Homework guidelines for Computational Statistics

All homeworks for this course should be written using RMarkdown in RStudio, and submitted as a PDF document. RMarkdown is a plain-text format that allows you to create rich documents in a variety of common formats, such as PDF, HTML, or MS Word. The basic idea is that you write a plain-text document with some plain-text formatting notations, and your machine (such as RStudio) turns that into a pretty PDF, HTML, or Word document. The power of RMarkdown is that it allows you to include R code and equations in a way that is easy to write, but produces attractive and legible output.

To create a new RMarkdown document, in RStudio you go to File -> New File -> R Markdown. This creates a new plain-text file with a .Rmd ending. Once we have created a new .Rmd plain-text document, you write your homework answers in that document, including any R code or equations you want, and then when you are done (or want to check how the output will look so far), you "knit" the .Rmd file into the final PDF, Word, or HTML document, which is what you will submit as your finished product. You will always have two documents for any assignment: the plain-text .Rmd file, and the prettier PDF, Word or HTML file you "knit" the .Rmd file into.

To "knit" the pretty document, you click on "Knit PDF" (for instance) in the Source pane in RStudio, which will create the PDF in the same directory as the .Rmd file, and also open it in RStudio to preview it. For this class, we want the homework submissions to be in PDF format, because that works well with Blackboard. Homework file names should be in the following format: HW_01_lastname_firstinitial.pdf, where of course 01 gets replaced with whatever the appropriate homework number is.

Generating PDFs

When you create a new RMarkdown document, RStudio asks you what format you want your final output to be in: PDF, Word, or HTML. Unfortunately, unlike HTML or Word formats, generating PDFs with RMarkdown requires one additional step. You have a few different options for getting PDFs to work in RStudio:

1. To generate PDFs directly with RStudio, you first need to install another program called "Latex." On a Mac, go here https://tug.org/mactex/ to install MacTex; on a Windows machine go here to install Miktex http://miktex.org/download. Once installed, this allows you to knit your .Rmd directly into PDF format.

2. If for any reason you have trouble installing Latex or it doesn't work, don't worry. The other way to generate a PDF is to "knit" the .Rmd as HTML, and then open the HTML in your browswer and print the page to a PDF. All Macs can do this, and in Windows, the Chrome browser has this ability built in.

3. A final option, similar to #2, is to "knit" the .Rmd as a Word document, and then open it in Word and Save As a PDF.

Any of these methods should work, and if you have any problems with method 1, don't waste your time trying to fix it, just go straight to 2 or 3, which do not require installing anything special.

The RMarkdown syntax

RMarkdown comes from an earlier language called Markdown, which was originally designed for writing plain HTML. RMarkdown is just Markdown with the additional ability to include R code and equations. For the purposes of the homeworks, you don't need anything fancy beyond the R code and equations, but if you want to do any fancier formatting, here is a quick reference sheet: http://en.support.wordpress.com/markdown-quick-reference/ . As you can see there, Markdown is just a way to generate nice looking

formatting using plain text. In fact, if you know HTML, you can actually just insert HTML directly into your RMarkdown code and it will mostly work fine.

When you open a new RMarkdown file, as you will see, it creates a new .Rmd document with some text already in there. If you knit this, you can see a nice example of how to format R code to get the right output. You can also use the homework_solution_example.Rmd file as your template, which is in the Course Resources folder.

If you want a quick review of how RMarkdown works in RStudio, you can go here: http://rmarkdown.rstudio.com/RMarkdownCheatSheet.pdf.

R Code

The lectures for this course were produced with RMarkdown, and as you can see from them, they do a nice job showing both the raw R code, and the output from that code, in pretty boxes in the final document. To include R code in your .Rmd file that produces nice output, you put your R code inside a triple set of apostrophe marks like so:

```
```{r}
summary(cars)
```
```

The default text RStudio includes every time you create a new .Rmd document also has some examples in it. When we knit the .Rmd file with this R code in it, it doesn't just print the R code – it actually *runs* the R code, and prints both the code itself, and the output of running that code – in this case, the summary information about the built-in cars dataset.

```r
summary(cars)
```

```
     speed            dist
 Min.   : 4.0   Min.   :  2
 1st Qu.:12.0   1st Qu.: 26
 Median :15.0   Median : 36
 Mean   :15.4   Mean   : 43
 3rd Qu.:19.0   3rd Qu.: 56
 Max.   :25.0   Max.   :120
```
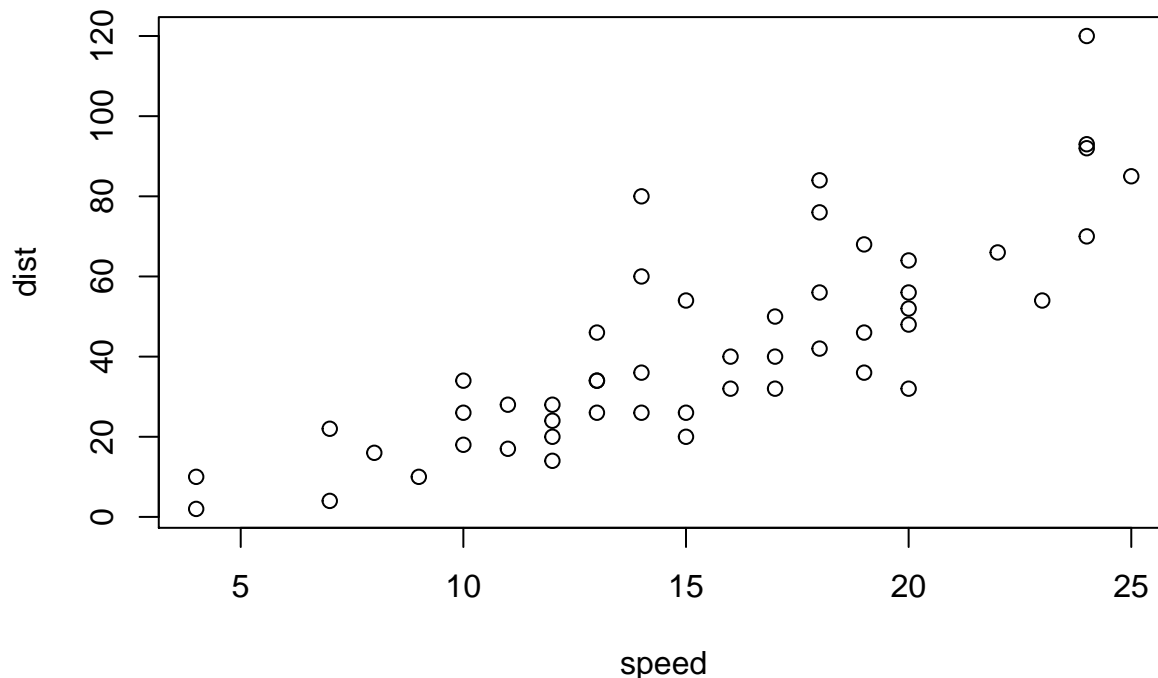
Plots

You can also embed R plots in your file, which is all done automatically just by including the R code that generates it:

```
```{r}
plot(cars)
```
```

And when you knit it, the above code produces:

```r
plot(cars)
```

You can put any R code inside the triple-apostrophes, and RStudio will quietly run it in the background and output the PDF, Word, or HTML document with the output included. Again, see the sample code when you create a new RMarkdown file, or you can even look directly at the .Rmd file that generated these Homework Guidelines, which I have included in the Course Resources.
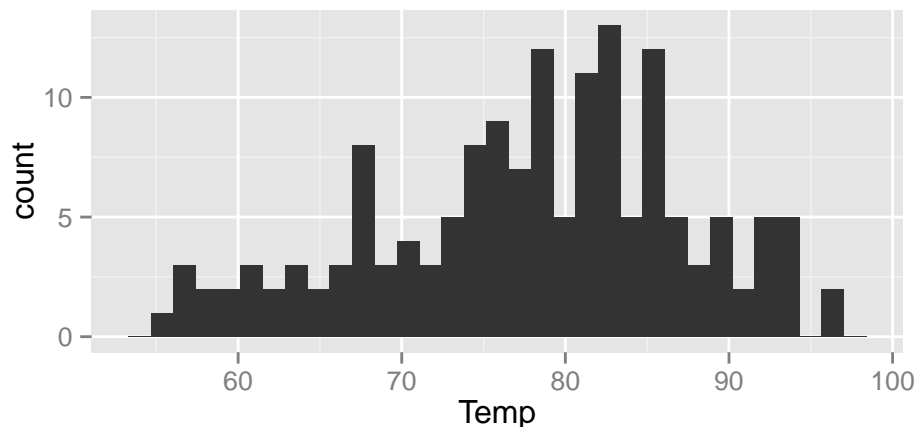
Note that if you are using any packages that need to be loaded, such as `ggplot2`, those packages need to be loaded in your .Rmd file just as they would be if you were running an R script. But as with R code, the package only needs to be run once. So if we wanted a fancier plot using `ggplot2` in our homework file, we would run somthing like:

```
```{r fig.width=5, fig.height=2.5, comment=NA}
library(ggplot2)
ggplot(data=airquality,aes(x=Temp)) + geom_histogram()
```
```

This produces the following when we knit it:

```
library(ggplot2)
ggplot(data=airquality,aes(x=Temp)) + geom_histogram()
```

```
## stat_bin: binwidth defaulted to range/30. Use 'binwidth = x' to adjust this.
```

Note that we can also change the figure size and width, which gives a bit more control over looks. Again, more information on these tweaks can be found here: http://rmarkdown.rstudio.com/RMarkdownCheatSheet.pdf.

Math and equations

Last but not least, RMarkdown also lets you typeset equations. To include mathematical equations, you just put your math in between `$` signs: `$3x+4$` outputs $3x + 4$.

There are a few basic examples of syntax for getting pretty equations. On the left is the raw syntax, and on the right the pretty output you get when you knit your .Rmd document.

In the long run, it is much easier to learn these basics than to write your math by hand, scan it, and send it in.

Basics: `$3x + 4y - 6.0z / 12 * 43.8$`: $3x + 4y - 6.0z/12 * 43.8$

Exponents: `$3^{2x}$` : 3^{2x}

Subscripts: `$Y_{i}$` : Y_i

Summation: `$\sum_{i=1}^{10} x_i$`: $\sum_{i=1}^{10} x_i$

Integral: `$\int_{1}^{10} x dx$`: $\int_1^{10} x dx$

Fractions: `$\frac{3x-9}{2}$` : $\frac{3x-9}{2}$

Hat: `$\hat{x}$`: $\hat{x}$

Bar: `$\bar{x}$`: $\bar{x}$

Square root: `$\sqrt{b^2-4ac}$`: $\sqrt{b^2 - 4ac}$

Some greek: `$\alpha$`: α, `$\beta$`: β, `$\chi$`: χ, `$\delta$`: δ, `$\epsilon$`: ϵ, `$\lambda$`: λ, `$\mu$`: μ, `$\pi$`: π, `$\rho$`: ρ, `$\sigma$`: σ, `$\theta$`: θ, `$\infty$`: ∞

To put an equation on its own line and centered, use two `$$` instead of just one:

`$$p(x; \mu, \sigma) = \frac{1}{\sigma \sqrt{2 \pi}} e^{\frac{-(x-\mu)^2}{2 \sigma^2}}$$`

$$p(x; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$$

For more math code, see here: http://faculty.cbu.ca/srodney/ShortSymbInd.pdf, though you probably won't need much more than is included here.

Give it a try! Open a new RMarkdown file, tinker with it a bit, add a few equations, and then hit "Knit HTML," or "Knit PDF" if you've installed Latex. You'll see that it's super-easy and produces nice-looking output instantly.

And again, remember you can use the homework_solution_example.Rmd in the Course Resources folder as your homework template.