MONASH University

MONASH BUSINESS SCHOOL

Introduction to Data Analysis and Visualisation using R

Professor Di Cook, Econometrics and Business Statistics

Workshop for the Institute for Safety, Compensation and Recovery Research

GROUP OF EIGHT AUSTRALIA

AACSB ACCREDITED EFMD EQUIS ACCREDITED ASSOCIATION AMBA ACCREDITED

Outline

- Session 1: Motivation, why and how to think about data, and getting started with R
- Session 2: Making basic plots, grammar of graphics, good practices
- Session 3: Wrangling your data into shape for analysis
- Session 4: Advanced graphics, layering, using maps

Session 1

Motivation, why and how to think about data, and getting started with R

What is exploratory data analysis?

- EDA is concerned about **letting the data speak**, and discovering what is in the data as opposed to predicting from the data
- Initial data analysis is a part of EDA, where data quality and model assumptions are checked using descriptive statistics, prior to modeling
- EDA complements model building: **"The combination of some data and an aching desire for an answer does not ensure that a reasonable answer can be extracted from a given body of data"** Tukey, 1986.

Examples

These are two examples of data sets that I've analysed in recent years, and learned a lot by making plots.

- Education: Every four years students across the globe are tested on their math, reading and science skills and surveyed about their educational experience and social environment, as part of assessing workforce readiness of teenagers. `http://pisa2012.acer.edu.au`
- Climate: Monitors and sensors are located across the globe measuring aspects of the environment, e.g. Scripps Inst. of Oceanography

The data scan be pulled from the web, and the code that produced the plots in these slides is in the `.Rmd` version, so that you can reproduce this work yourself.

Math Gender Gap

The Sydney Morning Herald
National

work relax eat sleep

Investigations Interactives Health Education Public Service News World War 1 Centenary Clique Photos Photo Galleries

You are here: Home » National » Education »

Search here... National Search

Numbers point to maths 'gap'

May 2, 2011

Read later

Caroline Milburn

Tweet Share 0 Share 0 Share Pin it submit

Email article Print Reprints & permissions

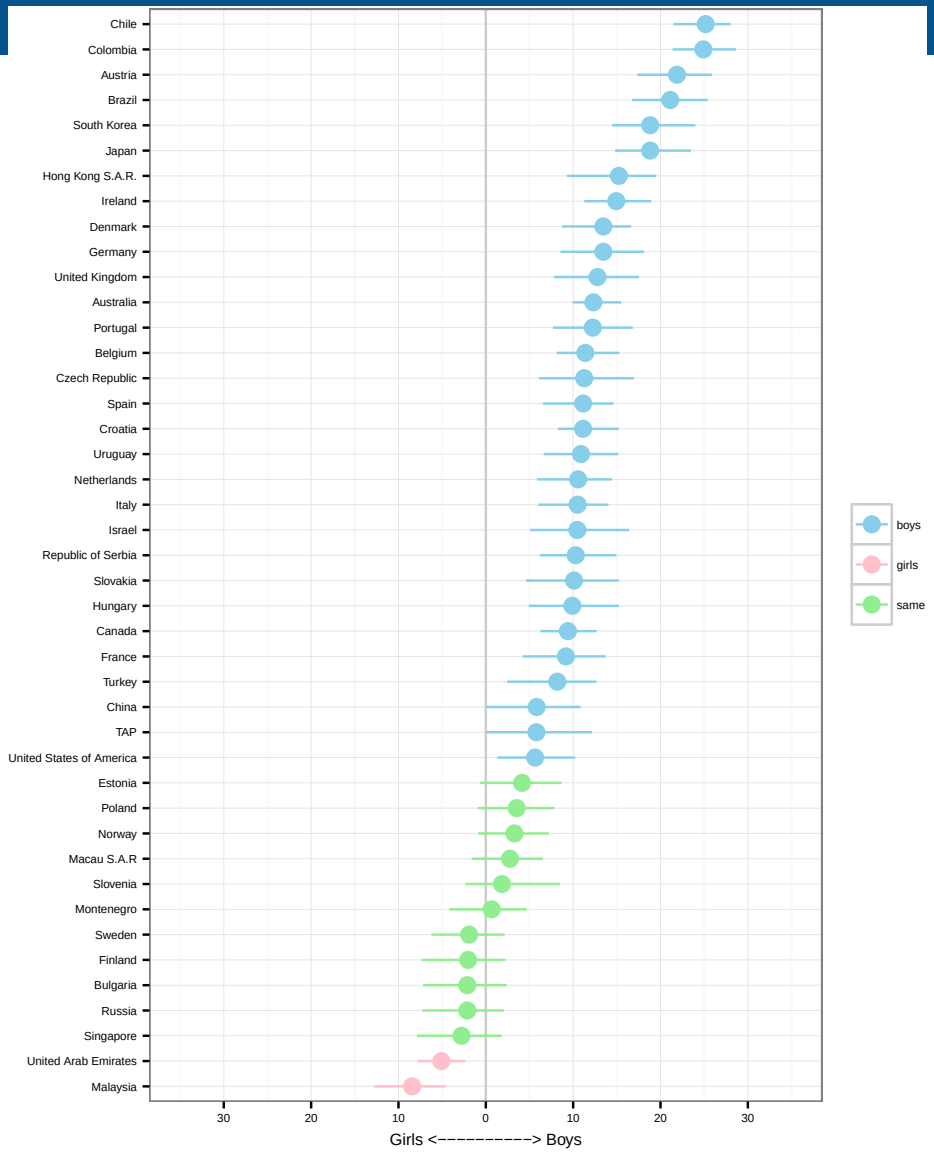
Australia is one of the few countries with a maths gap in favour of boys.

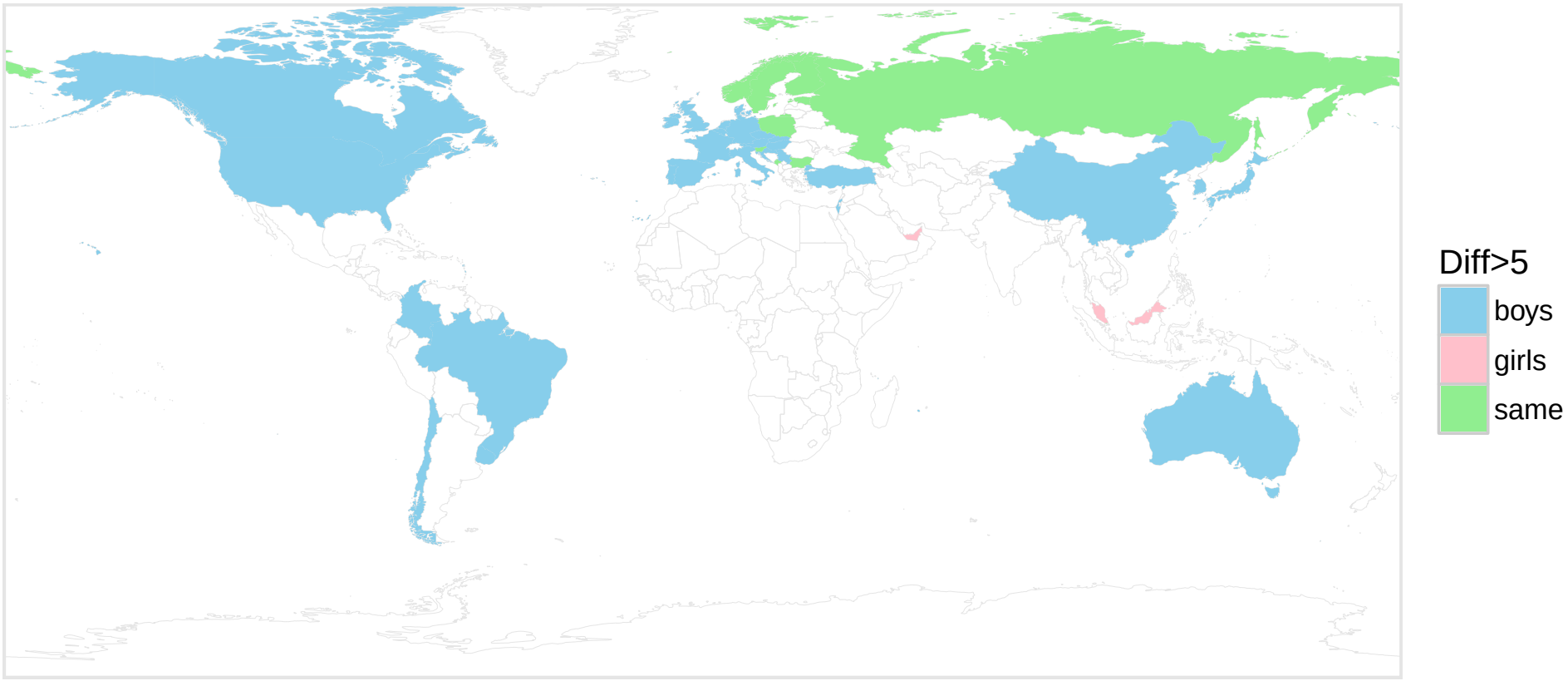
GIRLS are performing at much lower levels in maths than boys — and the gap is widening as students progress through school.

A study of NAPLAN numeracy test results for students in years 3, 5, 7 and 9, to be presented

"Girls are performing at much lower levels in maths than boys - and the gap is widening as students progress through school."
SMH, 2011

Figure 1:





What's the deal about carbon dioxide?

The Sydney Morning Herald
Environment

DANDENONG
WORLD FARE

UN Climate Conference | Weather | Climate Change | Whale Watch | Animals | Conservation | Energy Sn

You are here: Home » Environment »

CSIRO team's study erodes credibility of key soil carbon model

November 1, 2015 Read later

Peter Hannam
Environment Editor, The Sydney Morning Herald
View more articles from Peter Hannam
Follow Peter on Twitter | Follow Peter on Google+ | Email Peter

Tweet | Share 459 | Share 11 | Share | Pin it 1 submit
Email article | Print | Reprints & permissions

Sydney Barrister Peter King argues a royal commission into banks' treatment of farmers is needed. *Photo: Jessica Shapiro*

Australia's method of measuring how much carbon is being stored in its soil is flawed, undermining the credibility of government programs to pay farmers to sequester the climate change inducing element, a new study by CSIRO researchers has found.

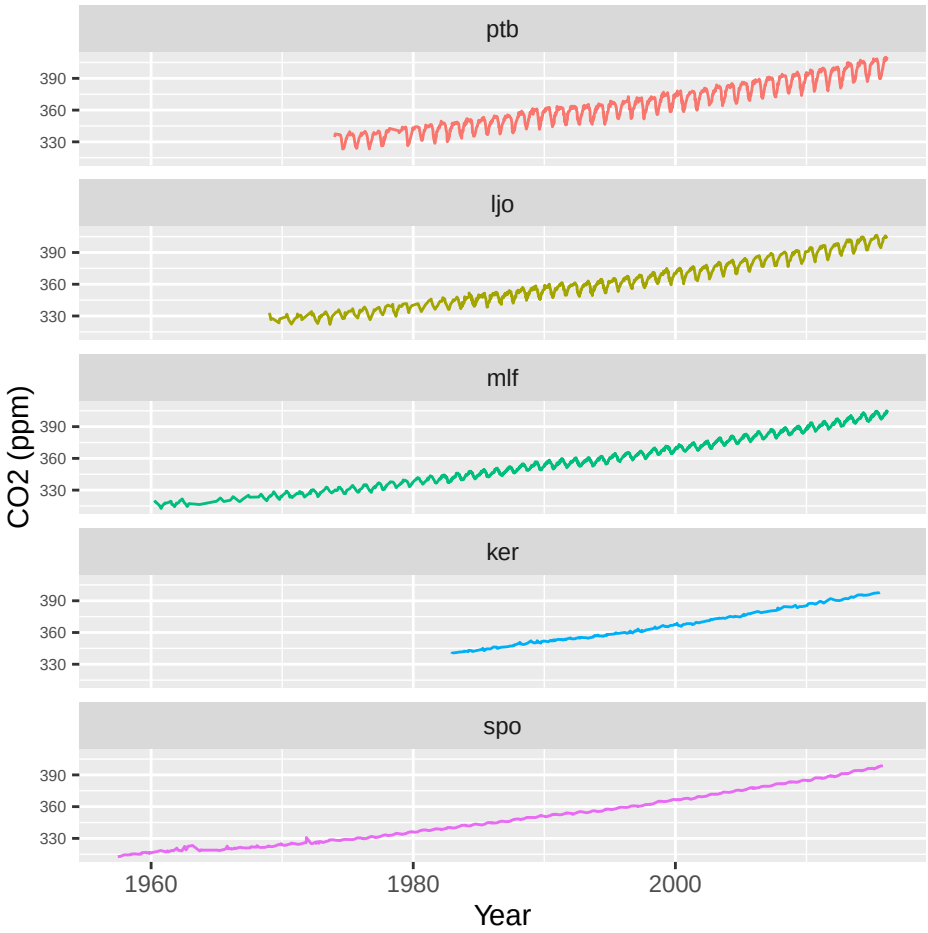
"Australia's method of measuring how much carbon is being stored in its soil is flawed, undermining the credibility of government programs to pay farmers to SEQUESTER the climate change inducing element."

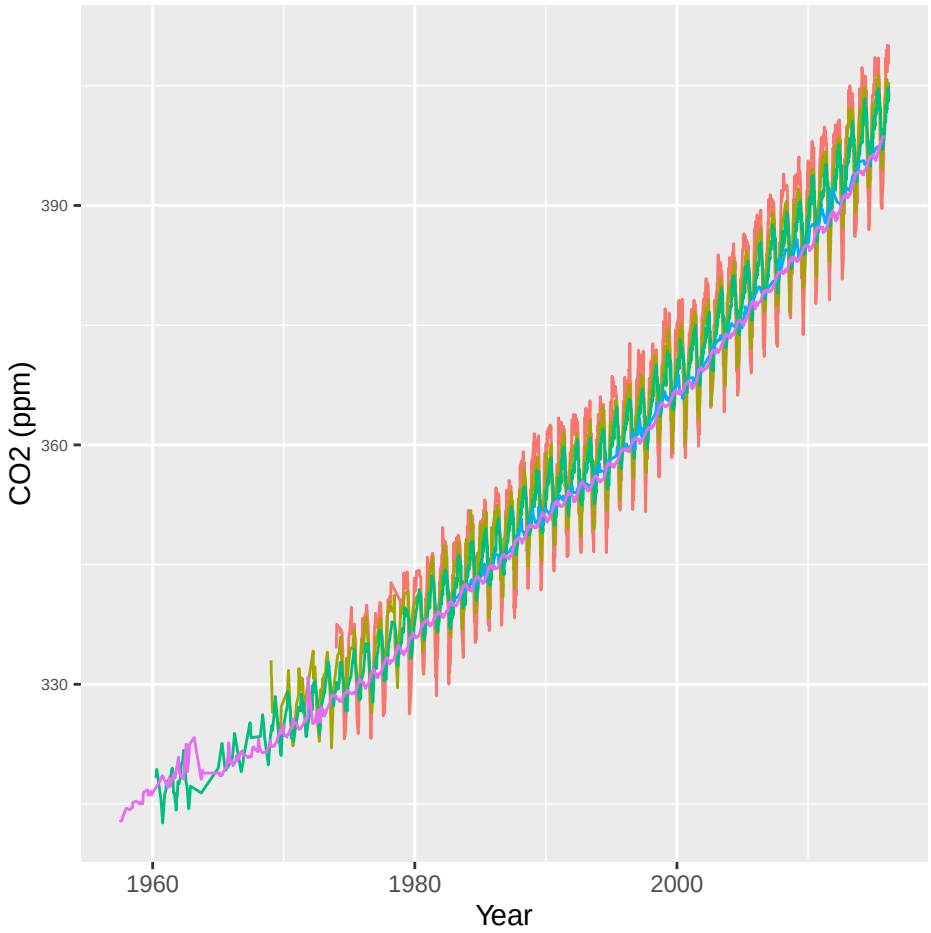
Figure 2:

- "Scientific consensus states that carbon emissions must be reduced by 80% by 2050 to avoid temperature rise of more than 2°C." Carbon Neutral
- Carbon offsets: Carbon offsetting is the use of carbon credits to enable businesses to compensate for their emissions.
- Kyoto protocol in 1992, attempt to get international cooperation to reduce emissions.

Carbon dioxide data

- Data is collected at a number of locations world wide.
- See Scripps Inst. of Oceanography
- Let's pull the data from the web and take a look ...
-
- Recordings from South Pole (SPO), Kermadec Islands (KER), Mauna Loa Hawaii (MLF), La Jolla Pier, California (LJO), Point Barrow, Alaska (PTB).





What do we learn?

- CO_2 is increasing, and it looks like it is exponential increase. **I really expected that the concentration would have flattened out with all of the efforts to reduce carbon emissions.**
- The same trend is seen at every location - REALLY? Need some physics to understand this.
- Some stations show seasonal pattern - actually the more north the more seasonality - WHY?

These slides

- This is a "live" document
- Code and explanations together
- Run the software to make the calculations on the data, and produce nice presentation, or Word or pdf or html document

(Slides and material for this workshop can be found at http://dicook.github.io/ICSRR.)

Why R?

"R has become the most popular language for data science and an essential tool for Finance and analytics-driven companies such as Google, Facebook, and LinkedIn." Microsoft 2015

R is . . .

- **Free** to use
- **Extensible** Over 7300 user contributed add-on packages currently on `CRAN`! More than 10000 on `github.com`
- **Powerful** With the right tools, get more work done, faster.
- **Flexible** Not a question of *can*, but *how*.
- **Frustrating** Flexibility comes at a cost

R does . . .

- Graphics, statistics, machine learning, etc.
- Data acquisition, munging, management
- Literate programming (dynamic reports)
- Web applications

RStudio is . . .

From Julie Lowndes:
If R were an airplane, RStudio would be the airport, providing many, many supporting services that make it easier for you, the pilot, to take off and go to awesome places. Sure, you can fly an airplane without an airport, but having those runways and supporting infrastructure is a game-changer.

The RStudio IDE

- Source editor: (1) Docking station for multiple files, (2) Useful shortcuts ("Knit"), (3) Highlighting/Tab-completion, (4) Code-checking (R, HTML, JS), (5) Debugging features
- Console window: (1) Highlighting/Tab-completion, (2) Search recent commands
- Other tabs/panes: (1) Graphics, (2) R documentation, (3) Environment pane, (4) File system navigation/access, (5) Tools for package development, git, etc

Data analysis cycle

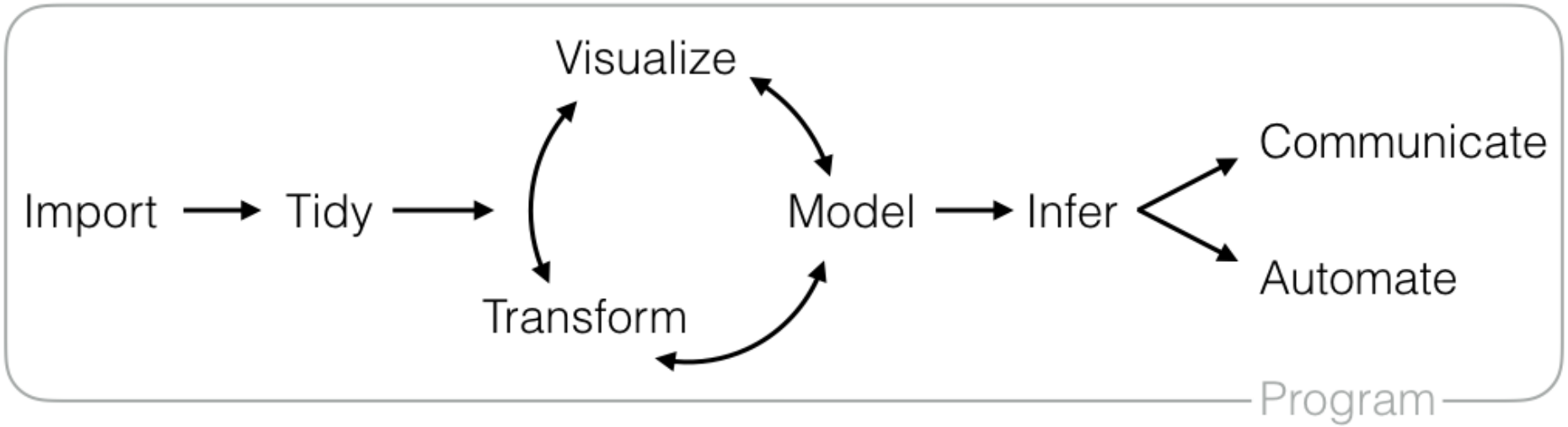


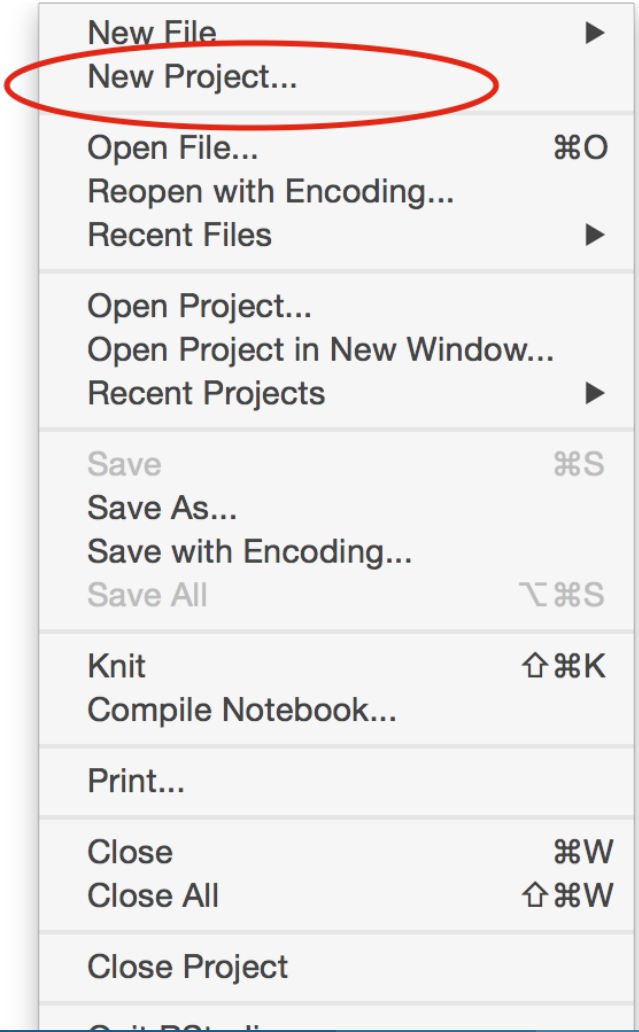
Figure 3:

(Diagram from Hadley Wickham)

Let's get started...

Create a project to contain all of the material covered in this set of tutorials:

- File -> New Project -> New Directory -> Empty Project



Hello R Markdown!

- File -> New File -> R Markdown -> OK -> Knit HTML

~/Desktop/github/local/Monash-R - gh-pages - RStudio Source Editor

Untitled.Rmd

Knit HTML | Run

```
1  ---
2  title: "Untitled"
3  author: "Carson Sievert"
4  date: "February 23, 2016"
5  output: html_document
6  ---
7
8  ```{r setup, include=FALSE}
9  knitr::opts_chunk$set(echo = TRUE)
10 ```
11
12 ## R Markdown
13
14 This is an R Markdown document. Markdown is a simple formatting syntax for authoring
   HTML, PDF, and MS Word documents. For more details on using R Markdown see
   <http://rmarkdown.rstudio.com>.
15
16 When you click the **Knit** button a document will be generated that includes both
   content as well as the output of any embedded R code chunks within the document. You
   can embed an R code chunk like this:
17
18 ```{r cars}
19 summary(cars)
20 ```
21
22 ## Including Plots
23
24 You can also embed plots, for example:
25
26 ```{r pressure, echo=FALSE}
27 plot(pressure)
28 ```
29
30 Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing
   of the R code that generated the plot.
31
```

5:13 | Untitled | R Markdown

~/Desktop/github/local/Monash-R/Untitled.html

Untitled.html | Open in Browser | Find | Publish

Untitled

Carson Sievert

February 23, 2016

R Markdown

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

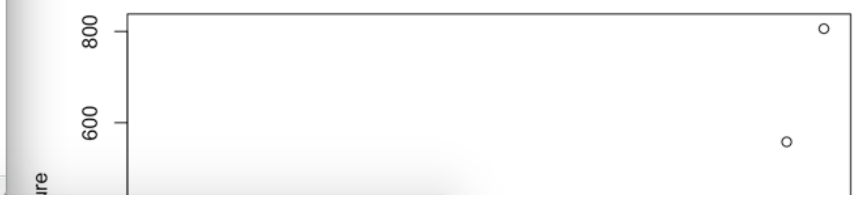
When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

```
summary(cars)
```

```
##      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
```

Including Plots

You can also embed plots, for example:



What is R Markdown?

- From the R Markdown home page:

R Markdown is an authoring format that enables easy creation of dynamic documents, presentations, and reports from R. It combines the core syntax of ***markdown*** *(an easy-to-write plain text format)* ***with embedded R code chunks*** *that are run so their output can be included in the final document. R Markdown documents are fully reproducible (they can be automatically regenerated whenever underlying R code or data changes).*

- RStudio's **cheatsheet** gives a nice, concise overview of its capabilities.
- RStudio's **reference guide** lists its options.

Your turn

Clear the code from the R markdown document, then as work through the **Your turn**s you can add your code blocks and document your answers. The end result will be nicely organised work.

Reading data

- I primarily use the `readr` package for reading data now. It mimics the base R reading functions but is implemented in C so reads large files quickly, and it also attempts to identify the types of variables.
- For many of the your turns, we will use data pulled from the State of New York, Assembled Workers' Compensation Claims: Beginning 2000 which is provided publicly by data.gov.

```
workers <- read_csv(file="data/Assembled_Workers__Compensatio
dim(workers)
#> [1] 50 51
colnames(workers)
#>  [1] "Claim Identifier"
#>  [2] "Claim Type"
#>  [3] "District Name"
#>  [4] "Average Weekly Wage"
#>  [5] "Current Claim Status"
#>  [6] "Claim Injury Type"
```

Some Basics

- Assign values to a name with <- is called **gets**
- `n_max=50` option to the `read_csv` function reads just the first 50 lines
- `dim` reports the dimensions of the data matrix
- `colnames` shows the column names (you can see these by looking at the object in the RStudio environment window, too)
- `$` specify the column to use
- `typeof` indicates the information format in the column, what R thinks
- complex variable names containing spaces, etc, can be used, as long as they are wrapped in single quotes `workers$Claim Type` `

Data types

- `list`'s are heterogenous (elements can have different types)
- `data.frame`'s are heterogeneous but elements have same length
- `vector`'s and `matrix`'s are homogenous (elements have the same type)
  - That's why `c(1, "2")` ends up being a character string.
- `function`'s can be written to save repeating code again and again
- If you'd like to know more, see Hadley Wickham's online chapters on data structures and subsetting

Operations

- Use built-in *vectorized* functions to avoid loops

```
set.seed(1000)
x <- rnorm(4)
x
#> [1] -0.446 -1.206  0.041  0.639
sum(x + 10)
#> [1] 39
```

- R has rich support for documentation, see `?sum`

- Use [to extract elements of a vector.

```
x[1]
#> [1] -0.45
x[c(T, F, T, T, F, F)]
#> [1] -0.446  0.041  0.639
```

- Extract *named* elements with `$`, `[[`, and/or `[`

```
x <- list(
  a = 10,
  b = c(1, "2")
)
x$a
#> [1] 10
x[["a"]]
#> [1] 10
x["a"]
#> $a
#> [1] 10
```

Examining 'structure'

- `str()` is probably my favorite `R` function. It shows you the "structure" of *any* R object (and *everything* in R is an object!!!)

```
str(x)
#> List of 2
#>  $ a: num 10
#>  $ b: chr [1:2] "1" "2"
```

Missing values

- NA is the indicator of a missing value in R
- Most functions have options for handling missings

```
mean(workers$`Birth Year`)
#> [1] NA
mean(workers$`Birth Year`, na.rm=TRUE)
#> [1] 1968
```

Counting categories

- the `table` function can be used to tabulate numbers

```
table(workers$Gender)
#>
#>  F  M  U
#> 15 34  1
table(workers$Gender, workers$`Zip Code`)
#>
#>      11951 11953
#>   F     12     3
#>   M     22    12
#>   U      1     0
```

Getting Help

- Reading documentation only gets you so far. What about *finding* function(s) and/or package(s) to help solve a problem???
- Google! (I usually prefix "CRAN" to my search; others might suggest `http://www.rseek.org/`
- Ask your question on a relevant StackExchange outlet such as `http://stackoverflow.com/` or `http://stats.stackexchange.com/`
- It's becoming more and more popular to bundle "vignettes" with a package (**dplyr** has *awesome* vignettes)

```
browseVignettes("dplyr")
```

Some Oddities

- Yes, + is a function (which calls compiled C code)

```
`+`
#> function (e1, e2)  .Primitive("+")
```

- What's that? You don't like addition? Me neither!

```
"+" <- function(x, y) "I forgot how to add"
1 + 2
#> [1] "I forgot how to add"
```

- But seriously, don't "overload operators" unless you know what you're doing

```
rm("+")
```

Your Turn

1. Read in the first 1000 lines of the NY workers compensation data
2. Tabulate the Claim Types
3. Compute the average and standard deviation of the birth years
4. How many missing values does `Birth Year` have?

Credits

Notes prepared by Di Cook, building on joint workshops with Carson Sievert, Heike Hofmann, Eric Hare, Hadley Wickham.