces

For
example, this
line looks insanely
chopped.

10 Miscellaneous

superscript²
~~strikethrough~~

11 Let's talk about mixing R code with markdown

11.1 Getting to know your dataset

```
dim(iris)
```

```
## [1] 150 5
```

Number of rows

```
nrow(iris)
```

```
## [1] 150
```

Number of columns

```
ncol(iris)
```

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

Names of variables

```
names(iris) # colnames(iris) also gives that information
```

```
## [1] "Sepal.Length" "Sepal.Width" "Petal.Length" "Petal.Width"  
## [5] "Species"
```

First 6 rows

```
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
```

First 2 rows

```
head(iris,2) # alternately, can use iris[1:2,]
```

```
## 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
```

Last 6 rows

```
tail(iris) # Number of rows can be controlled, see earlier example involving the head command
```

```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width  Species
## 145          6.7          3.3          5.7          2.5 virginica
## 146          6.7          3.0          5.2          2.3 virginica
## 147          6.3          2.5          5.0          1.9 virginica
## 148          6.5          3.0          5.2          2.0 virginica
## 149          6.2          3.4          5.4          2.3 virginica
## 150          5.9          3.0          5.1          1.8 virginica
```

First row

```
iris[1,]
```

```
##      Sepal.Length Sepal.Width Petal.Length Petal.Width Species
## 1          5.1          3.5          1.4          0.2  setosa
```

First row, first column

```
iris[1,1]
```

```
## [1] 5.1
```

Name of third column

```
names(iris)[3]
```

```
## [1] "Petal.Length"
```

3 entries from third column

```
head(iris[3],3) # alternately, can use iris[1:3,3]
```

```
##      Petal.Length
## 1          1.4
## 2          1.4
## 3          1.3
```

Structure of the dataframe (dataset)

```
str(iris)
```

```
## 'data.frame':   150 obs. of  5 variables:
## $ Sepal.Length: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
## $ Sepal.Width : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
## $ Petal.Length: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
## $ Petal.Width : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
## $ Species      : Factor w/ 3 levels "setosa","versicolor",...: 1 1 1 1 1 1 1 1 1 1 ...
```

Summary of the dataframe

```
summary(iris) # for factor/categorical variables, this gives a count of all categories
```

```
## Sepal.Length Sepal.Width Petal.Length Petal.Width
## Min. :4.300 Min. :2.000 Min. :1.000 Min. :0.100
## 1st Qu.:5.100 1st Qu.:2.800 1st Qu.:1.600 1st Qu.:0.300
## Median :5.800 Median :3.000 Median :4.350 Median :1.300
## Mean :5.843 Mean :3.057 Mean :3.758 Mean :1.199
## 3rd Qu.:6.400 3rd Qu.:3.300 3rd Qu.:5.100 3rd Qu.:1.800
## Max. :7.900 Max. :4.400 Max. :6.900 Max. :2.500
## Species
## setosa :50
## versicolor:50
## virginica :50
##
##
##
```

Create a dataframe

```
Person=c("A","B","C","D","E")
Age=c(15,20,25,30,35)
page=data.frame(Person,Age)
mean(Age) # gives the mean of the variable Age, prior to the creation of the dataset
```

```
## [1] 25
```

```
Age="" # (resetting that)
mean(Age) # Haha
```

```
## Warning in mean.default(Age): argument is not numeric or logical: returning
## NA
```

```
## [1] NA
```

```
mean(page$Age)
```

```
## [1] 25
```

```
mean(page[,2])
```

```
## [1] 25
```

```
summary(page)
```

```
## Person Age
## A:1 Min. :15
## B:1 1st Qu.:20
## C:1 Median :25
## D:1 Mean :25
## E:1 3rd Qu.:30
## Max. :35
```