This is the Title of Your Report Generated from R Markdown

Vinh "MBALearnsToCode" Luong

What is Markdown?

Markdown is an easy text file format that allows you to write up a document and apply stylings to create highly presentable output that can be published in various ways, including as PDF and as HTML webpages.

Common Markdown syntax

Refer to the PDF output (created by "**Knit PDF**" in this RStudio window) to see how the below inputs are rendered in the final PDF output:

Make a 1st-Level Header by inserting 1 hash character in front

Make a 2nd-Level Header by inserting 2 hash characters in front

Make a 3rd-Level Header by inserting 3 hash characters in front

make italic text by putting 1 underline character at each end

make bold text by putting 2 star characters at each end

then, obviously, this is how to make bold italic text

Make an numbered (ordered) list by simply putting "1."",2."", "3.", etc., like so:

1. First item
2. Second item
3. Third item

Make an unordered list by putting "-" and a space in front of each item, like so:

- Unordered item
- Unordered item
- Unordered item

Make hyperlinked text, e.g. referring to Google.com like this.

Images can be embedded easily, with a caption too, like Figure 1 below:

Figure 1: Chicago Booth logo

Embedding R code in R Markdown

The real power of R Markdown is in allowing you to embed R code into your document, and get the code executed and its results presented nicely the output - so that you can tell a wholesome, coherent story of your analysis.

You can insert R code to be run between the triple back-ticks like so:

```
library(ggplot2)   # comment: load GGPlot2 package
summary(cars)   # summarize Cars data set
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

R code blocks run sequentially, so you can continue with your R analysis in a separate code block, e.g. to make a plot like so: (note the figure width and heigth (in inches) settings in the {})

```
qplot(speed, dist, data=cars) + geom_smooth()   # a quick plot
```

```
## geom_smooth: method="auto" and size of largest group is <1000, so using loess. Use 'method = x' to cl
```

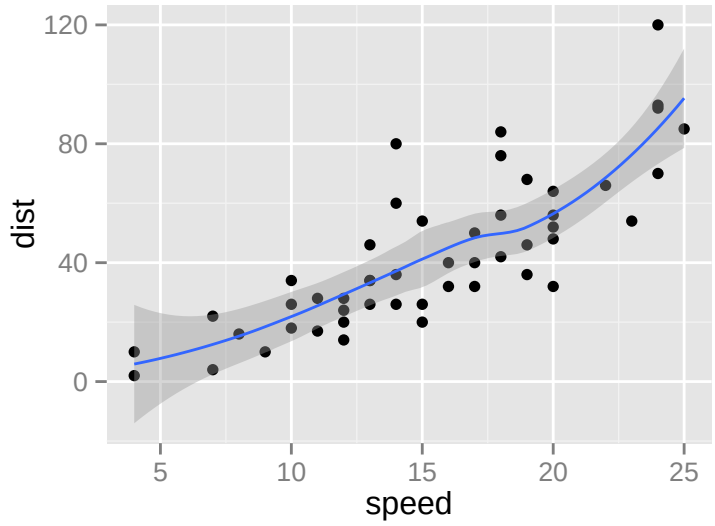

Figure 2:

Note that in the above examples:

- Each code block is repeated (echoed),
- Each code block's results are displayed, and
- All messages / warnings from running the R code are displayed

in the output document. Sometimes we may want to alter these display behaviors for our needs.

You can set a code block to be **not** echoed by specifying ***echo=FALSE*** like the following code block, which silently sets a variable **RVar** to value 1.2:

You can hide the outputs from a code block by specifying ***results='hide'*** like in the following code block, which will not display the variable *RVar*'s value:

```
print(RVar)
```

You can hide messages and/or warnings by specifying ***message=FALSE*** and/or ***warning=FALSE***:

```
qplot(speed, dist, data=cars) + geom_smooth()   # a quick plot without messages / warnings
```

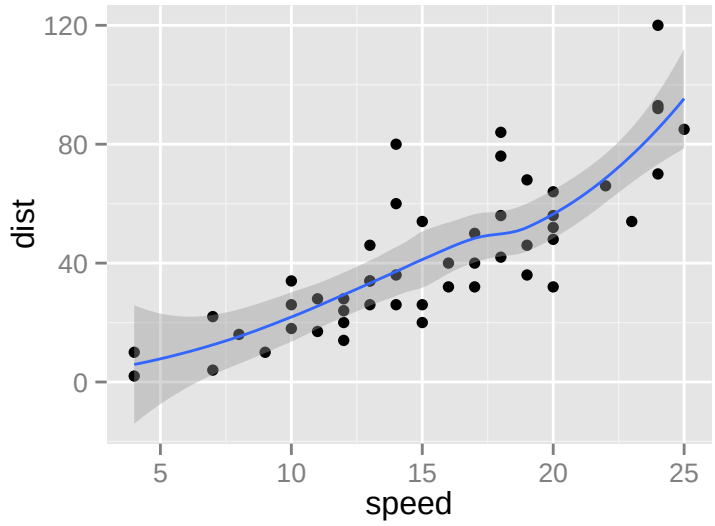

Figure 3:

Lastly, you can run certain short R code "in-line" and display its value by putting the code between single back-ticks, e.g. like in referring to the value of the *RVar* variable: RVar's value is currently *1.2*.

THE END.