Introduction to R for Reproducible Research

Jason M. Bryer, Ph.D.

Excelsior College
www.bryer.org
jason@bryer.org

November 20, 2014
Albany useR Group

Agenda

What is R?

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues...

What is R?

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues...

R provides a wide variety of statistical (linear and non linear modeling, classical statistical tests, time-series analysis, classifcation, clustering, ...) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open Source route to participation in that activity. (R-project.org)

Pros

- FREE! *R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.*

Pros

- FREE! *R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.*
- Available for multiple platforms (i.e. Windows, Mac, Linux).

Pros

- FREE! *R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.*
- Available for multiple platforms (i.e. Windows, Mac, Linux).
- Easily extensible with (currently) over 2,000 packages listed on CRAN.

Pros

- FREE! *R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.*
- Available for multiple platforms (i.e. Windows, Mac, Linux).
- Easily extensible with (currently) over 2,000 packages listed on CRAN.
- Scriptable.

Pros

- FREE! *R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.*
- Available for multiple platforms (i.e. Windows, Mac, Linux).
- Easily extensible with (currently) over 2,000 packages listed on CRAN.
- Scriptable.
- Publication grade graphics.

Pros

- FREE! *R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.*
- Available for multiple platforms (i.e. Windows, Mac, Linux).
- Easily extensible with (currently) over 2,000 packages listed on CRAN.
- Scriptable.
- Publication grade graphics.
- Multiple ways of doing the same thing.

Pros

- FREE! *R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.*
- Available for multiple platforms (i.e. Windows, Mac, Linux).
- Easily extensible with (currently) over 2,000 packages listed on CRAN.
- Scriptable.
- Publication grade graphics.
- Multiple ways of doing the same thing.
- Quickly becoming the de facto standard among statistician.

Cons

- Has a steeper learning curve.

Cons

- Has a steeper learning curve.
- Multiple ways of doing the same thing.

Cons

- Has a steeper learning curve.
- Multiple ways of doing the same thing.
- Can have difficulty with *very* large datasets.

R's Roots... S

- S is a language that was developed by John Chambers and others at Bell Labs.
- S was initiated in 1976 as an internal statistical analysis environment - originally implemented as Fortran libraries.
- Early versions of the language did not contain functions for statistical modeling.
- In 1988 the system was rewritten in C and began to resemble the system that we have today (this was Version 3 of the language). The book Statistical Models in S by Chambers and Hastie (the blue book) documents the statistical analysis functionality.
- Version 4 of the S language was released in 1998 and is the version we use today. The book Programming with Data by John Chambers (the green book) documents this version of the language.

History of S

- In 1993 Bell Labs gave StatSci (now Insightful Corp.) an exclusive license to develop and sell the S language.
- In 2004 Insightful purchased the S language from Lucent for $2 million and is the current owner.
- In 2006, Alcatel purchased Lucent Technologies and is now called Alcatel-Lucent.
- Insightful sells its implementation of the S language under the product name S-PLUS and has built a number of fancy features (GUIs, mostly) on top of it-hence the "PLUS".
- In 2008 Insightful is acquired by TIBCO for $25 million; future of S-PLUS is uncertain.
- The S language itself has not changed dramatically since 1998.
- In 1998, S won the Association for Computing Machinery's Software System Award.

In "Stages in the Evolution of S", John Chambers writes:

> *"[W]e wanted users to be able to begin in an interactive environment, where they did not consciously think of themselves as programming. Then as their needs became clearer and their sophistication increased, they should be able to slide gradually into programming, when the language and system aspects would become more important."*

http://www.stat.bell-labs.com/S/history.html

History of R

- 1991: Created in New Zealand by Ross Ihaka and Robert Gentleman. Their experience developing R is documented in a 1996 JCGS paper.
- 1993: First announcement of R to the public.
- 1995: Martin M?achler convinces Ross and Robert to use the GNU General Public License to make R free software.
- 1996: A public mailing list is created (R-help and R-devel)
- 1997: The R Core Group is formed (containing some people associated with S-PLUS). The core group controls the source code for R.
- 2000: R version 1.0.0 is released.
- 2012: R version 2.15.2 is released on October 31, 2012.
- 2013: R version 3.0 is released on April 3, 2013
- There are now over 6,000 packages listed on CRAN.

What is LaTeX

LaTeX...

- is a document preparation system for high-quality typesetting.

What is LaTeX

LaTeX...

- is a document preparation system for high-quality typesetting.
- is *not* a word processor.

What is LaTeX

LaTeX...

- is a document preparation system for high-quality typesetting.
- is *not* a word processor.
- began in 1985 by Leslie Lampart.

What is LaTeX

LaTeX...

- is a document preparation system for high-quality typesetting.
- is *not* a word processor.
- began in 1985 by Leslie Lampart.
- is now maintained by the LaTeX3 Project.

What is LaTeX

LaTeX...

- is a document preparation system for high-quality typesetting.
- is *not* a word processor.
- began in 1985 by Leslie Lampart.
- is now maintained by the LaTeX3 Project.
- is FREE!

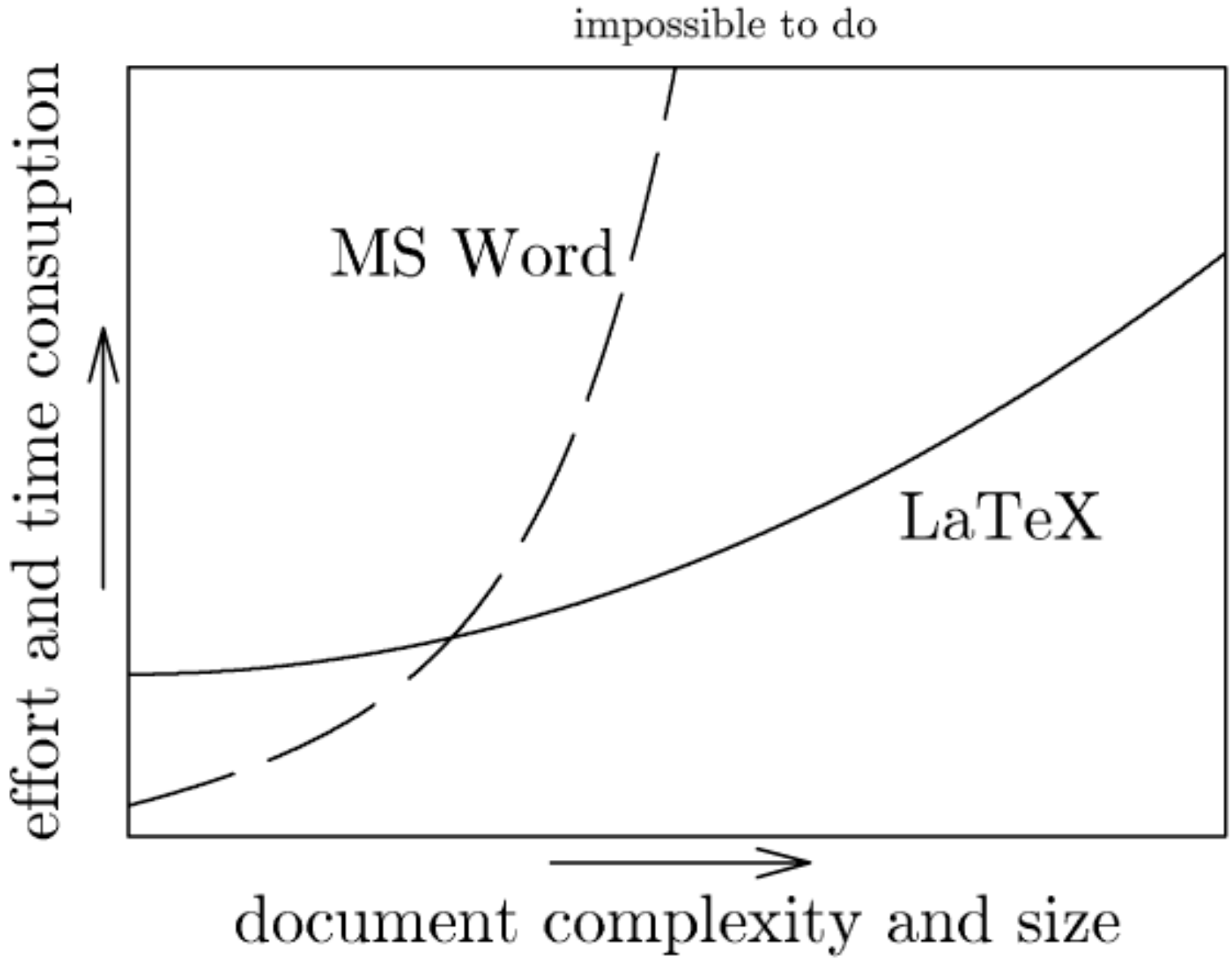
What is LaTeX

LaTeX...

- is a document preparation system for high-quality typesetting.
- is *not* a word processor.
- began in 1985 by Leslie Lampart.
- is now maintained by the LaTeX3 Project.
- is FREE!

  The fundamental idea around LaTeX is to focus on the content, *not* the formatting.

Why use LaTeX



Why use LaTeX

LaTeX Features

- Typesetting journal articles, technical reports, books, and slide presentations.
- Control over large documents containing sectioning, cross-references, tables and figures.
- Typesetting of complex mathematical formulas.
- Advanced typesetting of mathematics with AMS-LaTeX.
- Automatic generation of bibliographies and indexes.
- Multi-lingual typesetting.
- Inclusion of artwork, and process or spot colour.
- Using PostScript or Metafont fonts.

Installing R

The latest version of R can be obtained from `http://cran.r-project.org`.
The current version of R is:

```
> R.version$version.string
[1] "R version 3.1.1 (2014-07-10)"
```

Installing R

The latest version of R can be obtained from `http://cran.r-project.org`. The current version of R is:

```
> R.version$version.string
```

```
[1] "R version 3.1.1 (2014-07-10)"
```

For Windows the following should also be installed:

- RTools `http://www.murdoch-sutherland.com/Rtools/`

For Mac the following should also be installed which are available from `http://cran.r-project.org/bin/macosx/tools`

- gfortran-4.2.3
- tcl/tk 8.5.5

Installing LaTeX

For Windows:

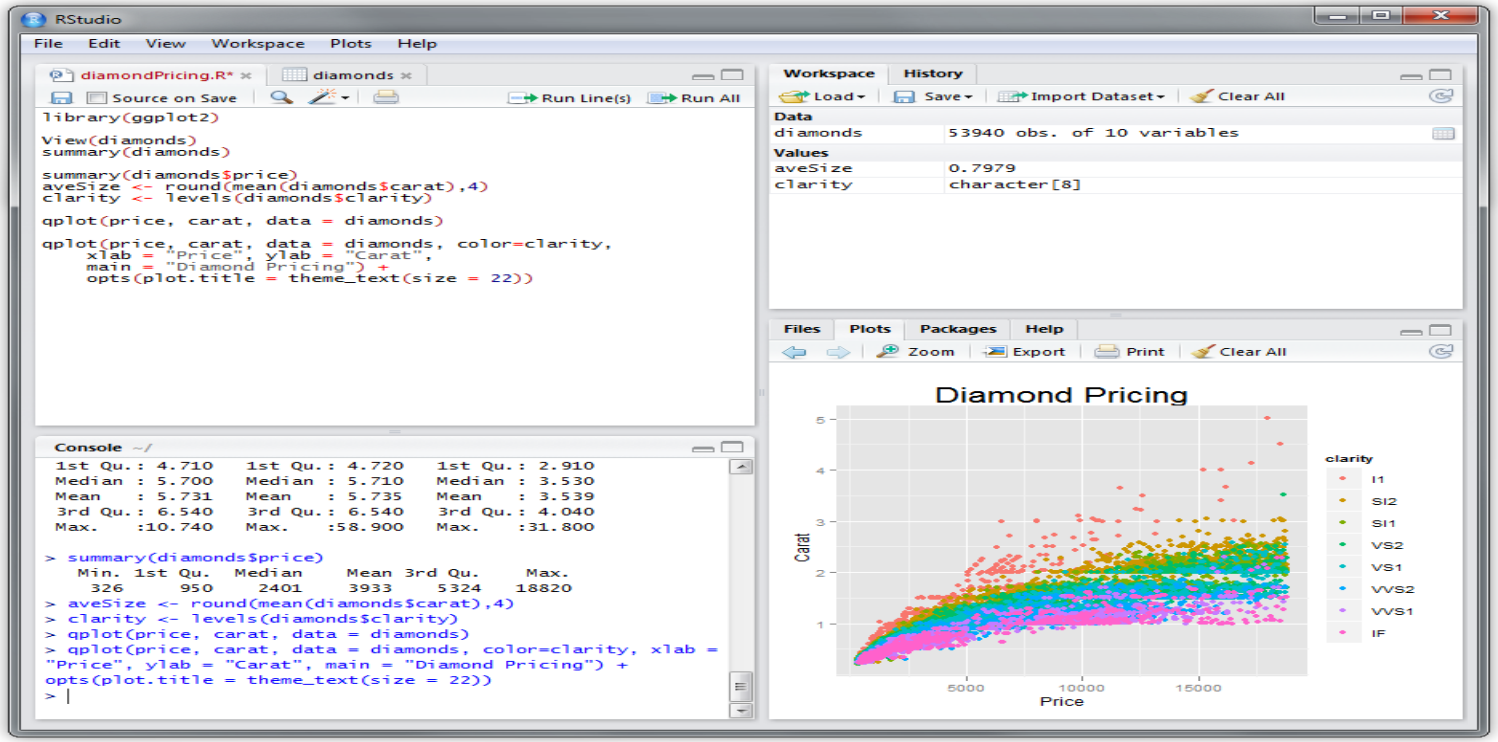
- MiKTeX `http://miktex.org/`

For Mac:

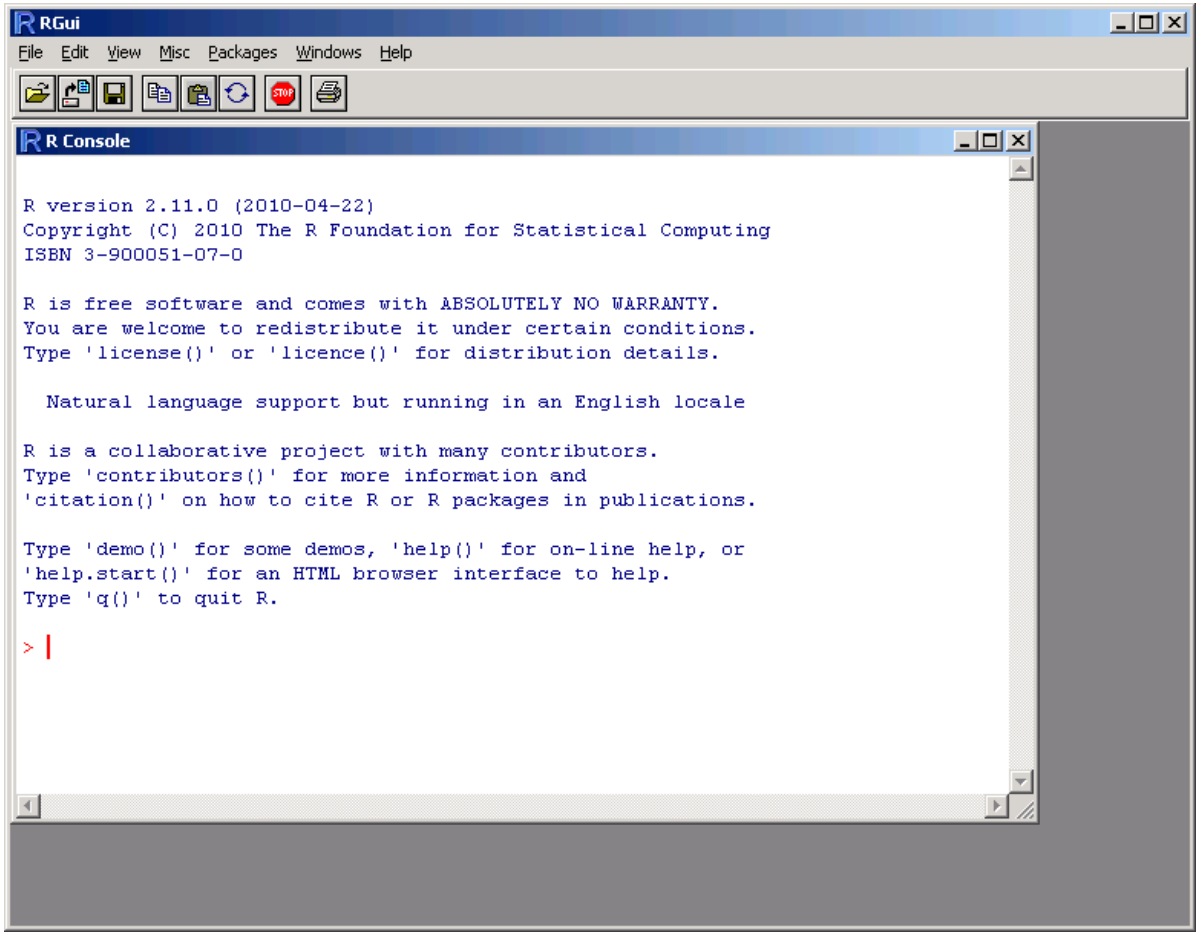
- MacTeX `http://www.tug.org/mactex/2011/`

Editors for R & LaTeX

There are many editors for R including the built in command line interface. However, we will make use of a relatively new Integrated Development Environment (IDE) designed specifically for R, namely RStudio (`http://rstudio.org`). It is available for Mac OS X, Windows, Linux, and as a Linux based server (which then runs in a web browser).



The R Environment



R as a Big Calculator

```
> 2 + 2
[1] 4
```

R as a Big Calculator

```
> 2 + 2
[1] 4
> 1 + sin(9)
[1] 1.4
```

R as a Big Calculator

```
> 2 + 2
[1] 4
> 1 + sin(9)
[1] 1.4
> 23.76 * log(8)/(23+atan(9))
[1] 2
```

Installing Packages

Both Windows and Mac have a menu system for installing packages, however the `install.packages` function allows for the installation to be scriptable.

```
> install.packages(c("psych", "gdata", "foreign", "devtools", "roxyg
```

Loading Packages

The `library` command will load a package into the current R session.

```
> library(psych)
> library(gdata)
> library(foreign)
```

Loading Packages

The `library` command will load a package into the current R session.

```
> library(psych)
> library(gdata)
> library(foreign)
```

For a list of packages that have been downloaded, but not necessarily attached, the `library()` function without any parameters will return that list.

```
> library()
```

Useful Packages

Package	Description
psych	Package contains lots of useful functions for descriptive statistics.
foreign	Contains functions to read SPSS files.
gdata	Contains functions to read Excel spreadsheets.
RODBC	Package contains functions to read and write data from ODBC databases (e.g. Oracle, MS SQLServer).
RMySQL	Package for interfacing with MySQL databases.
RSQLite	Package for the creation and editing of SQLite databases embedded within R.
MASS	Package to accompany Venables and Ripley's *Modern Applied Statistics with S*. See `http://www.stats.ox.ac.uk/pub/MASS4/`.
ggplot2	Fantastic package for creating really nice looking graphics `http://had.co.nz/ggplot2`.
rcmdr	R Commander is a graphical frontend for R.

Available Packages

The `search()` function will return all packages that are currently attached to the system.

```
> search()
```

```
 [1] ".GlobalEnv"        "package:foreign"   "package:gdata"
 [4] "package:psych"     "package:graphics"  "package:grDevices"
 [7] "package:utils"     "package:datasets"  "package:ggplot2"
[10] "package:stats"     "package:methods"   "Autoloads"
[13] "package:base"
```

Available Packages

The `search()` function will return all packages that are currently attached to the system.

```
> search()
```

```
 [1] ".GlobalEnv"        "package:foreign"   "package:gdata"
 [4] "package:psych"     "package:graphics"  "package:grDevices"
 [7] "package:utils"     "package:datasets"  "package:ggplot2"
[10] "package:stats"     "package:methods"   "Autoloads"
[13] "package:base"
```

You can then use the `ls()` function to return a list of functions in a particular package.

```
> ls('package:foreign')
```

```
 [1] "data.restore"  "lookup.xport"  "read.S"         "read.arff"
 [5] "read.dbf"      "read.dta"      "read.epiinfo"   "read.mtp"
 [9] "read.octave"   "read.spss"     "read.ssd"       "read.systat"
[13] "read.xport"    "write.arff"    "write.dbf"      "write.dta"
[17] "write.foreign"
```

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites
- The `help.search()` function will search the help file for a particular word or phrase. For example:

```
> help.search('cross tabs')
```

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites
- The `help.search()` function will search the help file for a particular word or phrase. For example:

  ```
  > help.search('cross tabs')
  ```

- To get documentation on a specific function, the `help()` function, or simply `?functionName` will open the documentation page in the web browser.

Getting Help

- R provides extensive documentation and help. The `help.start()` function will launch a webpage with links to:
  - The R manuals
  - The R FAQ
  - Search engine
  - and many other useful sites
- The `help.search()` function will search the help file for a particular word or phrase. For example:

  ```
  > help.search('cross tabs')
  ```

- To get documentation on a specific function, the `help()` function, or simply `?functionName` will open the documentation page in the web browser.
- Lastly, to search the R mailing lists, use the `RSiteSearch()` function.

Reading Excel Files

```
> students = read.xls("ECStudents.xls", sheet=1)
```

Reading Excel Files

```
> students = read.xls("ECStudents.xls", sheet=1)
> names(students)
[1] "Level"     "Division" "Degree"    "Enrolled"
[5] "Military" "Credits"  "ZipCode"   "State"
[9] "Country"
```

Reading Excel Files

```
> students = read.xls("ECStudents.xls", sheet=1)
> names(students)
[1] "Level"     "Division" "Degree"    "Enrolled"
[5] "Military" "Credits"  "ZipCode"   "State"
[9] "Country"
> nrow(students)
[1] 30494
```

Reading SPSS Files

The `foreign` package provides a function to read SPSS files.

```
> shy = read.spss("Exercise2.sav", use.value.labels=FALSE, to.data.f
> names(shy)
```

```
[1] "rowtype_"  "varname_"  "age"
[4] "technical" "social"    "frequency"
```

This data file contains six columns: social anxiety (soax), restricted emotionality (reemo), restricted affectionate behavior (reaff), intimate self-disclosure (isd), a single degree-of-freedom continuous measure of shyness (shy), and a three-group experimental structural variable (group).

Reading CSV Files

R can read virtually any type of plain text file with the `read.table` function. For convenience, the `read.csv` will provide a quick way of reading comma-separated values (CSV) files. For example:

```
> students = read.csv(file.choose(), header=TRUE)
```

Reading SQL Databases

```
> channel = odbcDriverConnect(connection="dburl:1521/live", readOnly
> students = sqlQuery(channel, "SELECT * FROM students")
> odbcClose(channel)
```

Data Frames

```
> head(students, n=5)
  Level Division Degree Enrolled Military Credits
1    GL       BU    MBA 09/24/09        N       9
2    GL       BU    MBG 07/25/05        N       3
3    GL       BU    MBG 08/30/05        N      NA
4    GL       BU    MBG 09/02/05        N       3
5    GL       BU    MBG 10/19/05        N      NA
  ZipCode State Country
1   27295    NC
2   77566    TX
3   11435    NY
4   07866    NJ
5   76065    TX
```

Data Frame Structure

```
> str(students)
'data.frame':        30494 obs. of  9 variables:
 $ Level    : chr  "GL" "GL" "GL" "GL" ...
 $ Division: chr  "BU" "BU" "BU" "BU" ...
 $ Degree   : chr  "MBA" "MBG" "MBG" "MBG" ...
 $ Enrolled: chr  "09/24/09" "07/25/05" "08/30/05" "09/02/05" ...
 $ Military: chr  "N" "N" "N" "N" ...
 $ Credits : int  9 3 NA 3 NA 6 NA NA NA 3 ...
 $ ZipCode : chr  "27295" "77566" "11435" "07866" ...
 $ State    : chr  "NC" "TX" "NY" "NJ" ...
 $ Country : chr  "" "" "" "" ...
```

NA vs. NULL

R is just as much a programming language as it is a statistical software package. As such it represents null differently for programming (using NULL) than for data (using NA).

NA vs. NULL

R is just as much a programming language as it is a statistical software package. As such it represents null differently for programming (using NULL) than for data (using NA).

> *NULL represents the null object in R: it is a reserved word. NULL is often returned by expressions and functions whose values are undefined.*

NA vs. NULL

R is just as much a programming language as it is a statistical software package. As such it represents null differently for programming (using NULL) than for data (using NA).

> *NULL represents the null object in R: it is a reserved word. NULL is often returned by expressions and functions whose values are undefined.*
>
> *NA is a logical constant of length 1 which contains a missing value indicator. NA can be freely coerced to any other vector type except raw. There are also constants NA_integer_, NA_real_, NA_complex_and NA_character_of the other atomic vector types which support missing values: all of these are reserved words in the R language.*

For more details, see `http://opendatagroup.com/2010/04/25/r-na-v-null/`

Frequency Tables

One-way frequency table

```
> table(students$Division)
```

```
   BU    HS    LA    NU    TE
 2433   231  8134 17088  2608
```

Frequency Tables

One-way frequency table

```
> table(students$Division)
```

```
   BU    HS    LA    NU    TE
 2433   231  8134 17088  2608
```

Two-way frequency table (the first parameter will be the rows, second the columns)

```
> mytable = table(students$Military, students$Division)
> mytable
```

```
     BU    HS    LA    NU    TE
N  1248   188  3140 16428   872
Y  1185    43  4994   660  1736
```

Tables of Proportions

Cell Percentages:

```
> prop.table(mytable)
```

```
      BU     HS     LA     NU     TE
N 0.0409 0.0062 0.1030 0.5387 0.0286
Y 0.0389 0.0014 0.1638 0.0216 0.0569
```

Tables of Proportions

Cell Percentages:

```
> prop.table(mytable)
```

```
      BU     HS     LA     NU     TE
N 0.0409 0.0062 0.1030 0.5387 0.0286
Y 0.0389 0.0014 0.1638 0.0216 0.0569
```

Row Percentages:

```
> prop.table(mytable, 1)
```

```
      BU     HS     LA     NU     TE
N 0.0570 0.0086 0.1435 0.7510 0.0399
Y 0.1375 0.0050 0.5795 0.0766 0.2014
```

Tables of Proportions

Cell Percentages:

```
> prop.table(mytable)
```

```
      BU     HS     LA     NU     TE
  N 0.0409 0.0062 0.1030 0.5387 0.0286
  Y 0.0389 0.0014 0.1638 0.0216 0.0569
```

Row Percentages:

```
> prop.table(mytable, 1)
```

```
      BU     HS     LA     NU     TE
  N 0.0570 0.0086 0.1435 0.7510 0.0399
  Y 0.1375 0.0050 0.5795 0.0766 0.2014
```

Column Percentages:

```
> prop.table(mytable, 2)
```

```
     BU    HS    LA    NU    TE
  N 0.513 0.814 0.386 0.961 0.334
  Y 0.487 0.186 0.614 0.039 0.666
```

Descriptive Statistics

Mean and standard deviation:

```
> mean(students$Credits, na.rm=TRUE)
[1] 4.5
> sd(students$Credits, na.rm=TRUE)
[1] 3.5
```

Descriptive Statistics

Mean and standard deviation:

```
> mean(students$Credits, na.rm=TRUE)
[1] 4.5
> sd(students$Credits, na.rm=TRUE)
[1] 3.5
```

However, the mean, median, 25th and 75th quartiles, min, and max can be returned in a single statement using the summary function:

```
> summary(students$Credits)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
      0       3       3       4       6      33
   NA's
  18533
```

Descriptive Statistics

The `psych` package contains the `describe` and `describe.by` functions which provide a convenient way of calculating summary statistics.

```
> describe(students$Credits)
```

```
  vars     n mean  sd median trimmed mad min max
1    1 11961  4.5 3.5      3     4.2 4.5   0  33
  range skew kurtosis   se
1    33    1      2.2 0.03
```

Descriptive Statistics (cont.)

The `describe.by` will calculate summary statistics by grouping variables. The `mat` parameter will return the results in matrix form.

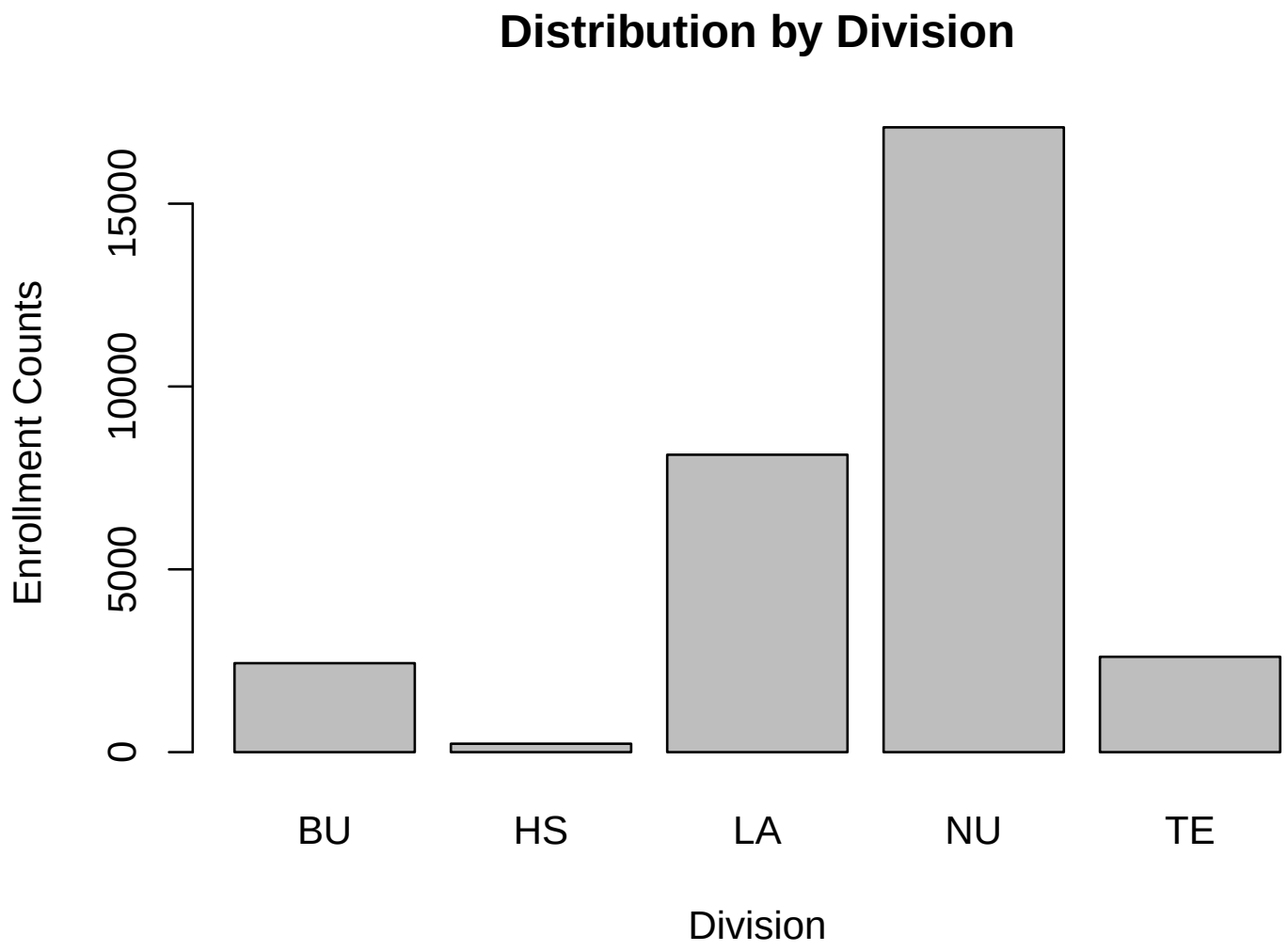
```
> describe.by(students$Credits, students$Division, na.rm=TRUE, mat=TRUE)
```

```
   item group1 vars    n mean  sd median trimmed
11    1     BU    1  932  4.8 3.7      3     4.4
12    2     HS    1  127  5.5 3.6      6     5.1
13    3     LA    1 2879  4.6 3.6      3     4.2
14    4     NU    1 7079  4.4 3.4      3     4.2
15    5     TE    1  944  4.5 3.4      3     4.1
   mad min max range skew kurtosis    se
11 4.4   0  24    24 1.52     3.73 0.120
12 4.4   0  17    17 0.90     0.54 0.319
13 4.4   0  33    33 1.47     4.24 0.067
14 4.4   0  24    24 0.64     0.47 0.040
15 4.4   0  24    24 1.28     3.31 0.109
```

Histograms

```
> barplot(table(students$Division),
    main='Distribution by Division',
    xlab='Division', ylab='Enrollment Counts')
```

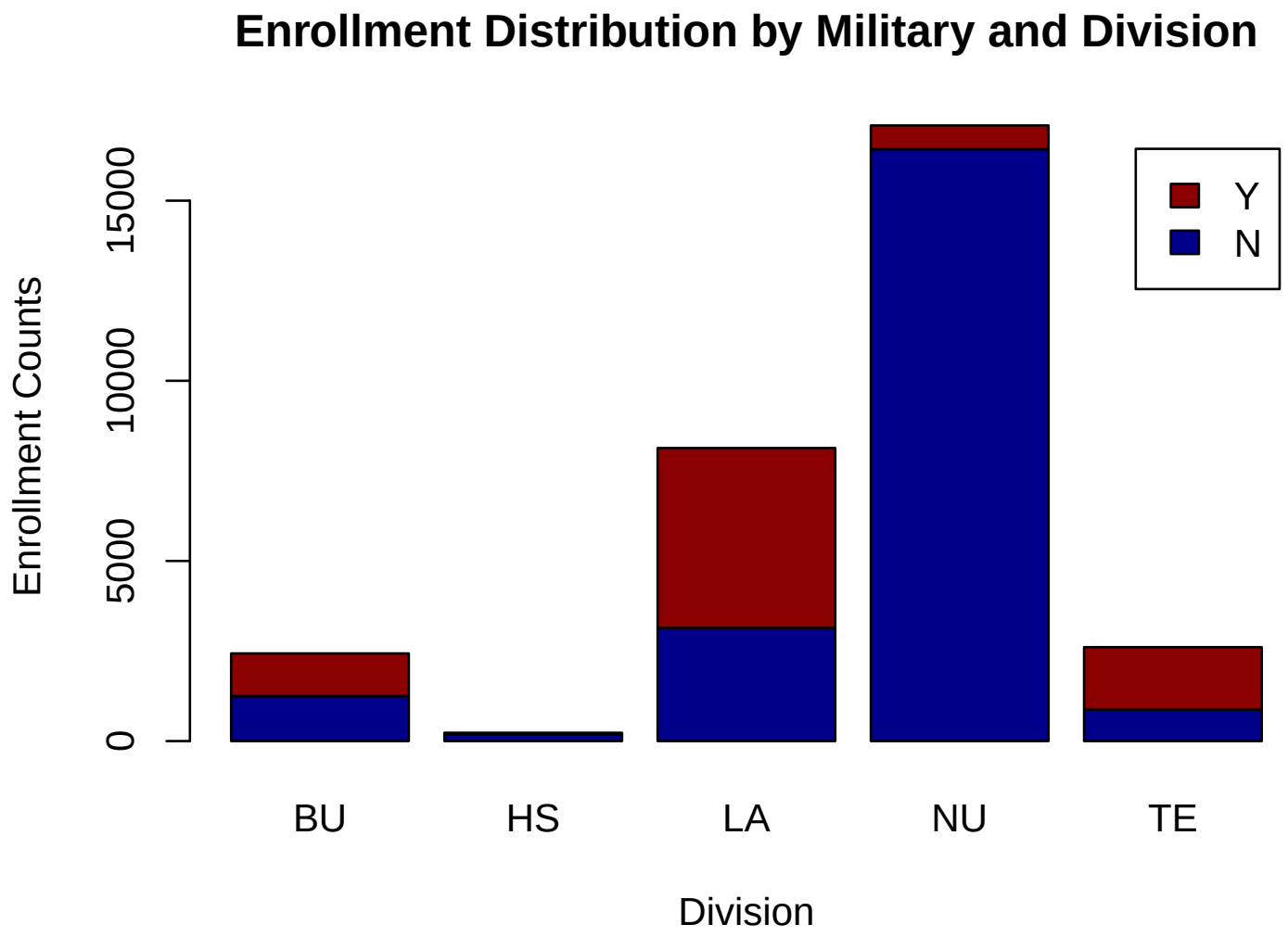
Histograms (cont.)



Histograms (cont.)

```
> counts = table(students$Military, students$Division)
> barplot(counts,
   main='Enrollment Distribution by Military and Division',
   xlab='Division', ylab='Enrollment Counts',
   legend=rownames(counts), col=c('darkblue', 'darkred'))
```

Histograms (cont.)



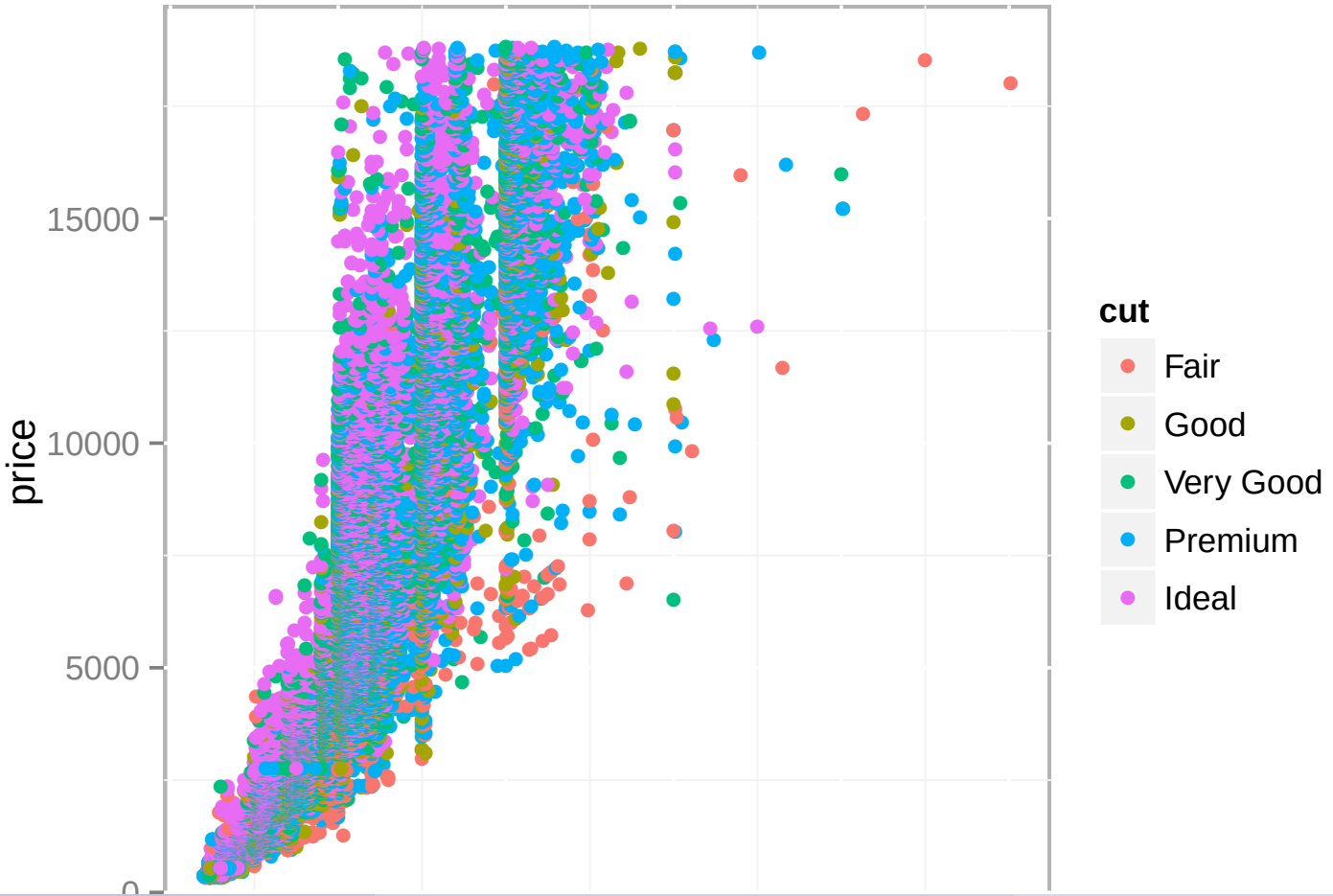
ggplot2: A Grammar of Graphics

- ggplot2 is an R package that provides an alternative framework based upon Wilkinson's (2005) Grammar of Graphics.
- ggplot2 is, in general, more flexible for creating "prettier" and complex plots.
- Works by creating layers of different types of objects/geometries (i.e. bars, points, lines, polygons, etc.)
- ggplot2 has at least three ways of creating plots:
  1. `qplot`
  2. `ggplot(...) + geom_XXX(...) + ...`
  3. `ggplot(...) + layer(...)`

  We will focus only on the second.

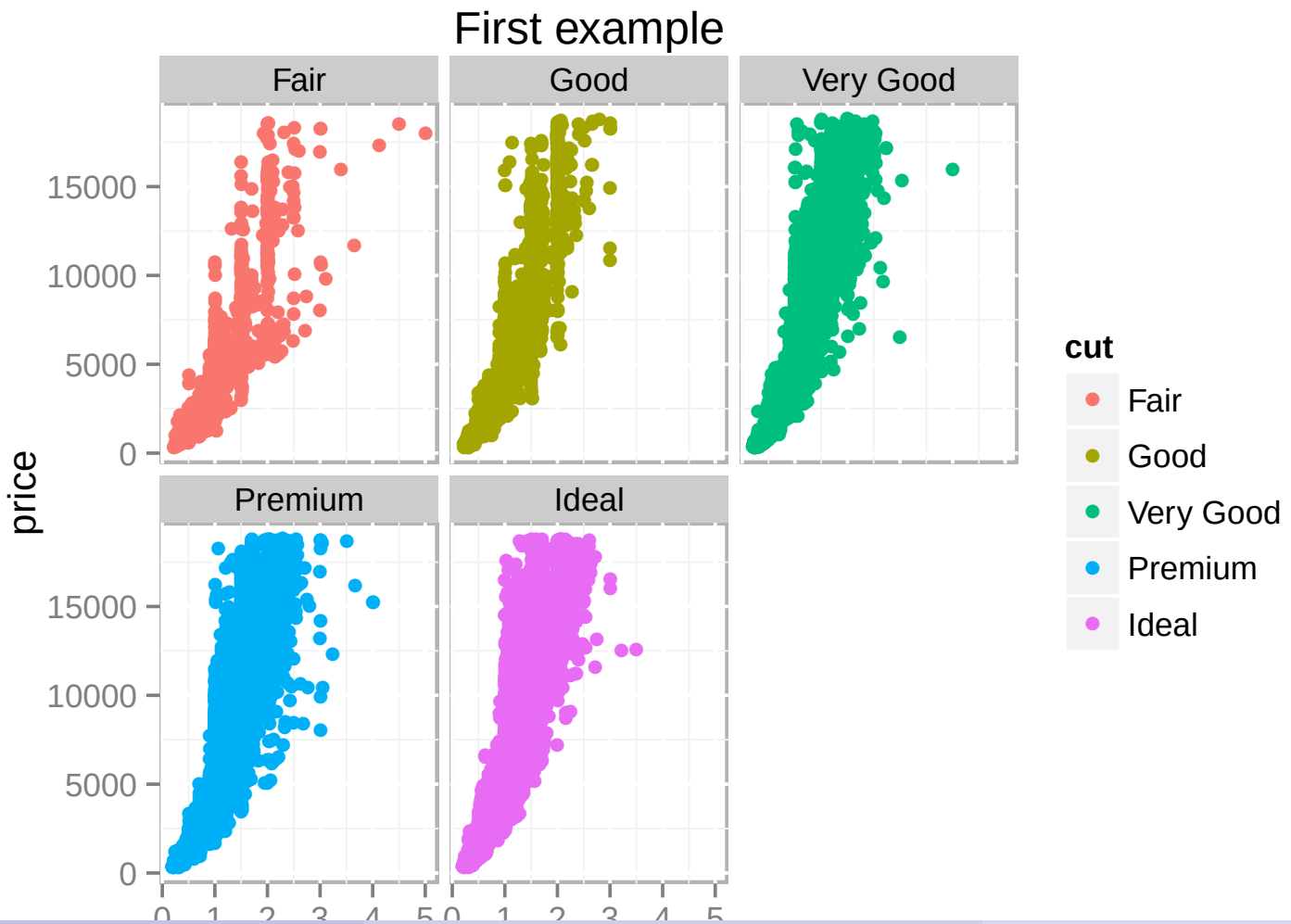
First Example

```
> data(diamonds)
> p <- ggplot(diamonds, aes(x=carat,y=price,colour=cut)) +
    geom_point()
> print(p)
```



First Example

```
> p <- p + facet_wrap(~cut) +
    ggtitle("First example")
> print(p)
```



Parts of a ggplot2 statement

- Data

```
ggplot(myDataFrame, aes(x=x, y=y)
```

Parts of a ggplot2 statement

- Data

  ```
  ggplot(myDataFrame, aes(x=x, y=y)
  ```

- Layers

  ```
  geom_point(), geom_histogram()
  ```

Parts of a ggplot2 statement

- Data

  `ggplot(myDataFrame, aes(x=x, y=y)`

- Layers

  `geom_point(), geom_histogram()`

- Facets

  `facet_wrap(~ cut), facet_grid(~ cut)`

Parts of a ggplot2 statement

- Data

  `ggplot(myDataFrame, aes(x=x, y=y)`

- Layers

  `geom_point(), geom_histogram()`

- Facets

  `facet_wrap(~ cut), facet_grid(~ cut)`

- Scales

  `scale_y_log10()`

Parts of a ggplot2 statement

- Data

  `ggplot(myDataFrame, aes(x=x, y=y)`

- Layers

  `geom_point(), geom_histogram()`

- Facets

  `facet_wrap(~ cut), facet_grid(~ cut)`

- Scales

  `scale_y_log10()`

- Other options

  `ggtitle('my title'), ylim(c(0, 10000)), xlab('x-axis label')`

Lots of geoms

geom_abline
geom_jitter
geom_area
geom_line
geom_bar
geom_linerange
geom_bin2d
geom_path
geom_blank
geom_point
geom_boxplot
geom_pointrange
geom_contour
geom_polygon
geom_crossbar
geom_quantile
geom_density
geom_rect
geom_density2d
geom_ribbon
geom_errorbar
geom_rug
geom_errorbarh
geom_segment
geom_freqpoly
geom_smooth
geom_hex
geom_step
geom_histogram
geom_text
geom_hline
geom_tile
geom_vline

Your First LaTeX File

```
\documentclass{article}
\title{Introduction to R and \LaTeX{} for IR}
\author{Jason Bryer}
\date{May 2010}
\begin{document}
\maketitle
Hello Albany useR Group!
\end{document}
```

Your First LaTeX File

```
\documentclass{article}
\title{Introduction to R and \LaTeX{} for IR}
\author{Jason Bryer}
\date{May 2010}
\begin{document}
\maketitle
Hello Albany useR Group!
\end{document}
```

- This document is an article.

Your First LaTeX File

```
\documentclass{article}
\title{Introduction to R and \LaTeX{} for IR}
\author{Jason Bryer}
\date{May 2010}
\begin{document}
\maketitle
Hello Albany useR Group!
\end{document}
```

- This document is an article.
- Its title is *Introduction to R for Reproducible Research*.

Your First LaTeX File

```
\documentclass{article}
\title{Introduction to R and \LaTeX{} for IR}
\author{Jason Bryer}
\date{May 2010}
\begin{document}
\maketitle
Hello Albany useR Group!
\end{document}
```

- This document is an article.
- Its title is *Introduction to R for Reproducible Research*.
- Its author is *Jason Bryer*.

Your First LaTeX File

```
\documentclass{article}
\title{Introduction to R and \LaTeX{} for IR}
\author{Jason Bryer}
\date{May 2010}
\begin{document}
\maketitle
Hello Albany useR Group!
\end{document}
```

- This document is an article.
- Its title is *Introduction to R for Reproducible Research*.
- Its author is *Jason Bryer*.
- It was written in *May 2010*.

Your First LaTeX File

```
\documentclass{article}
\title{Introduction to R and \LaTeX{} for IR}
\author{Jason Bryer}
\date{May 2010}
\begin{document}
\maketitle
Hello Albany useR Group!
\end{document}
```

- This document is an article.
- Its title is *Introduction to R for Reproducible Research*.
- Its author is *Jason Bryer*.
- It was written in *May 2010*.
- The document consists of a *title* followed by the text *Hello Albany useR Group!*.

LaTeX Help Sheet

X 2ε Cheat Sheet

lasses

is two-sided.
t divisions.
t or \chapter divisions.
?).
ns-serif font.
ginning of a document:
ss}. Use \begin{document} to start
document} to end the document.

nentclass options

Font size.
er Paper size.
Use two columns.
Set margins for two-sided.
Landscape orientation. Must use dvips -t landscape.
Double-space lines.
lass[*opt,opt*]{*class*}.

h margins.
ns with \marginsize{*l*}{*r*}{*t*}{*b*}.
umns with \begin{multicols}{*n*}.
X symbol font.
document}. Usage: \usepackage{*package*}

hor of document.
le of document.
e.
o before \begin{document}. The
tle goes at the top of the document.

Empty header, footer and no page numbers.

tructure

\subsubsection{*title*}
\paragraph{*title*}
\subparagraph{*title*}

can be followed with an *, like
o supress heading numbers.
mdepth}{*x*} supresses heading numbers of
hapter has depth 0.

ients

Comment block (not printed).
Indented quotation block.
Like quote with indented paragraphs.
Quotation block for verse.

Lists

\begin{enumerate}	Numbered list.
\begin{itemize}	Bulletted list.
\begin{description}	Description list.
\item *text*	Add an item.
\item[*x*] *text*	Use *x* instead of normal bullet or number. Required for descriptions.

References

\label{*marker*}	Set a marker for cross-reference, often of the form \label{sec:item}.
\ref{*marker*}	Give section/body number of marker.
\pageref{*marker*}	Give page number of marker.
\footnote{*text*}	Print footnote at bottom of page.

Floating bodies

\begin{table}[*place*]	Add numbered table.
\begin{figure}[*place*]	Add numbered figure.
\begin{equation}[*place*]	Add numbered equation.
\caption{*text*}	Caption for the body.

The *place* is a list valid placements for the body. t=top, h=here, b=bottom, p=separate page, !=place even if ugly. Captions and label markers should be within the environment.

Text properties

Font face

Command	*Declaration*	*Effect*
\textrm{*text*}	{\rm *text*}	Roman family
\textsf{*text*}	{\sf *text*}	Sans serif family
\texttt{*text*}	{\tt *text*}	Typewriter family
\textmd{*text*}	{\md *text*}	Medium series
\textbf{*text*}	{\bf *text*}	**Bold series**
\textup{*text*}	{\up *text*}	Upright shape
\textit{*text*}	{\it *text*}	*Italic shape*
\textsl{*text*}	{\sl *text*}	*Slanted shape*
\textsc{*text*}	{\sc *text*}	Small Caps shape
\emph{*text*}	{\em *text*}	*Emphasized*
\textnormal{*text*}	{\normalfont *text*}	Document font
\underline{*text*}		Underline

The command (*tttt*) form handles spacing better than the declaration (*ttt*) form.

Font size

\tiny	tiny	\Large	Large
\scriptsize	scriptsize	\LARGE	LARGE
\footnotesize	footnotesize	\huge	huge
\small	small		
\normalsize	normalsize	\Huge	Huge
\large	large		

These are declarations and should be used in the form {\small ...} or without braces to affect the entire document.

Verbatim text

\begin{verbatim}	Verbatim environment.
\begin{verbatim*}	Spaces are shown as ␣.
\verb!text!	Text between the delimiting characters (in this case !) is verbatim.

Justification

Environment	*Declaration*
\begin{center}	\centering
\begin{flushleft}	\raggedright
\begin{flushright}	\raggedleft

Miscellaneous

\linespread{*x*} changes the line spacing by the multiplier *x*.

Text-mode symbols

Symbols

&	\&	_	_	...	\ldots	•	\textbullet
$	\$	^	\^{}	|	\textbar	\	\textbackslash
%	\%	~	\~{}	#	\#	|	\textbar

Accents

ò	\`o	ó	\'o	ô	\^o	õ	\~o	ō	\=o
ȯ	\.o	ö	\"o	o͡	\c o	ŏ	\v o	ő	\H o
ç	\c c	o̩	\d o	o̲	\b o	o͡o	\t oo	œ	\oe
Œ	\OE	æ	\ae	Æ	\AE	å	\aa	Å	\AA
ø	\o	Ø	\O	ł	\l	Ł	\L	ı	\i
ȷ	\j	¡	~`	¿	?`				

Delimiters

` `	" ``	{ \{	[[	((	< \textless
' '	" ''	} \}	]]	))	> \textgreater

Dashes

Name	*Source*	*Example*	*Usage*
hyphen	-	X-ray	In words.
en-dash	--	1–5	Between numbers.
em-dash	---	Yes—or no?	Punctuation.

Line and page breaks

\\	Begin new line without new paragraph.
*	Prohibit pagebreak after linebreak.
\kill	Don't print current line.
\pagebreak	Start new page.
\noindent	Do not indent current line.

Miscellaneous

\today	May 11, 2002.
$\sim$	Prints ~ instead of \~{}, which makes ˜.
~	Space, disallow linebreak (W.J.~Clinton).
\@.	Indicate that the . ends a sentence when following an uppercase letter.
\hspace{*l*}	Horizontal space of length *l* (Ex: *l* = 20pt).
\vspace{*l*}	Vertical space of length *l*.
\rule{*w*}{*h*}	Line of width *w* and height *h*.

Jason Bryer, Ph.D. (www.bryer.org)

LaTeX Help Sheet

ironments

nment

\> Go to tab stop.

t on "invisible" lines with \kill at the end
ly \\ is used to separate lines.

nment

]{*cols*}
os]{*cols*}
width}[*pos*]{*cols*}

specification

ified column.
column.
tified column.
\parbox[t]{*width*}.
cl instead of inter-column space.
vertical line between columns.

s

ontal line between rows.
ontal line across columns x through y.
cols}{*text*}
l that spans n columns, with *cols* column
ication.

surround text with $ or use

erscript^x	_{x}	Subscript$_x$
	\sum_{k=1}^n	$\sum_{k=1}^{n}$

mbols

≥	\geq	≠	\neq
×	\times	÷	\div
∘	\circ	⋯	\cdots
β	\beta	γ	\gamma
ε	\epsilon	ε	\varepsilon
η	\eta	θ	\theta
ι	\iota	κ	\kappa
μ	\mu	ν	\nu
π	\pi	ρ	\rho
τ	\tau	υ	\upsilon
χ	\chi	ψ	\psi
Γ	\Gamma	Δ	\Delta
Λ	\Lambda	Ξ	\Xi
Σ	\Sigma	ϒ	\Upsilon
Ψ	\Psi	Ω	\Omega

ls

22°C: $22^{\circ}\mathrm{C}$.

y and citations

X, you need to run latex, bibtex, and
o resolve dependencies.

Citation types

\cite{*key*}	Full author list and year. (Watson and Crick 1953)
\citeA{*key*}	Full author list. (Watson and Crick)
\citeN{*key*}	Full author list and year. Watson and Crick (1953)
\shortcite{*key*}	Abbreviated author list and year. ?
\shortciteA{*key*}	Abbreviated author list. ?
\shortciteN{*key*}	Abbreviated author list and year. ?
\citeyear{*key*}	Cite year only. (1953)

All the above have an NP variant without parentheses; Ex. \citeNP.

BibTeX entry types

@article	Journal or magazine article.
@book	Book with publisher.
@booklet	Book without publisher.
@conference	Article in conference proceedings.
@inbook	A part of a book and/or range of pages.
@incollection	A part of book with its own title.
@manual	Technical documentation.
@mastersthesis	Master's thesis.
@misc	If nothing else fits.
@phdthesis	PhD. thesis.
@proceedings	Proceedings of a conference.
@techreport	Tech report, usually numbered in series.
@unpublished	Unpublished.

BibTeX fields

address	Address of publisher. Not necessary for major publishers.
author	Names of authors, of format ....
booktitle	Title of book when part of it is cited.
chapter	Chapter or section number.
edition	Edition of a book.
editor	Names of editors.
institution	Sponsoring institution of tech. report.
journal	Journal name.
key	Used for cross ref. when no author.
month	Month published. Use 3-letter abbreviation.
note	Any additional information.
number	Number of journal or magazine.
organization	Organization that sponsors a conference.
pages	Page range (2,6,9--12).
publisher	Publisher's name.
school	Name of school (for thesis).
series	Name of series of books.
title	Title of work.
type	Type of tech. report, ex. "Research Note".
volume	Volume of a journal or book.
year	Year of publication.

Not all fields need to be filled. See example below.

Common BibTeX style files

abbrv	Standard	abstract	alpha with abstract
alpha	Standard	apa	APA
plain	Standard	unsrt	Unsorted

The LaTeX document should have the following two lines just before \end{document}, where bibfile.bib is the name of the BibTeX file.

```
\bibliographystyle{plain}
\bibliography{bibfile}
```

BibTeX example

The BibTeX database goes in a file called *file*.bib, which is processed with bibtex file.

```
@String{N = {Na\-ture}}
@Article{WC:1953,
  author  = {James Watson and Francis Crick},
  title   = {A structure for Deoxyribose Nucleic Acid},
  journal = N,
  volume  = {171},
  pages   = {737},
  year    = 1953
}
```

Sample LaTeX document

```
\documentclass[11pt]{article}
\usepackage{fullpage}
\title{Template}
\author{Name}
\begin{document}
\maketitle

\section{section}
\subsection*{subsection without number}
text \textbf{bold text} text. Some math: $2+2=5$
\subsection{subsection}
text \emph{emphasized text} text. \cite{WC:1953}
discovered the structure of DNA.

A table:
\begin{table}[!th]
\begin{tabular}{|l|c|r|}
\hline
first  &  row  &  data \\
second &  row  &  data \\
\hline
\end{tabular}
\caption{This is the caption}
\label{ex:table}
\end{table}

The table is numbered \ref{ex:table}.
\end{document}
```

What is Sweave?

Sweave...

- is a function built into R.

What is Sweave?

Sweave...

- is a function built into R.
- bridges R with LaTeX by allowing for R code to be embedded directly into a LaTeX file.

What is Sweave?

Sweave...

- is a function built into R.
- bridges R with LaTeX by allowing for R code to be embedded directly into a LaTeX file.
- is designed "to create dynamic reports, which can be updated automatically if data or analysis change" (Leisch, 2002).

What is Sweave?

Sweave...

- is a function built into R.
- bridges R with LaTeX by allowing for R code to be embedded directly into a LaTeX file.
- is designed "to create dynamic reports, which can be updated automatically if data or analysis change" (Leisch, 2002).
- is documented here: `help("Sweave", package="utils")`

What is Sweave?

Sweave...

- is a function built into R.
- bridges R with LaTeX by allowing for R code to be embedded directly into a LaTeX file.
- is designed "to create dynamic reports, which can be updated automatically if data or analysis change" (Leisch, 2002).
- is documented here: `help("Sweave", package="utils")`
- allows R code to be embedded directly into LaTeX files using the following format:

```
<<label,options>>=
2 + 2
@
```

What is Sweave?

Sweave...

- is a function built into R.
- bridges R with LaTeX by allowing for R code to be embedded directly into a LaTeX file.
- is designed "to create dynamic reports, which can be updated automatically if data or analysis change" (Leisch, 2002).
- is documented here: `help("Sweave", package="utils")`
- allows R code to be embedded directly into LaTeX files using the following format:

```
<<label,options>>=
2 + 2
@
```

- Page 13 of the *Sweave User Manual* contains the complete list of options.

Sweave Example

```
\documentclass[a4paper]{article}
<<results=hide,echo=FALSE>>=
library(gdata)
library(xtable)
@
\title{Sweave Example}
\author{Jason Bryer}
\begin{document}
\maketitle
In this example will will look at the frequency of
military status by division.
<<results=hide>>=
students = read.xls("ECStudents.xls", sheet=1)
@
```

Sweave Example

```
\documentclass[a4paper]{article}
<<results=hide,echo=FALSE>>=
library(gdata)
library(xtable)
@
\title{Sweave Example}
\author{Jason Bryer}
\begin{document}
\maketitle
In this example will will look at the frequency of
military status by division.
<<results=hide>>=
students = read.xls("ECStudents.xls", sheet=1)
@
```

Sweave Example

```
\documentclass[a4paper]{article}
<<results=hide,echo=FALSE>>=
library(gdata)
library(xtable)
@
\title{Sweave Example}
\author{Jason Bryer}
\begin{document}
\maketitle
In this example will will look at the frequency of
military status by division.
<<results=hide>>=
students = read.xls("ECStudents.xls", sheet=1)
@
```

Sweave Example

```
\documentclass[a4paper]{article}
<<results=hide,echo=FALSE>>=
library(gdata)
library(xtable)
@
\title{Sweave Example}
\author{Jason Bryer}
\begin{document}
\maketitle
In this example will will look at the frequency of
military status by division.
<<results=hide>>=
students = read.xls("ECStudents.xls", sheet=1)
@
```

Sweave Example

```
<<echo=FALSE,results=tex>>=
xtable(table(students$Military, students$Division))
@
\begin{center}
<<fig=TRUE,echo=FALSE>>=
barplot(table(students$Military, students$Division),
    main='Distribution by Military & Division',
    xlab='Division', ylab='Enrollment Counts')
@
\end{center}
\end{document}
```

Sweave Example

```
<<echo=FALSE,results=tex>>=
xtable(table(students$Military, students$Division))
@
\begin{center}
<<fig=TRUE,echo=FALSE>>=
barplot(table(students$Military, students$Division),
    main='Distribution by Military & Division',
    xlab='Division', ylab='Enrollment Counts')
@
\end{center}
\end{document}
```

Sweave Example

```
<<echo=FALSE,results=tex>>=
xtable(table(students$Military, students$Division))
@
\begin{center}
<<fig=TRUE,echo=FALSE>>=
barplot(table(students$Military, students$Division),
    main='Distribution by Military & Division',
    xlab='Division', ylab='Enrollment Counts')
@
\end{center}
\end{document}
```

Sweave Example

Sweave Example

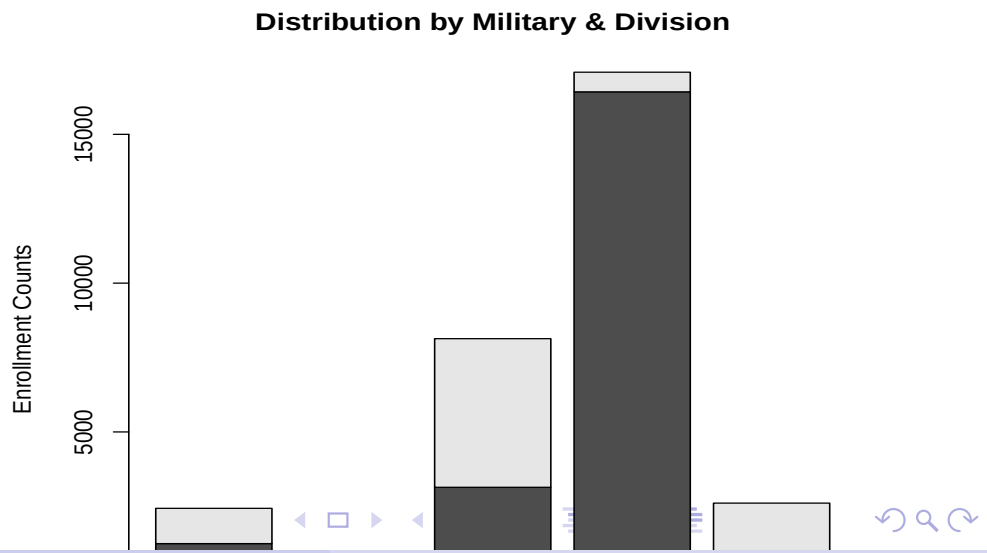
Jason Bryer

May 29, 2010

In this example will will look at the frequency of military status by division.

```
> students = read.xls("ECStudents.xls", sheet = 1)
```

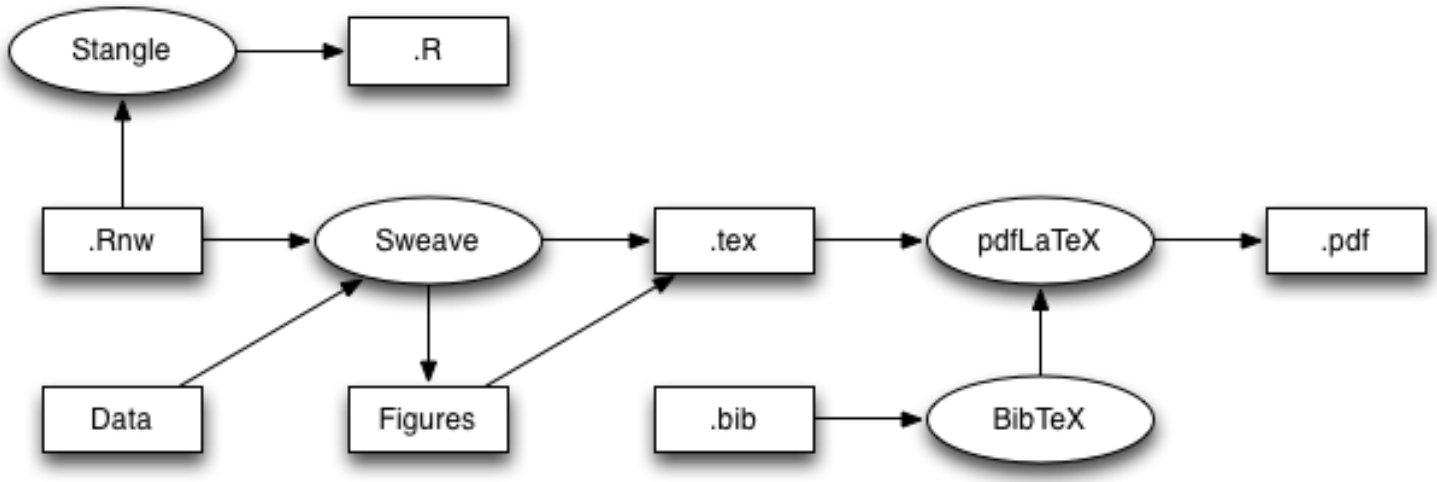
	BU	HS	LA	NU	TE
N	1248	188	3140	16428	872
Y	1185	43	4994	660	1736



Sweave Workflow

There are several steps required to go from a source file containing LaTeX and R to a final document. Specifically...

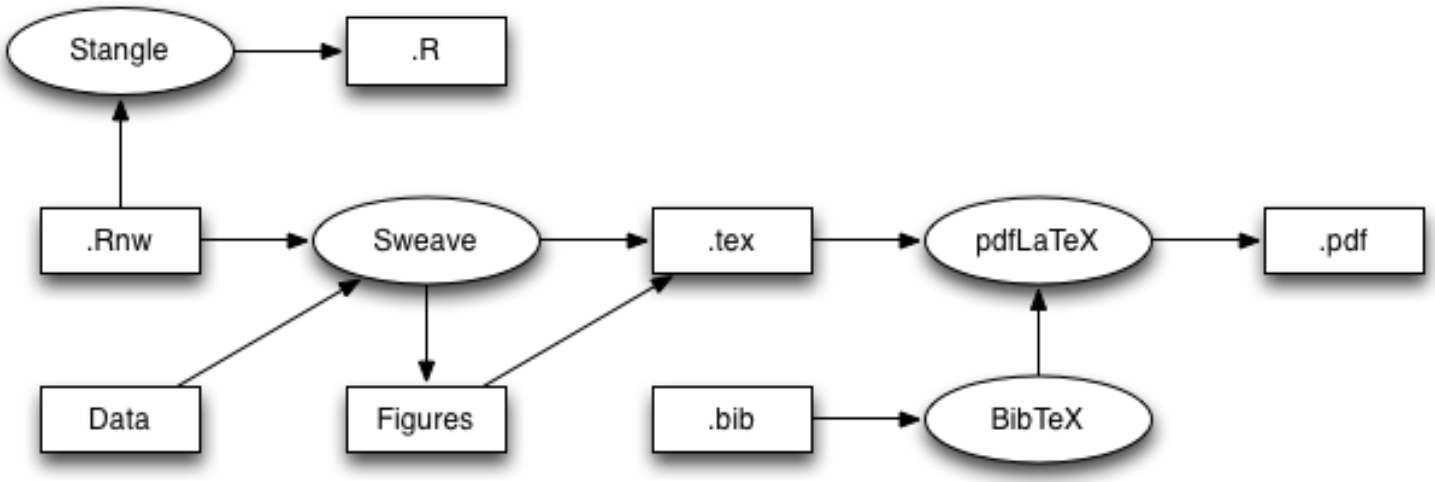
- Create a source file. This is a plain text file, usually with a .Rnw file extension.
- Run the `Sweave` function in R. This will create a .tex file.
- Run `latex` on the resulting .tex file. This will vary slightly based upon the platform but will result in a postscript and/or PDF file.



Sweave Workflow

There are several steps required to go from a source file containing LaTeX and R to a final document. Specifically...

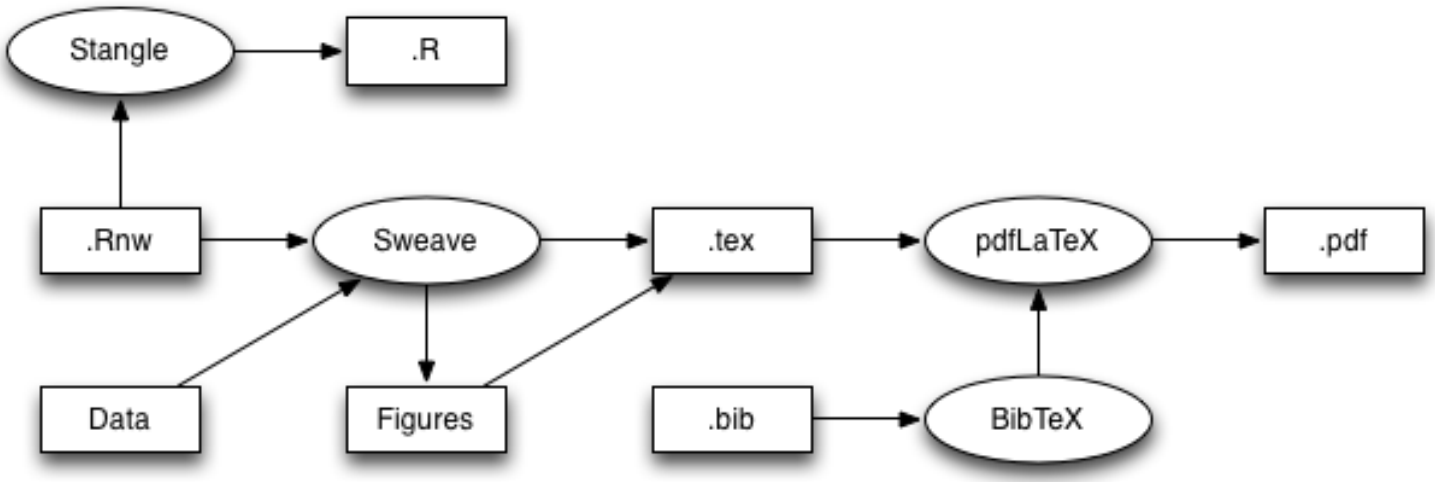
- Create a source file. This is a plain text file, usually with a .Rnw file extension.
- Run the `Sweave` function in R. This will create a .tex file.
- Run `latex` on the resulting .tex file. This will vary slightly based upon the platform but will result in a postscript and/or PDF file.



Sweave Workflow

There are several steps required to go from a source file containing LaTeX and R to a final document. Specifically...

- Create a source file. This is a plain text file, usually with a .Rnw file extension.
- Run the `Sweave` function in R. This will create a .tex file.
- Run `latex` on the resulting .tex file. This will vary slightly based upon the platform but will result in a postscript and/or PDF file.



Conclusions

My goals today were to...

- Introduce a new way of thinking about report generation.
- To *not* provide a comprehensive overview of R and LaTeX, but instead to...
- Introduce the vast communities of R and LaTeX.
- Provide a cheaper, and perhaps more efficient, way of generating IR reports.
- Could you see yourself, or your institution, utilizing these frameworks?

Conclusions

My goals today were to...

- Introduce a new way of thinking about report generation.
- To *not* provide a comprehensive overview of R and LaTeX, but instead to...
- Introduce the vast communities of R and LaTeX.
- Provide a cheaper, and perhaps more efficient, way of generating IR reports.
- Could you see yourself, or your institution, utilizing these frameworks?

Conclusions

My goals today were to...

- Introduce a new way of thinking about report generation.
- To *not* provide a comprehensive overview of R and LaTeX, but instead to...
- Introduce the vast communities of R and LaTeX.
- Provide a cheaper, and perhaps more efficient, way of generating IR reports.
- Could you see yourself, or your institution, utilizing these frameworks?

Conclusions

My goals today were to...

- Introduce a new way of thinking about report generation.
- To *not* provide a comprehensive overview of R and LaTeX, but instead to...
- Introduce the vast communities of R and LaTeX.
- Provide a cheaper, and perhaps more efficient, way of generating IR reports.
- Could you see yourself, or your institution, utilizing these frameworks?

Conclusions

My goals today were to...

- Introduce a new way of thinking about report generation.
- To *not* provide a comprehensive overview of R and LaTeX, but instead to...
- Introduce the vast communities of R and LaTeX.
- Provide a cheaper, and perhaps more efficient, way of generating IR reports.
- Could you see yourself, or your institution, utilizing these frameworks?

Further Reading

Name	URL
R-Bloggers	http://r-bloggers.com
R in Action	http://www.manning.com/kabacoff/
R for SAS & SPSS Users	http://oit.utk.edu/scc/RforSAS&SPSSusers.pdf
An Introduction to R	http://cran.r-project.org/doc/manuals/R-intro.pdf
simpleR: Using R for Introductory Statistics	http://cran.r-project.org/doc/contrib/Verzani-SimpleR.pdf
Quick-R	http://statmethods.net
Task Views	http://cran.r-project.org/web/views
R Seek: An R Search Engine	http://www.rseek.org
R Reference Card	http://cran.r-project.org/doc/contrib/Short-refcard.pdf
The Personality Project	http://www.personality-project.org/r
R Cheat Sheets	http://devcheatsheet.com/tag/r
ggplot2	http://had.co.nz/ggplot2
More Math Into LaTeX (First section is free)	http://www.ctan.org/tex-archive/info/mil/mil.pdf
Wikibooks	http://en.wikibooks.org/wiki/LaTeX
LaTeX Help Sheet	http://www.scribd.com/doc/191838/LaTeX-Help-Sheet
Sweave User Manual	
Beamer LaTeX style used to create this presentation.	http://latex-beamer.sourceforge.net

Thank You

Jason Bryer (jason@bryer.org)

http://github.com/jbryer/IntroR

http://www.bryer.org