

Intro to R for Decision Modeling

Getting Ready

SickKids and DARTH

Change `eval` to `TRUE` if you want to knit this document.

1. Introduction to R

R is a specialized program for data manipulation, statistical analysis, plotting and programming. R has several advantages, including:

- R makes it easy to store and analyze large dataset,
- R can perform data cleaning and ensure that these data cleaning steps are reproducible,
- R includes a large, consistent, integrated collection of tools for data analysis,
- R can create publication-quality graphics,
- R can integrate text with data analysis to create Word documents and facilitate manuscript development,
- R is based on a programming language that allows you to develop your own personalized methods to perform analysis,
- R has a large number of user-developed *packages* that can perform specific types of analyses, which can speed up the process of analysis significantly. These packages are developed by R users to help researchers use novel methods for presentation and analysis.

2. Introduction to RStudio

RStudio is a user-friendly interface for R that we will be using throughout this course. Within RStudio, you will see that the screen is divided into four parts:

- The top right hand corner contains the *workspace*. The workspace is your control centre and gives an at-a-glance overview of what has been done so far in your R session.
- The bottom right hand corner is used to load packages, view plotted figures and look at help files.
- On the bottom left is R itself - the console - this is the machine that will crunch your data. All errors will be displayed here.
- If a file is open in RStudio a text editor will appear in the top-left corner.

3. Introduction to R Markdown

R Markdown is a convenient way to keep a record of all the analysis you have done. R Markdown allows you to include text, R code and output in your document. Once you are happy with your analysis you can then output (or *knit*) your document to a PDF, Word document, or HTML page. This means you can write your manuscript, perform your analysis and output tables and graphics all within the same program and then create a Word document or PDF report or a manuscript to submit to a journal.

R markdown documents are stored by default by R using an `.rmd` extension. This document is an example of such an R Markdown document! To create a new R Markdown document in RStudio go to **File -> New File -> R Markdown**. A dialog box will open where you can name your `.rmd` file and specify the file format you would like to use as your knitted output (e.g. `.doc`, `.pdf` or HTML).

As you can see at the top of this document, every R Markdown document starts with a header surrounded by `---` to specify the title, document format, and output type. This information is called *metadata*.

Regular text (like this) can be typed directly into the document while R code is typed into and run from *code chunks*. The following section is an R Markdown *code chunk* that calculates $2 + 2$:

```
# test
2 + 2

message("a")
```

You can include as many *code chunks* as you want in your documents so you can intersperse the text of your manuscript with the data analysis. You chunks are defined using the two commands above that designate the beginning and end of your chunk. You can insert a new chunk using the **Insert** button at the top of this window or writing the commands to define the beginning and end of the chunks yourself.

You can run the code in a *code chunk* by clicking the **little green arrow** on the top right corner of the *code chunk*. You can run the whole chunk also pressing CTRL + SHIFT + ENTER. To only run specific lines of code in a *code chunk*, you can select them and press CTRL + ENTER.

Try running the code in the above *code chunk*.

It is good practice to insert *comments* in your R code to explain what analysis you are doing. This makes it easier to navigate your analysis (and remember what you were doing!). You can make a *comment* anywhere in the *code chunk* by inserting a `#` before your “comment”. In the above *code chunk*, the comment highlighted that we were doing a “test”. There is a fine balance between comments within a chunk and text outside it. The amount of commenting is dependent on what you are aiming to do with the final document. If you are writing a manuscript then the text should include *only* the information you wish to include in the manuscript and the comments should be used to explain what you are doing in the code. If you are using R Markdown to give a commented version of your data analysis then you can use the text to describe your analysis and minimize the number of comments within the code.

When you have more *code chunks* in your R Markdown document, you may need more flexible ways to run code and view outputs. At the top right corner of this window, you can see a green arrow pointing towards a button that says **“Run”**. If you click on this button, you will find ways to run multiple *code chunks* in your document. Specifically, if you put your cursor in a block of text, you can **Run All Chunks Above** or **Run All Chunks Below** your cursor. You can also **Run All** the code chunks in your document. There are several keyboard shortcuts that can be used to run chunks in an R Markdown document. A great resource for those is the following webpage: <https://rmd4sci.njtierney.com/keyboard-shortcuts>.

To knit the R Markdown document go to **File -> Knit Document** or press **Knit** at the top left-hand corner of this window. Try knitting this R Markdown document to a Word document.

To knit a R Markdown document to a PDF document, you will need to download MiKTeX on your machine first. It is an up-to-date implementation of TeX/LaTeX and related programs. MiKTeX can be downloaded from the following webpage: <https://miktex.org/download>.

Throughout this course, we will providing all the materials as R Markdown documents.