

# Getting Started with the New Statistics in R

David Erceg-Hurn, Geoff Cumming, Robert Calin-Jageman

2016-07-31

## Overview

R is a popular and powerful free program that can be used to conduct most of the statistical analyses outlined in *Introduction to the New Statistics*. Unlike programs such as SPSS where analyses are usually conducted by clicking on menus, in R analyses are typically performed by typing *commands*.

This document is not a comprehensive guide to using R. Rather, it contains some tips and pointers to websites and other resources that will help you learn how to use R to conduct the analyses outlined in *Introduction to the New Statistics*.

You will notice that some words in this document are a blue colour. These are hyperlinks. If you click on the blue text, you will be redirected to websites that contain information about using R.

This guide is split into three sections. The first section provides some tips about installing and learning the basics of R. If you've never used R before you should read this section - if you already know how to use R you can skip it. The second section provides a brief overview of a new R package, *itns*, that contains most of the datasets used in *Introduction to the New Statistics*. You can use the datasets in the *itns* package to work through the examples covered in the book and the end-of-chapter exercises. The final section provides a brief overview of R packages and functions that can be used to conduct the analyses covered in *Introduction to the New Statistics*.

## Part One - Installing R and Learning the Basics

To install R, visit the [RStudio website and following the installation instructions](#). That webpage also contains links to interactive tutorials for R beginners. The tutorials will help you learn how to perform basic tasks like importing and manipulating datasets. Other useful resources for learning R include:

- [R for Data Science](#) - An online book by Garrett Grolmund and Hadley Wickham that will teach you how to import, tidy, and explore data.
- [Kelly Black's R Tutorial](#) - An introductory tutorial focusing on the basics of R.
- [How to Learn R Blog](#) - A collection of resources that will help you learn R.
- [Quick-R](#) - A website that contains example code for running basic analyses.
- [R Quick Reference Card](#) - A list of key commands built into R.

Also remember that Google is your friend. If you have a question about how to do something in R, it is likely that someone else has already asked the same question and that there is an answer on the Internet. For example if you type 'R how to create a histogram' into Google, you will find many links to webpages showing you the R code that you need to plot a histogram.

In the remainder of the document, we assume that you have a basic understanding of how to use R.

## Part Two - The *itns* Dataset Package

'*itns*' is an R package which contains most of the datasets used in *Introduction to the New Statistics*. The datasets were converted from Microsoft Excel files (found on the book's website) into R data frames. The *itns* package is not yet on CRAN, but can be downloaded from github using the *devtools* package:

```
# install.packages("devtools")
library(devtools)
install_github("gitrman/itns")
```

The table below contains a list of names of the datasets in the package, and the sections of the book where they are mentioned:

#### itns Package Data Frames

Name	Section	Topic
college_survey1	Ch 3 End of Chapter Exercises 2 & 3	Descriptive Statistics & Plots
religious_belief	Ch 3 End of Chapter Exercise 4	Descriptive Statistics & Plots
college_survey1	Ch 5 End of Chapter Exercises 2 & 3	Single Sample Confidence Interval
college_survey2	Ch 5 End of Chapter Exercise 4	Single Sample Confidence Interval
stickgold	Ch 6 End of Chapter Exercise 5	Single Sample Confidence Interval
pen_laptop1	Ch 7.6-7.12	Two Independent Groups
pen_laptop2	Ch 7.36-7.38	Two Independent Groups
anchor_estimate	Ch 7 End of Chapter Exercise 3	Two Independent Groups
clean_moral1	Ch 7 End of Chapter Exercise 4	Two Independent Groups
clean_moral2	Ch 7 End of Chapter Exercise 4	Two Independent Groups
math_gender_iat	Ch 7 End of Chapter Exercise 5	Two Independent Groups
thomason1	Ch 8, 11, 12	Two Dependent Groups, Scatterplots, Regression
thomason2	Ch 8	Two Dependent Groups
thomason3	Ch 8, 12.18	Two Dependent Groups, Regression
emotion_hartrate	Ch 8 End of Chapter Exercise 3	Two Dependent Groups
labels_flavor	Ch 8 End of Chapter Exercise 4	Two Dependent Groups
ma_anchor_adjust	Ch9 End of Chapter Exercise 1	Meta-Analysis
ma_flag_priming	Ch9 End of Chapter Exercise 2	Meta-Analysis
ma_math_gender_iat	Ch9 End of Chapter Exercise 3	Meta-Analysis
ma_power_performance	Ch9 End of Chapter Exercise 4	Meta-Analysis
body_well	Ch 11, 12	Correlation, Regression
exam_scores	Ch 11 End of Chapter Exercise 2	Correlation
sleep_beauty	Ch 11 End of Chapter Exercise 6	Correlation
campus_involvement	Ch 11 End of Chapter Exercise 7	Correlation
home_prices	Ch 12 End of Chapter Exercise 2	Regression
home_prices_holdout	Ch 12 End of Chapter Exercise 2h	Regression
altruism_happiness	Ch 12 End of Chapter Exercise 3	Regression
rattan	Ch 14.10-14.12	One-Way Independent Group Contrasts and Comparisons
organic_moral	Ch 14 End of Chapter Exercise 5	One-Way Independent Group Contrasts and Comparisons
videogame_aggression	Ch 15 End of Chapter Exercise 3	Analysing factorial designs
self_explain_time	Ch 15 End of Chapter Exercise 4	Analysing factorial designs
natsal	Ch 16.11	Robust Methods - Two Independent Groups
dana	Ch 16 End of Chapter Exercise 3	Robust Methods - Two Independent Groups

Once you have installed the package, you can use the `library()` function to load it, `str()` to examine metadata for each data frame, and functions such as `head()` and `tail()` to print the first or last few rows to your screen.

```
library(itns)      # loads the package
str(pen_laptop1)  # displays metadata
```

```
## 'data.frame': 65 obs. of 2 variables:
## $ group : Factor w/ 2 levels "Laptop","Pen": 2 2 2 2 2 2 2 2 2 2 ...
## $ transcription: num 12.1 6.5 8.1 7.6 12.2 10.8 1 2.9 14.4 8.4 ...
```

```
head(pen_laptop1) # prints the first few rows
```

```
## group transcription
## 1 Pen 12.1
## 2 Pen 6.5
## 3 Pen 8.1
## 4 Pen 7.6
## 5 Pen 12.2
## 6 Pen 10.8
```

```
tail(pen_laptop1) # prints the last few rows
```

```
## group transcription
## 60 Laptop 10.3
## 61 Laptop 9.0
## 62 Laptop 12.8
## 63 Laptop 12.0
## 64 Laptop 34.7
## 65 Laptop 4.1
```

To access further details about each dataset, type a question mark and the name of the dataset, for example:

```
?pen_laptop1
```

or access the PDF help file [LINK TO GO HERE](#) on github.

The datasets in the `itns` package can be used to replicate analyses that appear in *Introduction to the New Statistics*, and to work through the book's end-of-chapter exercises using the packages and functions outlined in the next section of this guide.

## Part 3 - Helpful Packages and Functions

Most of the analyses described in *Introduction to the New Statistics* can be conducted using inbuilt R functions, or functions in packages that can be downloaded from CRAN or github. In this section we mention some useful functions, packages, and resources that will help you learn how to use them.

### Basic Descriptive Statistics

Functions to compute basic descriptive statistics are built into R. Some of these are listed in the table below:

#### Functions For Computing Descriptive Statistics

Statistic	Function
Mean	<a href="#">mean()</a>
Median	<a href="#">median()</a>
Minimum	<a href="#">min()</a>
Maximum	<a href="#">max()</a>

Statistic	Function
Variance	<a href="#">var()</a>
Standard Deviation	<a href="#">sd()</a>
Interquartile Range	<a href="#">IQR()</a>
Range	<a href="#">range()</a>
Percentile	<a href="#">quantile()</a>

[summary\(\)](#)

**Useful tutorials** \*[John Quirk's Summary and Descriptive Statistics](#)

### Computing Descriptive Statistics For Multiple groups

There are many ways to do this. We recommend using dplyr package:

dplyr [vignette](#), see the section on *Grouped Operations* using the `group_by()` and `summarise()` functions

Give an example of computing mean and standard deviation.

Other options include using the inbuilt R function [aggregate\(\)](#) or the [doBy package](#).

### Z-Scores

[John Quick's tutorial](#) shows how to use R's inbuilt `scale()` function to compute Z-scores. See also [Seam Dolinar's tutorial](#) on Calculating Z scores and finding tail probabilities.

### Plots

Three graphics systems - base graphics, lattice, and ggplot2.

For new users, would recommend learning ggplot2 first as it is the most flexible.

Deducer has an excellent GUI for ggplot2 that makes it easier to create plots.

[This tutorial](#) provides an overview of the three systems with examples.

Tim Appelhans has a [comprehensive tutorial](#) about creating publication quality graphics using lattice and ggplot2 systems.

he also notes that "google is your friend - for R related questions just type something like this 'r lattice how to change plot background color' The crux here is that you provide both the programming language - R - and the name of the package your question is related to - lattice. This way you will very likely find useful answers (the web is full of knowledgeable geeks)"

Histogram

Dotplot

Best place to start is probably the [Getting Started with Lattice Graphics](#) tutorial by Deepayan Sarkar.

His [Lattice book](#) provides more comprehensive information.

[ggplot2 book website](#) [ggplot2 online documentation](#)

[Winston Chang's R Graphic's Cookbook](#)

[Harvard Introduction to ggplot2](#)

[Zevross ggplot2 tutoial](#)

[ggplot2 tutorial](#)

DataCamp online courses (cost money) <https://www.datacamp.com/courses/data-visualization-with-ggplot2-1>

[RStudio ggplot2 cheatsheet](#)

## Confidence Intervals and Effect Sizes

Cat's eye picture Find tail areas for any T value, for any DF Find critical value of t given df and a tail probability

Calculate MOE and 95% CI on a mean for single sample, using Z or t

## The Normal Distribution and Sampling

Find tail areas for X or Z values Find X given z

## p Values, Null Hypothesis Significance Testing, and Confidence Intervals

Calculate p values for single samples, using z or t

[Kelly Black's tutorial](#) shows how to compute p values using z- or t-distributions.

[Kelly Black's tutorial](#) shows how to calculate confidence intervals for means using normal or t-distributions.

The easiest way to compute confidence intervals and p values is using the `t.test()` function.

ggplot showing CI on a mean?

## t.test() function

The `t.test()` function is built into R. It produces confidence intervals and p-values for single samples, two independent groups, and paired samples.

Give examples using itns data.

`t.test(x, y, conf.level=ci, var.equal = TRUE)` `t.test(x, y, conf.level=ci, var.equal = FALSE)` `t.test(x, y, paired = TRUE)`

## MBESS package

`library(MBESS)` Ken Kelly's [MBESS \(Methods for the Behavioural and Social Sciences\)](#) package.

Includes functions for computing confidence intervals around the standardised mean difference, unstandardised and standardised regression coefficients, reliability coefficients, ANOVA contrasts, R-squared, and more. Also includes functions for power analysis and sample size planning.

Demonstrate some key functions related to book.

CI on difference between means Cohen's d

Effect Sizes and Confidence Intervals Sample Size and Parameter Estimation Sample Size Planning

Any precision for planning functions Any statistical power functions

## WRS and WRS2 packages

### Standardized Paired Difference

Is there any function for this? Or show how to compute by hand and use noncentral T from MBESS CI on mean difference d using SD-average (of pre and post) as standardizer

## Meta-Analysis

There are numerous R packages that can be used to conduct meta-analyses for a wide variety of effect sizes such as means, mean differences, standardized mean differences, proportions, odds ratios, etc. See the [CRAN Meta-Analysis Task View](#) for a comprehensive list of them.

A popular and well documented package for conducting meta-analyses in R is [metafor](#). See the comprehensive [metafor website](#) for more information.

[metagear](#) is a relatively new package which has meta-analytic capabilities as well as functions that help users conduct systematic reviews and generate [PRISMA \(Preferred Reporting Items for Systematic Reviews and Meta-Analyses\)](#) flow charts. [This vignette](#) provides an overview of the metagear package.

Other useful sources of information about conducting meta-analyses in R include:

\* [A.C Del Re's Practical Tutorial](#) on Conducting Meta-Analysis in R using the metafor and MAd packages \* Stephanie Kovalchik's [Tutorial on Meta-Analysis in R](#) from the 2013 useR! Conference \* Schwarzer, Carpenter, and Rucker's [Meta-Analysis with R](#) book \* [R-Studio's tutorial](#) on running meta-analyses in R using the metafor package \* Simon Knight's Guide to Meta-Analysis in R - [part 1](#) and [part 2](#). \* Stephanie Hick's [Easy Introduction to Meta-Analysis in R](#) using the meta package

## multicon plots

egraph() - A function for plotting means as dots and error bars without caps around them catseye() - A function for creating cat's eye plots of group means diffPlot() - A function for creating difference plots for two group (both paired and independent) comparisons

## Open Science and Planning Research

MBESs does this

## Correlation

Basic scatterplot Adding a regression line

CIs on correlations CI on difference between two independent correlations Figure to display difference in two ind corr

## Regression

Calculation and figure for regression line in scatterplot

CI on the regression slope

lm() is the inbuilt R function for ordinary least-squares regression. confint() gives confidence intervals on regression parameters

Also MBESS can do this (aparently) as well as for standardized coefficients.

[Link to graphics.](#)

Show ggplot2 scatterplot with OLS line, maybe also show smoother, mention it can also fit robust and other regression lines.

Also countless other packages for fitting a variety of regression models in R.

Prediction intervals for individual values of Y at particular X values

## Frequencies, Proportions, and Risk

CI for a proportion CI for difference between two independent proportions Figure for CI on a single proportion Figure for difference with CI

ESCI uses Newcombe 1998 methods

Ratio between two variables - frequency tables CI on the difference Chi Square / Phi Coefficients

vcd and vcdExtra packages for visualising categorical data

PropCIs Does CIs for single, independent, and paired proportions. Includes risk score CI.

library(pairwiseCI) # for ARR NHS and RR using Score Method. Uses Prop.diff and Prop.ratio

And chi-square function I used for recent WCBCT analyses.

R manual to accompany Agresti's Categorical Data Analysis 2nd Ed by Laura Thompson [http://www.stat.ufl.edu/~aa/cda/Thompson\\_manual.pdf](http://www.stat.ufl.edu/~aa/cda/Thompson_manual.pdf)

## Extended Designs - One Independent Variable

One way independent groups design CI for planned contrasts, on two means Figures if possible

One way repeated esigns, if possible (beyond scope of book and ESCI)

ANOVA p values

## Extended Designs - Two Independent Variables

ESCI only does 2 x 2 design Means and CI on mean effects CI on single DF interaction (as difference in differences)

Simple main effect

Nice to include two way repeated measures if possible

## Open Science and Future Directions

Trimmed means for two independent means

Vignettes are long form documentation commonly included in packages. Because they are part of the distribution of the package, they need to be as compact as possible. The `html_vignette` output type provides a custom style sheet (and tweaks some options) to ensure that the resulting html is as small as possible. The `html_vignette` format:

- Never uses retina figures
- Has a smaller default figure size
- Uses a custom CSS stylesheet instead of the default Twitter Bootstrap style

## Vignette Info

Note the various macros within the `vignette` section of the metadata block above. These are required in order to instruct R how to build the vignette. Note that you should change the `title` field and the `\VignetteIndexEntry` to match the title of your vignette.

## Styles

The `html_vignette` template includes a basic CSS theme. To override this theme you can specify your own CSS in the document metadata as follows:

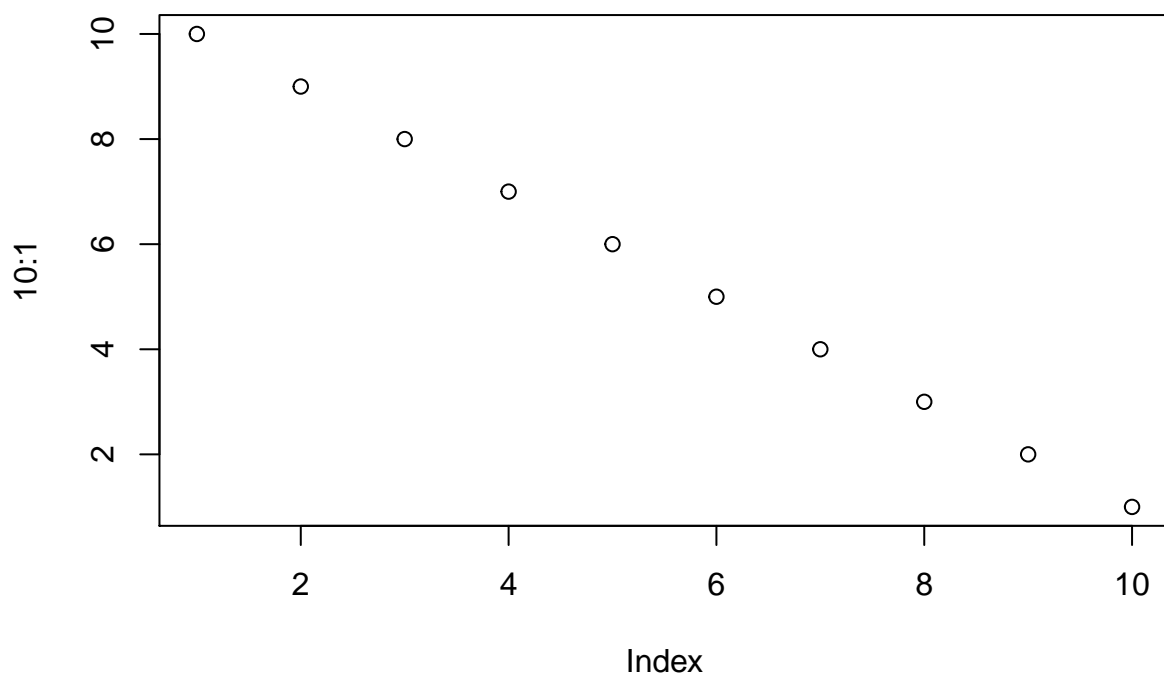
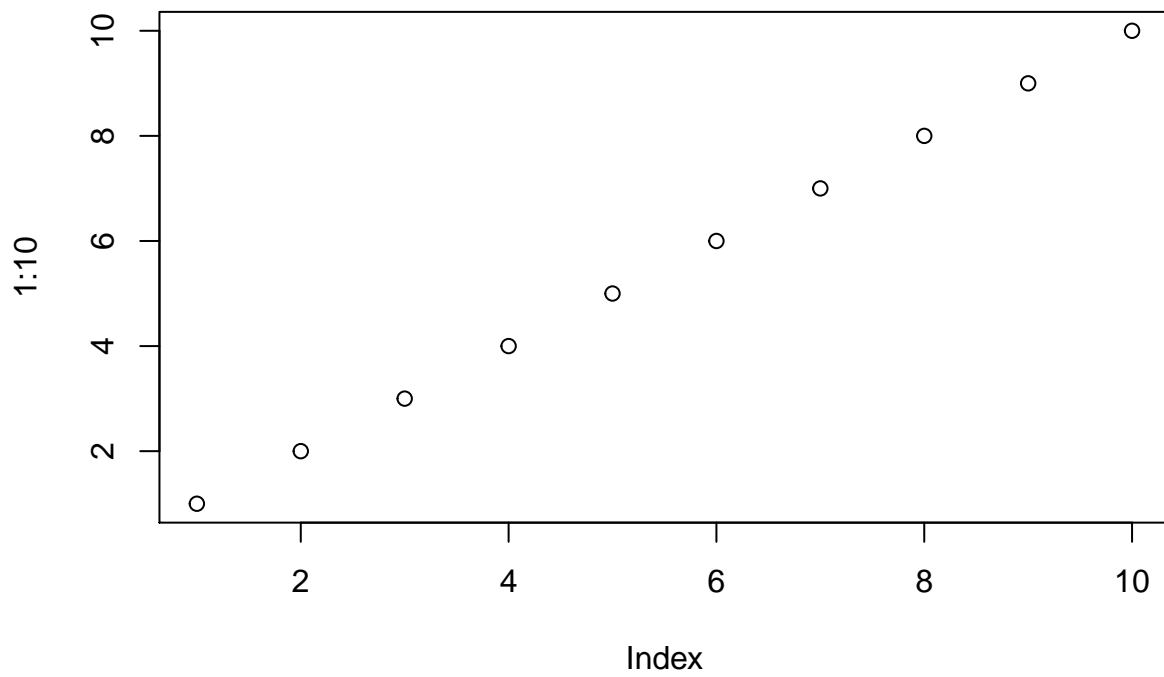
```
output:
  rmarkdown::html_vignette:
    css: mystyles.css
```

## Figures

The figure sizes have been customised so that you can easily put two images side-by-side.

```
plot(1:10)
plot(10:1)
```





You can enable figure captions by `fig_caption: yes` in YAML:

```
output:
  rmarkdown::html_vignette:
    fig_caption: yes
```

Then you can use the chunk option `fig.cap = "Your figure caption."` in **knitr**.

## More Examples

You can write math expressions, e.g.  $Y = X\beta + \epsilon$ , footnotes<sup>1</sup>, and tables, e.g. using `knitr::kable()`.

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160.0	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160.0	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108.0	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258.0	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360.0	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225.0	105	2.76	3.460	20.22	1	0	3	1
Duster 360	14.3	8	360.0	245	3.21	3.570	15.84	0	0	3	4
Merc 240D	24.4	4	146.7	62	3.69	3.190	20.00	1	0	4	2
Merc 230	22.8	4	140.8	95	3.92	3.150	22.90	1	0	4	2
Merc 280	19.2	6	167.6	123	3.92	3.440	18.30	1	0	4	4

Also a quote using `>`:

"He who gives up [code] safety for [code] speed deserves neither." ([via](#))

## Questions for Geoff and Bob

*Datasets I can't find referenced in text*

Name	Section
flag_priming	Ch 7 ???
super_golf	Ch 7 ???
habituation	Ch 8 ???
learning_genes	Ch 8 ???
sensitization	Ch 8 ???
ma_gambler_fallacy	Ch9 ???
ma_anchor_adjust_chicago	Ch9 ???
ma_anchor_adjust_everest	Ch9 ???
ch11_ex7	Ch 11 ???
ch12_ex3	Ch 12 ???
inauthentic	Ch 14 ???
iqboost	Ch 14 ???
blame1	Ch 15 ???
blame2	Ch 15 ???

---

<sup>1</sup>A footnote here.

