

Working with RStudio

Michael Friendly

Psych 6135

<http://euclid.psych.yorku.ca/www/psy6135/>

Getting started: Tools

- To profit best, you need to install both R and R Studio on your computer



The basic R system: R console (GUI) & packages

Download: <http://cran.us.r-project.org/>

Add my recommended packages:

source("http://friendly.apps01.yorku.ca/psy6140/R/install-hebb-pkgs.R")



The R Studio IDE: analyze, write, publish

Download:

<https://www.rstudio.com/products/rstudio/download/>

Add: R Studio-related packages, as useful



R package tools



Data prep: Tidy data makes analysis and graphing much easier.

Packages: [tidyverse](#), comprised of: [tidyr](#), [dplyr](#), [lubridate](#), ...



R graphics: general frameworks for making standard and custom graphics

Graphics frameworks: base graphics, [lattice](#), [ggplot2](#), [rgl](#) (3D)

Application packages: [car](#) (linear models), [vcd](#) (categorical data analysis), [heplots](#) (multivariate linear models)



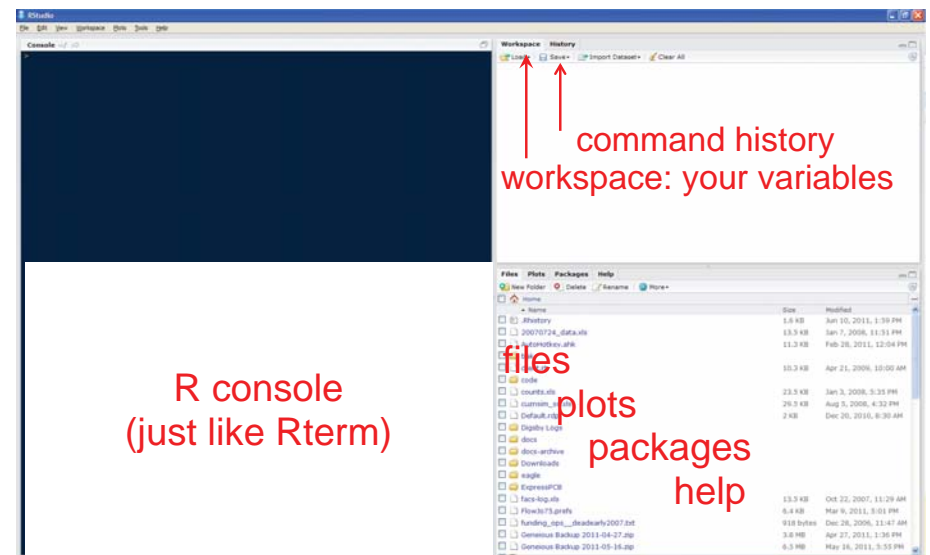
Publish: A variety of R packages make it easy to write and publish research reports and slide presentations in various formats (HTML, Word, LaTeX, ...), all within R Studio



Web apps: R now has several powerful connections to preparing dynamic, web-based data display and analysis applications.



Getting started: R Studio



R Studio navigation

R folder navigation commands:

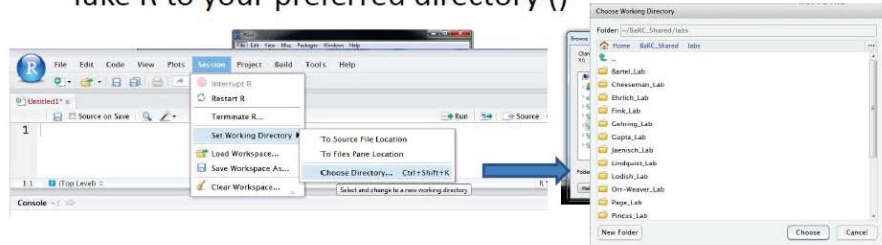
- Where am I?


```
> getwd()
[1] "C:/Dropbox/Documents/6135"
```
- Go somewhere:


```
> setwd("C:/Dropbox/")
> setwd(file.choose())
```

R Studio GUI

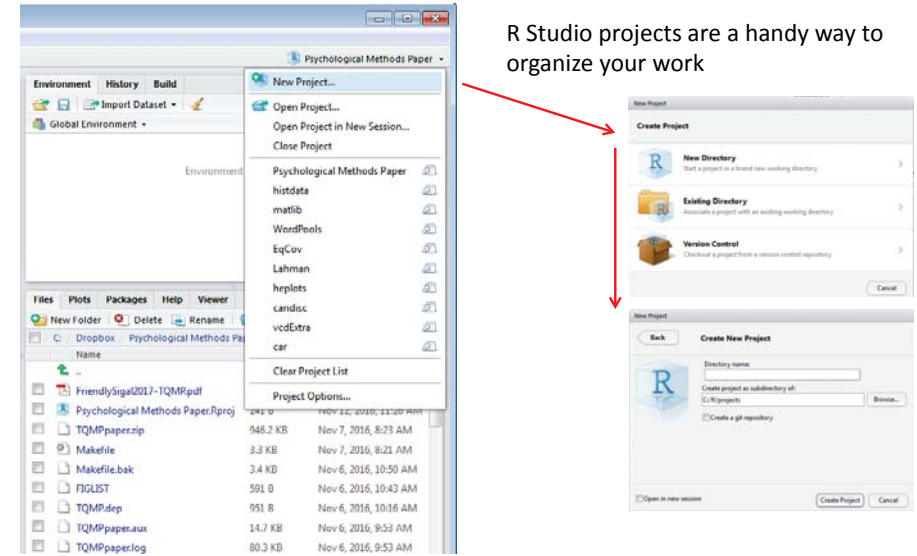
- Take R to your preferred directory ()



5

R Studio projects

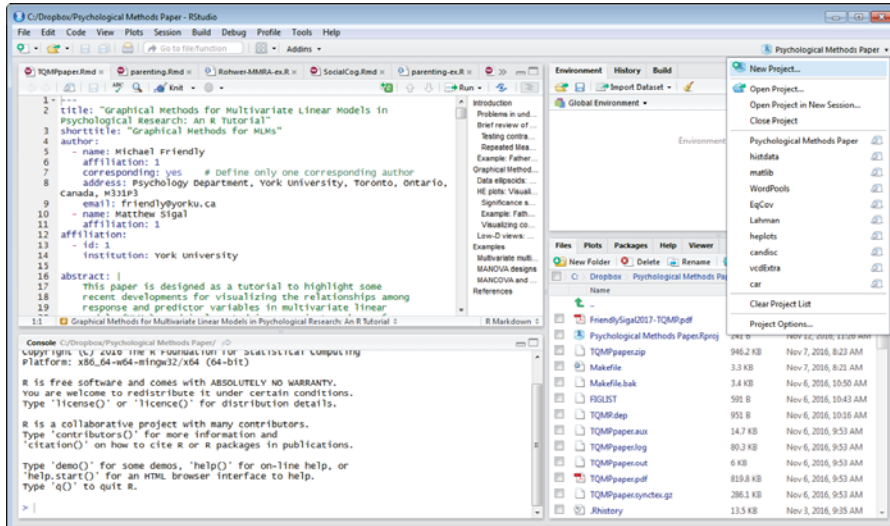
R Studio projects are a handy way to organize your work



6

R Studio projects

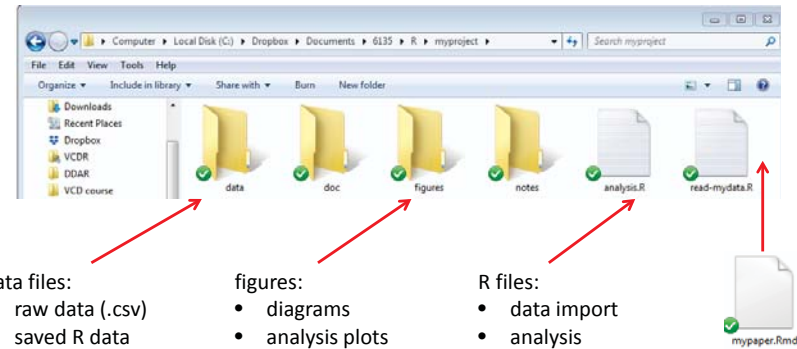
An R Studio project for a **research paper**: R files (scripts), Rmd files (text, R "chunks")



7

Organizing an R project

- Use a separate folder for each project
- Use sub-folders for various parts



data files:

- raw data (.csv)
- saved R data (.Rdata)

figures:

- diagrams
- analysis plots

R files:

- data import
- analysis

Write up files will go here (.Rmd, .docx, .pdf)

8

Organizing an R project

- Use separate R files for different steps:
 - Data import, data cleaning, ... → save as an RData file
 - Analysis: load RData, ...

read-mydata.R

```
# read the data; better yet: use RStudio File -> Import Dataset ...
mydata <- read.csv("data/mydata.csv")

# data cleaning ....

# save the current state
save("data/mydata.RData")
```

9

Organizing an R project

- Use separate R files for different steps:
 - Data import, data cleaning, ... → save as an RData file
 - Analysis: load RData, ...

analyse.R

```
# analysis
load("data/mydata.RData")

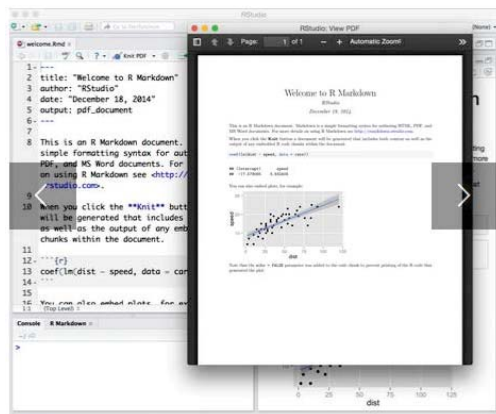
# do the analysis – exploratory plots
plot(mydata)

# fit models
mymod.1 <- lm(y ~ X1 + X2 + X3, data=mydata)

# plot models, extract model summaries
plot(mymod.1)
summary(mymod.1)
```

10

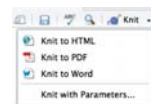
Reproducible analysis & reporting



R Studio, together with the knitr and markdown packages provide an easy way to combine writing, analysis, and R output into complete documents

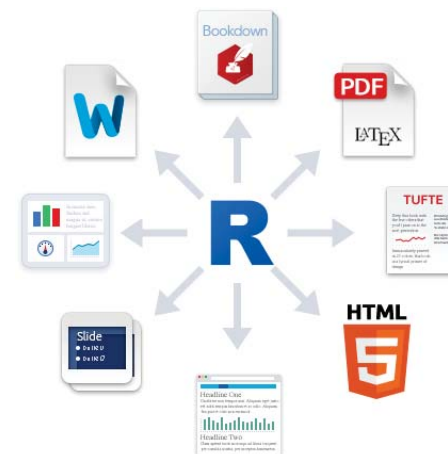
.Rmd files are just text files, using rmarkdown markup and knitr to run R on “code chunks”

A given document can be rendered in different output formats:

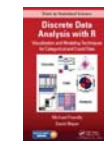


11

Output formats and templates



The integration of R, R Studio, knitr, rmarkdown and other tools is now highly advanced.



My last book was written entirely in R Studio, using .Rnw syntax → LaTeX → PDF → camera ready copy



The ggplot2 book was written using .Rmd format.

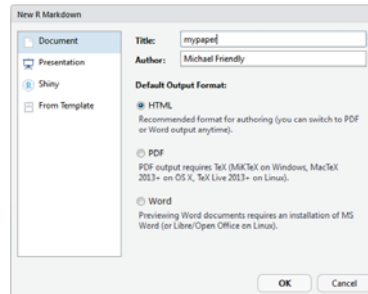
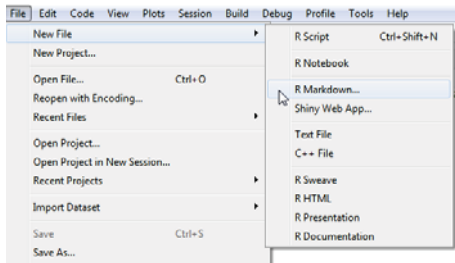
The [bookdown](#) package makes it easier to manage a book-length project – TOC, fig/table #s, cross-references, etc.

Templates are available for APA papers, slides, handouts, entire web sites, etc.

12

Writing it up

- In R Studio, create a .Rmd file to use R Markdown for your write-up
 - lots of options: HTML, Word, PDF (needs LaTeX)



13

Writing it up

- Use simple Markdown to write text
- Include code chunks for analysis & graphs

mypaper.Rmd, created from a template

```
1 ---
2 title: "mypaper"
3 author: "Michael Friendly"
4 date: "January 29, 2018"
5 output: html_document
6 ---
7
8 {r Setup, include=FALSE}
9 knitr::opts_chunk$set(echo = TRUE)
10
11 ## R Markdown
12
13 This is an R Markdown document. Markdown is a simple formatting
14 details on using R Markdown see http://rmarkdown.rstudio.com.
15
16 when you click the knit button a document will be generated
17 R code chunks within the document. you can embed an R code chunk
18
19 {r cars}
20 summary(cars)
21
22 ## Including Plots
23
24 you can also embed plots, for example:
25
26 {r pressure, echo=FALSE}
27 plot(pressure)
28
```

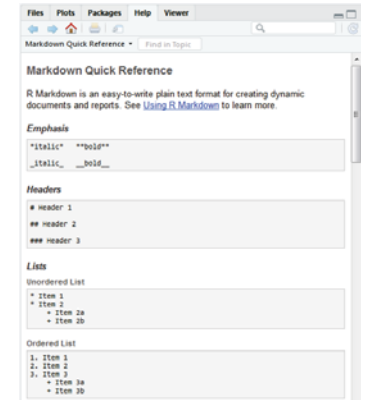
yaml header

Header 2

output code chunk

plot code chunk

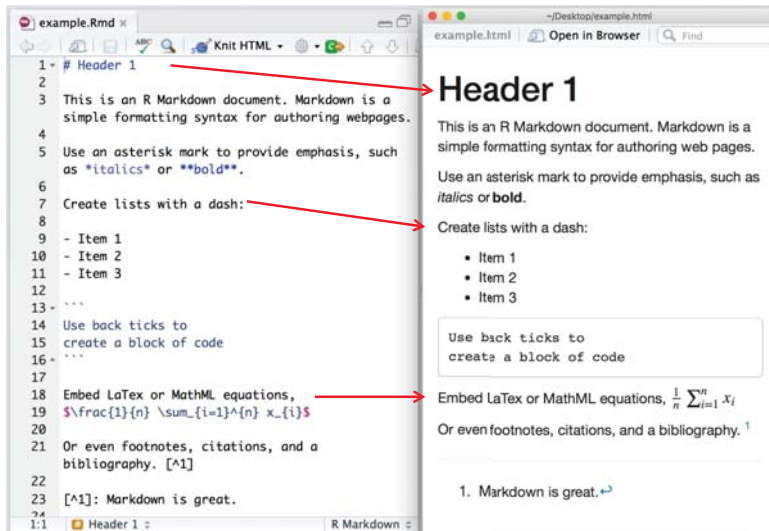
Help -> Markdown quick reference



14

rmarkdown basics

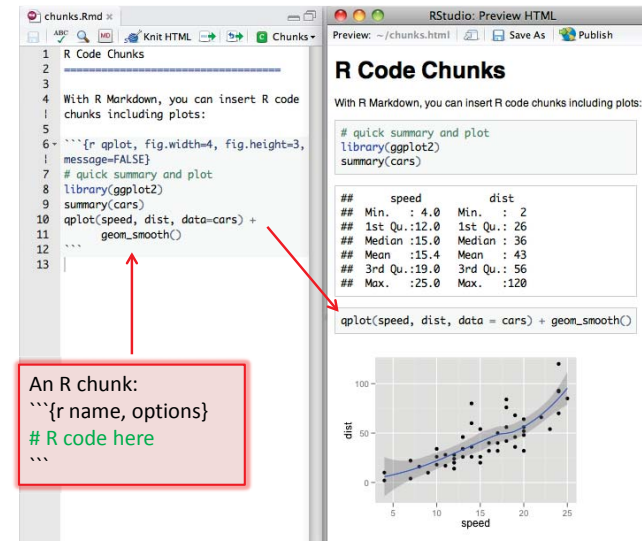
rmarkdown uses simple formatting for all standard document elements



15

R code chunks

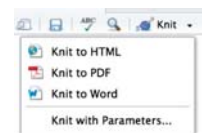
R code chunks are run by `knitr`, and the results are inserted in the output document



An R chunk:
`{r name, options}`
`# R code here`
`}`

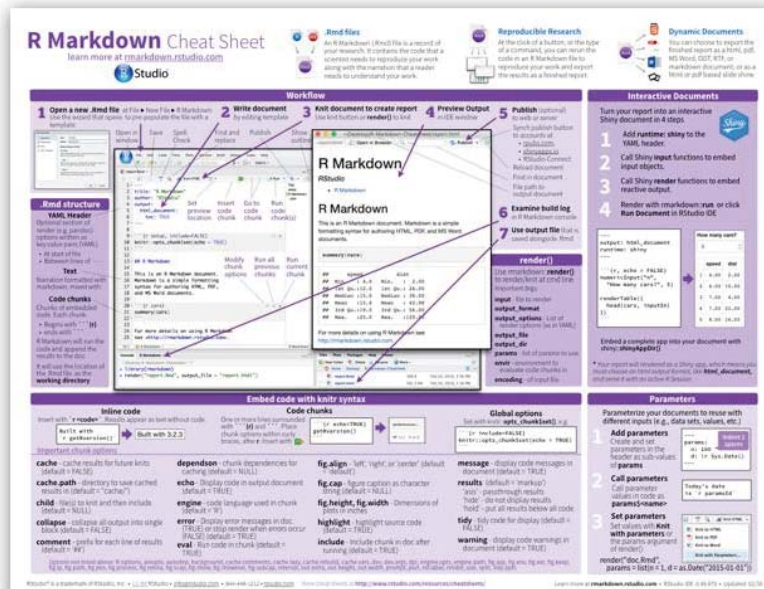
There are many options for controlling the details of chunk output – numbers, tables, graphs

Choose the output format:



16

The R Markdown Cheat Sheet provides most of the details
<https://www.rstudio.com/wp-content/uploads/2016/03/rmarkdown-cheatsheet-2.0.pdf>



R notebooks

Often, you just want to “compile” an R script, and get the output embedded in the result, in HTML, Word, or PDF. Just type **Ctrl-Shift-K** or tap the **Compile Report** button

