Econ 2B03 Assignment (Insert Number)

Insert Your Name and Student ID

Insert Assignment Due Date

Source Code (R Markdown) for this Webpage

Link to R Markdown Code for this Document

R and RStudio

We will be using R, a language and environment for statistical computing and graphics (see http://r-project.org), and RStudio, an integrated development environment for R (see http://rstudio.com). Both of these programs must be installed on your computer in order to work with this document, work on assignments etc. These powerful, free, and open source programs allow you to work anywhere and anytime you wish and not be tethered to a lab running closed, licensed proprietary software.

Navigate through the **Resources** icon at the top of this page for instructions on installing R, RStudio, and optionally TeX (MS Windows, macOS, Linux) and git (TeX allows you to generate PDF files and git can be used for version control).

What is R Markdown?

This is an R Markdown document. R Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents in RStudio. For more details on using R Markdown see http://rmarkdown.rstudio.com.

Although you might not realize it, by using R Markdown you are in fact conducting *reproducible research* (the idea that others may readily verify your findings and build upon them) since your code and narrative involve only one file (your R Markdown file). For your viewing pleasure see A reproducibility horror story!

Creating a new R Markdown Document in RStudio

From within RStudio navigate to the menu **File -> New File -> R Markdown**. Insert your name and assignment information at the top and delete the examples that follow after the fifth line.

Including R Results in your R Markdown Document

To include results generated from your R commands in your R Markdown document, starting in RStudio click on the **Insert** button from within the **Editor** pane (upper left by default) and select **R** from the pulldown menu, then write your code inside the 'chunk' that is inserted. If you want to run the code inside the chunk, simply click on the **Run** button and select the appropriate run argument.

You can also embed an R code chunk manually (make a code chunk with three back ticks followed by an r in braces; end the chunk with three back ticks).

Reading data from a URL

R can read data from a URL saving you from manually downloading the data in certain instances:

```
course <- read.table("https://socialsciences.mcmaster.ca/racinej/768/files/attend.RData")
## attach() makes the names of the data `course' known to R functions (scope)
attach(course)
```

You can summarize data in your R Markdown document:

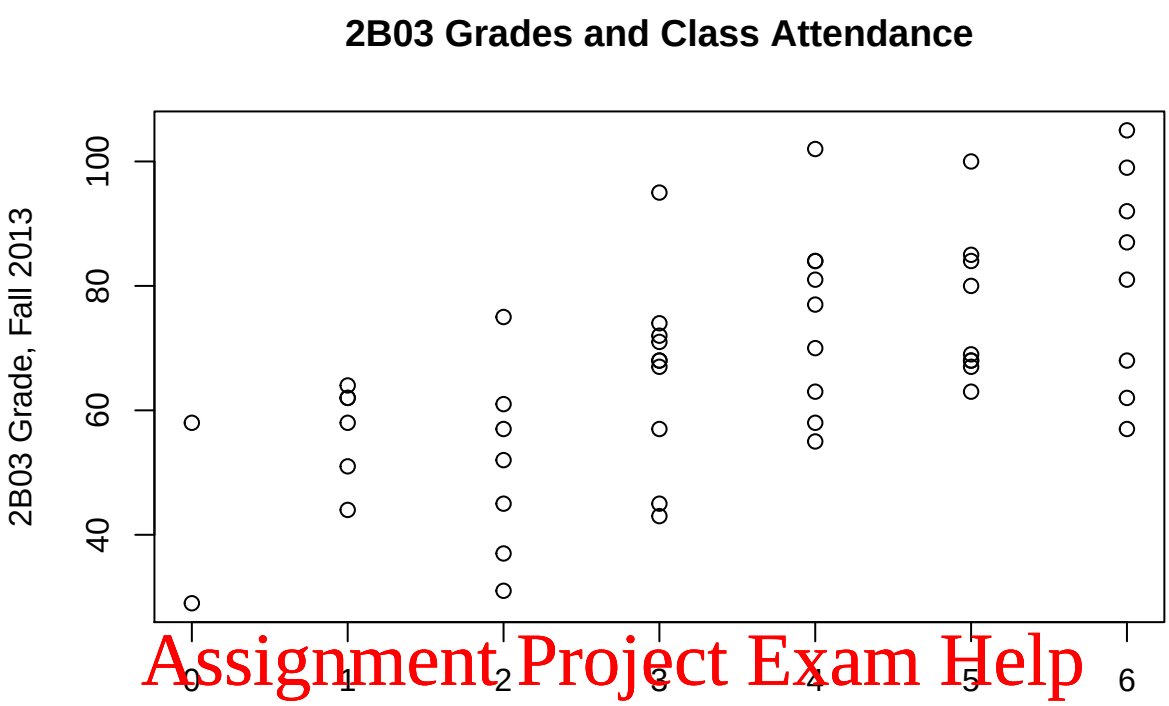
```
summary(course)
```

```
##      grade            attend
##  Min.   : 29.00   Min.   :0.00
##  1st Qu.: 57.50   1st Qu.:2.00
##  Median : 68.00   Median :4.00
##  Mean   : 67.75   Mean   :3.51
##  3rd Qu.: 80.50   3rd Qu.:5.00
##  Max.   :105.00   Max.   :6.00
```

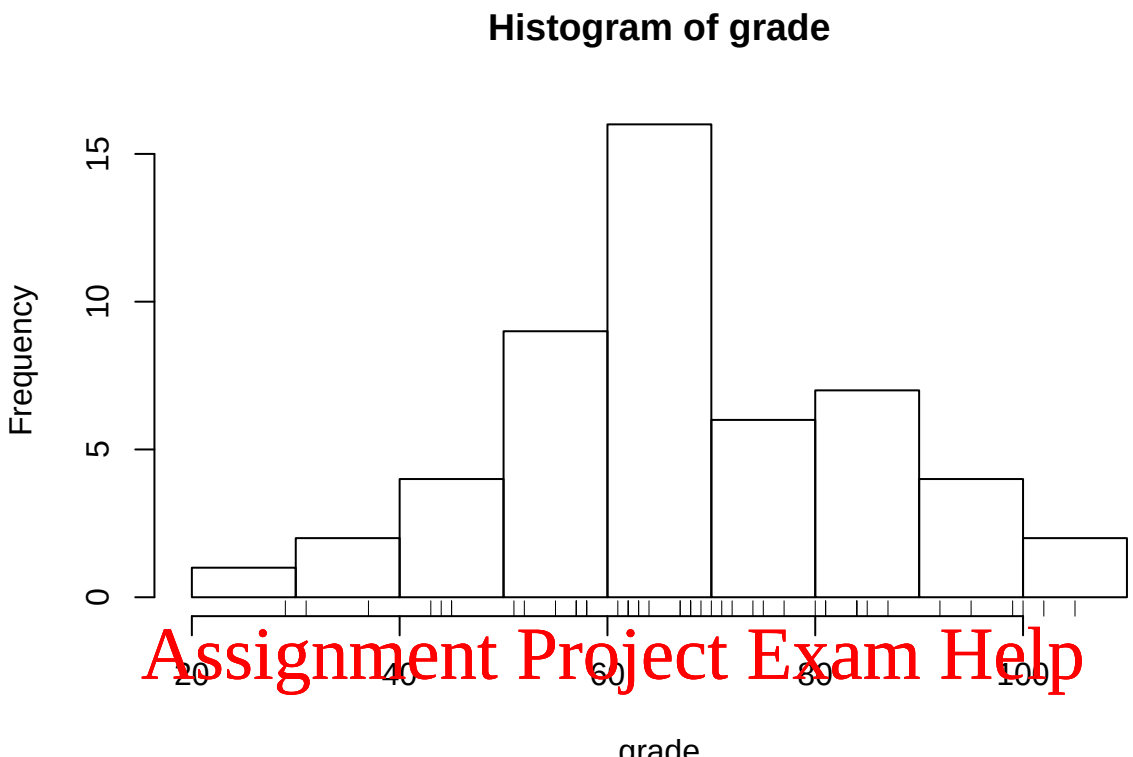
Including Plots

You can also embed simple scatter plots in your R Markdown document, for example:


```
plot(attend,grade,
     ylab="2B03 Grade, Fall 2013",
     xlab="# Times Present in Class When Assignments/Exams Returned",
     main="2B03 Grades and Class Attendance")
```



(Click on **R Markdown Code for this Document** and scroll down to view the R Markdown code for this plot.)



(Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot. Note that grades > 100 are possible due to bonus marks being awarded for using R Markdown for generating your assignments. Click on **R Markdown Code for this document** and scroll down to view the R Markdown code for this plot.)

Including Bulleted and Numbered lists

Making lists in R Markdown is simple providing you pay close attention to the Markdown convention:

- Bulleted lists are created by starting the line with a dash
  - To create sub lists indent four spaces then start with a dash

    If you want text to appear below a list item, create a a blank line and then indent an *additional* four more spaces beyond the spacing of the previous item
- Next item

1. Numbered lists start with a number followed by a period

   (i) Sub items are indented four spaces and start with e.g. (i) (or (a) if you prefer)

   If you want text to appear below a list item create a blank line and then indent an *additional* four more spaces beyond the spacing of the previous item
2. Next item

See Troubleshooting and Tips below for further information.

Including Tables

(Click on **R Markdown Code for this Document** and scroll down to view the R Markdown code for these tables.)

Creating tables is straightforward as the following two examples demonstrate:

Table Header	Second Header	Third Header
Table Cell 1	Cell 2	Cell 3
Cell 4	Cell 5	Cell 6

Table 2: Here's the caption. It, too, may span multiple lines.

Centered Header	Default Aligned	Right Aligned	Left Aligned
First	row	12.0	Example of a row that spans multiple lines.
Second	row	5.0	Here's another one. Note the blank line between rows.

For guidance on creating the above table and more sophisticated tables using R Markdown see Creating Tables in Markdown.

Including Verbatim (i.e. freeform) Text

You can include text that appears *exactly* as you type it in your document by enclosing the code chunk with three backticks at the top and bottom:


```
This          is verbatim x1        x2
        x3
```

Knitting your R Markdown Document

When you click the **Knit** button from within RStudio (this button should appear in the top left pane in RStudio by default), a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.

Flexible Document Creation

It is worth noting that your document does not have to be related to the R language in order to use Markdown. Your document can in fact use other computing languages (C++, SQL, Python, etc.), and it can even be totally unrelated to conducting statistical analysis (for instance, you might write a story, book, or collection of poems).

Printing Your Assignment for Submitting in Class

- You can **spell-check** your assignment before handing it in by navigating the menu **Edit -> Check Spelling**

- The default R Markdown output format is HTML. To generate an HTML document, simply click on the **Knit** button (you might need to first select the **Knit to HTML** option on the pull down menu associated with the **Knit** button); next, click on the **Open in Browser** button in the viewer that opens (this will open the document in your web browser); finally, print your document using your web browser's print facilities (this HTML document will remain in your working directory)
- If you have Microsoft Word installed on your system you can pull down the menu associated with the **Knit** button and select the **Knit to Word** option and, if all goes well, you will be presented with a Word document that you can print using Word's print facilities (this Word document will remain in your working directory)
- After installing TeX on your system, you can pull down the menu associated with the **Knit** button and select the **Knit to PDF** option then you can print using your PDF browser's print facilities (this PDF document will remain in your working directory)

Typesetting Mathematics

If you installed TeX/LaTeX on your computer, then you will see that R Markdown supports mathematics typesetting using TeX/LaTeX. So you can type math inline using the standard approach, e.g., enclose your TeX equation commands such as `$\sum_{i=1}^n(X_i-\bar X)^2/n$` in single dollar signs which will produce $\sum_{i=1}^n(X_i-\bar{X})^2/n$ in a paragraph. Or, to have your TeX equation appear on a separate line you enclose it in double dollar signs (e.g. `$$\frac{1}{n}\sum_{i=1}^n(X_i-\bar X)^2.$$`) which will produce

$$\frac{1}{n}\sum_{i=1}^{n}(X_i-\bar{X})^2.$$

Alternately, you can use `\(` and `\)` to enclose your TeX equation commands instead of single dollar signs (and `\[` and `\]` instead of double dollar signs).

Of course, whether rendering your final document in HTML, PDF, MS Word etc. the translation of the math you authored using TeX/LaTeX will be handled transparently using the underlying universal document converter *pandoc* (http://pandoc.org) when you knit your document.

Objects such as $p(A\cap B)$ and $p(A\cup B)$ can be typeset via `$p(A\cap B)$` and `$p(A\cup B)$`, or that $p(\bar{A}|B)$ can be typeset as `$p(\bar A|B)$` and so on.

Objects such as $\binom{n}{x}\pi^x(1-\pi)^{n-x}$ can be typeset via `${n \choose x}\pi^{x}(1-\pi)^{n-x}$`, and you can compute actual results inline *and* typeset the mathematics such as $Pr(X=x)=\binom{5}{2}0.4^2(1-0.4)^3=0.3456$ via `$Pr(X=2)={5 \choose 2}0.4^{2}(1-0.4)^{3}= \`r dbinom(2,5,.4)\` $`.

The following link will give you some helpful pointers for typesetting mathematics using TeX/LaTeX: Link to LaTeX/Mathematics.

Troubleshooting and Tips

1. If you are having trouble with TeX, R, RStudio, or git after you installed them, one reason for this could be that you did not use the default paths for the installation. If you overrode the defaults and experience trouble, try properly removing and re-installing *this time using the defaults*
2. Authoring in Markdown can take a bit of getting used to, particularly when making bulleted or numbered lists (which you will likely use for your assignment)
    - You must start a numbered list in column 1, with e.g. the number 1 followed by a period, e.g. `1.`
    - If you want text, an R code chunk, or anything else to line up properly under `1.` then everything on the lines that follow *must* be indented 4 spaces (i.e. start in column 5)

- If you use the tab key to insert spaces, by default a tab takes you only 2 spaces so you need to tab twice to get 4 spaces (or you can change the default by navigating **Tools -> Global Options -> Code -> Tab Width** and change 2 to 4)
- If you have a sublist (e.g. after `1.` you wish `(a)` to be properly indented using Markdown) your `(a)` must start in column 5, and again if you want text, an R code chunk, or anything else to line up properly under `(a)` everything *must* be further indented another 4 spaces (i.e. start in column 9)
- Note that you can select multiple lines of code then hit tab once or twice and it will indent all of the lines that you selected
- If you have an item or subitem (e.g. `1.` or `(a)`) with no text appearing after the item, then indented R code chunks that immediately follow on the next line might not display properly (e.g. where you have indented the entire chunk including the backticks); one simple solution is to add some descriptive text like `1. Answer` and `(a) R Code` and so forth
- If you create a list and e.g. `(a)` does not appear, it could be because you have text following `(a)` that has no space as in `(a)my text` - a space is needed
- Spacing and linebreaks are needed for proper formatting of R Markdown documents, so if things do not render as you expect try adding e.g. a blank line etc.

3. When knitting HTML code the default is to open a new window to show the output preview. If instead you wish the output preview to appear in the viewer pane (lower right corner of RStudio by default), **Tools -> Global Options -> R Markdown -> Show Output Preview in Viewer Pane**
4. Sometimes things get messed up and you need to clean up intermediate files generated while knitting (e.g. your code looks fine but is producing an error). To do so click on the triangle on the knit icon in the editor pane (upper left by default) and clear the knitr cache i.e. **knitr -> Clear Knitr Cache**
5. Sometimes people have old versions of R and RStudio lingering on their system from previous courses and things are not working as expected... it never hurts to make sure you have the latest versions installed (and also update all packages on your system via the update button on the **Packages -> Update** tab which appears in the lower right pane by default)
6. Note that if you use some math environments, e.g. `\begin{align}\beta&=1\\ \alpha&=0\end{align}` and so forth, you retain HTML and PDF compatibility but may lose MS Word compatibility; to regain MS Word compatibility (and lose PDF though retain HTML compatibility) R Markdown needs a hint, and if you enclose your `\begin{align}\beta&=1\\ \alpha&=0\end{align}` in double dollar signs as in `$$\begin{align}\beta&=1\\ \alpha&=0\end{align}$$` this appears to be sufficient
7. Your R Markdown file needs to have the extension `Rmd` (don't create arbitrarily named files)
8. If you are having difficulty reading data files that are located in a different directory from your R Markdown file, simply place the `Rmd` file and data file in the same directory
    - Alternately, after changing your working directory (**Session -> Set Working Directory**), from the *console* pane (lower left by default) type `getwd()` which should reveal the directory where your file is (e.g. `"/foo"`), and then enter the line `setwd("/foo")` in the line just above your call to `read.table()`
    - Alternately, append this directory to the call where you read the data (e.g. instead of `read.table("filename")` use`read.table("/foo/filename")`
    - For the hard core among you, you could modify the `root directory` via adding an R code chunk at the beginning of your document adding the R code `opts_knit$set(root.dir = "/foo")`
9. Chunks of R code begin and end with three backticks, and there *must* be a blank line between two consecutive chunks of R code (i.e. you can't have three backticks at the end of one code chunk *touch* the three backticks of the next line)

10. Your R code must reside inside an R code chunk (see the **Insert** button in the editor pane, top left by default) in order for it to be processed as R code (otherwise R Markdown will think it is text)
11. If a hint is provided preceded by a question mark as in `?fivenum` this means to enter `?fivenum` in the console which will pull up help for the R function `fivenum`
12. Typeset mathematics is *not* to be placed inside of R code chunks (it will throw an error since the dollar sign has a special meaning in R) nor are R Markdown tables and other such things
13. The default editor settings are to *insert matching parentheses and quotes*; if you find that this intrudes on your workflow, you can disable it via **Tools -> Global Options -> Code** then uncheck **Insert matching parens/quotes**
14. Markdown does not like spaces immediately following display math, so use `$$\alpha$$` rather than `$$ \alpha $$`