

Appendix A

A Glossary of R Jargon

Below is a selection of common R terms defined using SAS/SPSS jargon (or plain English when possible) and R jargon. Some definitions in SAS/SPSS jargon are quite loose given the fact that they have no direct analog of some R terms. Items in *italics* are included in the glossary. Definitions in R terms are often quoted (with permission) or paraphrased from S Poetry, by Patrick Burns [40].

Table A.1 Glossary of R Jargon

| | Defined in SAS/SPSS Terms | Defined in R Terms |
|---------------------|---|--|
| Apply | The process of having a procedure work on variables or observations/cases. Determines whether a procedure will act as a typical procedure or as a function instead. Also a function that does that. | The process of targeting a function on rows or columns. Also a <i>function</i> that does that. |
| Argument | Parameter, option or setting that controls what a procedure does. Includes variables to analyze. | Input to a <i>function</i> . |
| Array | Multiple datasets that are linked in layers. All variables must be only one type, e.g., all numeric or all character. | A <i>vector</i> with a <i>dim</i> attribute. The <i>dim</i> controls the number and size of dimensions. |
| Assignment function | The two-key sequence, “<-”, that places data or results of procedures or transformations into a variable or dataset. | The two-key sequence, “<-”, that gives <i>names</i> to <i>objects</i> . |
| Atomic object | A variable whose values are all of one type such as all numeric or all character. | An <i>object</i> whose <i>components</i> are all of one <i>mode</i> . Modes allowed are numeric, character, logical, or complex. |
| Attach | The process of adding a dataset or add-on module to your path. Attaching a dataset appears to copy the variables into an area | The process of adding a <i>database</i> to your <i>search list</i> . Also a <i>function</i> that does this. |

Table A.1 (continued)

| | Defined in SAS/SPSS Terms | Defined in R Terms |
|--------------------|---|--|
| | that lets you use them by a simple name like “gender” rather than by compound name like “mydata\$gender”. Done using the <code>attach</code> function. | |
| Attributes | Traits of a dataset like its variable names and labels. | Traits of objects such as <i>names</i> , <i>class</i> , or <i>dim</i> . |
| Class | An attribute of a variable or dataset that a procedure uses to change its default settings automatically. For variables, this is similar to setting the scale of a variable to help you decide what procedures it will work with. | The class attribute of an object determines which <i>method</i> of a <i>generic function</i> is used when the <i>object</i> is an <i>argument</i> in the <i>function</i> call. |
| Component | Like an entry in a SAS catalog. Can also be a variable in a dataset. | An item in a <i>list</i> . The <i>length</i> of a list is the number of components it has. |
| CRAN | The Comprehensive R Archive Network at http://cran.r-project.org/ . Consists of a set of sites around the world called mirrors that provide R and its add-on packages for you to download and install. | |
| Data frame | A dataset. | A set of <i>vectors</i> bound together in a <i>list</i> . They can be different <i>modes</i> or <i>classes</i> , e.g., numeric and character, but they must have equal <i>length</i> . |
| Database | One dataset or a set of them in a library, or an add-on module. | An item on the <i>search list</i> , or something that might be. Can be an R data file or a <i>package</i> . |
| Dim | A variable whose values are the number of rows and columns in a dataset. It is stored in the dataset itself. Also a procedure that prints or sets these values. | The <i>attribute</i> that describes the dimensions of an array. Also the <i>function</i> that retrieves or changes that attribute. |
| Element | A value. | An item in an <i>atomic vector</i> . |
| Extractor function | A procedure that gets more results from a dataset created by another procedure. | A <i>function</i> that has <i>methods</i> that apply to <i>modeling objects</i> . |
| Factor | A categorical variable and its value labels. Value labels may be nothing more than “1”, “2” if not assigned explicitly. | The type of <i>object</i> that represents a categorical variable. It stores its <i>labels</i> in its <i>levels attribute</i> . |
| Function | A procedure and/or a function. When you apply it down through cases, it is just like a procedure. But you can also apply it across rows like a function. | A program that is stored as an <i>object</i> . |
| Generic function | A procedure or function that has different default parameters set | A function whose behavior is determined by the <i>class</i> of one or |

Table A.1 (continued)

| | Defined in SAS/SPSS Terms | Defined in R Terms |
|---------|--|--|
| | depending upon the type of data you give it. | more of its <i>arguments</i> . The class of the relevant argument(s) determines which <i>method</i> the generic function will use. |
| Index | The order number of a variable in a dataset, or of a value in a variable. In our practice dataset, gender is the second variable so its index is 2. Gender is mydata[,2]. The first index selects rows, the second columns. If empty, it refers to all rows/columns. | The number of a <i>component</i> in a <i>list</i> or <i>data frame</i> , or of an <i>element</i> in a <i>vector</i> . |
| Install | You install packages just like add-ons, just once per version. | Adding a <i>package</i> into your <i>library</i> . |
| Label | A procedure that creates variable labels. Also a parameter that sets value labels using the <code>factor</code> or <code>ordered</code> procedures. | A <i>function</i> from the <code>Hmisc</code> <i>package</i> that creates variable labels. Also an <i>argument</i> that sets factor labels using the <code>factor</code> or <code>ordered</code> functions. |
| Length | The number of observations/cases in a variable (including missing values), or the number of variables in a dataset. | A measure of <i>objects</i> . For <i>vectors</i> , it is the number of its <i>elements</i> (including NAs). For <i>lists</i> or <i>data frames</i> , it is the number of its <i>components</i> . |
| Levels | The values that a categorical variable can have. Actually stored as a part of the variable itself in what appears to be a very short character variable (even when the values themselves are numbers). | An attribute to a factor object that is a character <i>vector</i> of the values the <i>factor</i> can have. Also an <i>argument</i> to the <code>factor</code> and <code>ordered</code> functions that can set the levels. |
| Library | Where a given version of R stores its base packages and the add-on modules you have installed. Also a procedure that loads a package from the library into working memory. You must do that in every R session before using a package. | A directory containing R <i>packages</i> that is set up so that the <i>library</i> function can <i>attach</i> it. Also a <i>function</i> that <i>attaches</i> a package from the <i>library</i> onto your <i>search list</i> . You must do that in every R session before using a function in the package. |
| List | Like a zipped collection of datasets that you can analyze easily without unzipping. | A set of <i>objects</i> of any <i>class</i> . Its <i>components</i> can be <i>vectors</i> , <i>data frames</i> , <i>matrices</i> and even other <i>lists</i> . |
| Load | Bringing a dataset (or collection of datasets) from disk into memory. You must do this before you can use data in R. Also the procedure that performs that task. | Bringing a R data file into your <i>workspace</i> . Also a <i>function</i> that performs that task. |

Table A.1 (continued)

| | Defined in SAS/SPSS Terms | Defined in R Terms |
|-----------------------------|---|--|
| Matrix | A dataset that must contain only one type of variable, e.g., all numeric or character. Helpful in cases where you might create a SAS/SPSS array to process repetitively. | A two-dimensional array; that is, a <i>vector</i> with a <i>dim</i> attribute of <i>length 2</i> . |
| Method | The analyses and/or graphs that a procedure will perform by default, that is different for different types of variables. The default settings for some procedures depend upon the scale of the variables you provide. E.g., <code>summary(temperature)</code> provides mean temperature, <code>summary(gender)</code> counts males & females. | A <i>function</i> that provides the calculation of a <i>generic function</i> for a specific <i>class</i> of object. |
| Mode | A variable's type such as numeric or character. | A fundamental property of an object. Can be numeric, character, logical, or complex. |
| Modeling function | A procedure that tests association or group differences. | A function that tests association or group differences and usually accepts a formula (e.g., $y \sim x$) and a data = <i>argument</i> . |
| NA | A missing value. | A missing value. See also <i>NaN</i> . |
| Names | Variable names. They are stored in a character variable that is part of a dataset or variable. Since R can use an <i>index</i> number instead, names are optional. Also a procedure that extracts or changes variable names. | An <i>attribute</i> of many <i>objects</i> that labels the <i>elements</i> or <i>components</i> of the object. Also the <i>function</i> that retrieves or sets this attribute. |
| NaN | A missing value. | Not a Number. Something that is undefined mathematically such as zero divided by zero. |
| NULL | An object you can use to drop variables or values. E.g., <code>x <- NULL</code> drops the variable <i>x</i> . | NULL has a zero <i>length</i> and no particular <i>mode</i> . |
| Numeric | A variable that contains only numbers. | The <i>atomic</i> mode that represents real numbers. This contains storage <i>modes</i> double, single, and integer. |
| Object | A dataset, a variable, or even a procedure. | Almost everything in R. If it has a <i>mode</i> , it is an object. Includes <i>data frames</i> , <i>vectors</i> , <i>matrices</i> , <i>lists</i> , and <i>functions</i> . |
| Object-oriented programming | A style of software in which the output of a procedure depends upon the type of data you provide it. R has an object orientation, but SAS and SPSS do not. | |
| Option | Settings that control some aspect of your R session, such as the width of each line of output. Also a function that queries or changes the settings. See also <i>par</i> . | |

Table A.1 (continued)

| | Defined in SAS/SPSS Terms | Defined in R Terms |
|-------------|---|---|
| Package | An add-on module like SAS/STAT or SPSS Advanced Models. | A collection of <i>functions</i> |
| Par | A function that queries or sets the parameters that control some aspects of traditional graphics output, like how many graphs appear on a page. | |
| 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 can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R." - http://www.r-project.org/ What is R | |
| Replacement | When you use subscripts on the left side of an assignment to change the values in an object. E.g., setting 9 to missing: <code>x[x = 9] <- NA</code> | |
| S | The language from which <i>R</i> evolved. | |
| S3, S4 | Used in the r-help files to refer to different versions of <i>S</i> . The differences between them are of importance mainly to advanced programmers. | |
| Save | Saves the datasets you choose by name. | Saves the <i>objects</i> you request to an R data file. |
| Save.image | Saves all your open datasets into a single file. | A <i>function</i> that writes all <i>objects</i> in your <i>workspace</i> to a R data file. |
| Search list | The collection of <i>databases</i> that R will search, in order, for objects. Similar to a path for your operating system. | |
| S-PLUS | The commercial version of <i>S</i> . Its main difference from <i>R</i> is that it includes a graphical user interface. | |
| Subscript | Choosing variables or values by the order in which they appear or by their name. | The extraction or <i>replacement</i> of an object using its <i>index</i> or <i>name</i> in square [brackets]. |
| Vector | A variable. It can exist on its own in memory or it can be part of a dataset. | A set of values or elements that have the same <i>mode</i> , i.e., an <i>atomic object</i> . |
| Workspace | The area of main memory where R does all its work. Data must be <i>loaded</i> into it from files and <i>packages</i> must be loaded into it from the <i>library</i> before you can use either. | |

Appendix B

A Comparison of SAS and SPSS Products with R Packages and Functions

With over 1200 add-on packages, many containing multiple procedures, R can do almost everything that SAS and SPSS can do and quite a bit more. People are releasing new packages at a rapid pace and R can give you the latest count with the following two commands. The first one uses the `available.packages` function to check Internet repositories for the packages that are currently available and store them in `myPackageNames`. The second command determines the number of unique names.

```
> myPackageNames <- available.packages()
> length (unique ( rownames (myPackageNames) ))
[1] 1449
```

So at the time of publication, there were 1449 add-on packages! If you use the `setRepositories ()` function (or `Packages > Select repositories...` on Windows) to add the bioinformatic packages, the number is even higher at 2242.

Table B.1 below focuses only on SAS and SPSS *products* and which of them have counterparts in R. As a result, some categories are extremely broad (e.g., regression) while others are quite narrow (e.g., conjoint). This list does not contain the hundreds of R packages that have no counterparts in the form of SAS or SPSS products. There are many important topics (e.g., mixed models, offered by all three) that are not listed because neither SAS nor SPSS sell a product focused just on that.

Much more detailed information about R packages is available organized in Task Views at <http://cran.r-project.org/web/views/index.html>. Another site to search by task is at <http://biostat.mc.vanderbilt.edu/s/finder/finder.html>. Detailed information about most R packages is available at <http://www.r-project.org/>, choose CRAN, then choose a mirror, then choose Packages./

Table B.1 Comparison of SAS and SPSS products to R packages

| Topic | SAS Product | SPSS Product | R Package (some are package- function) |
|--|---|--|--|
| Advanced models | SAS/STAT® | SPSS Advanced Models™ | stats, MASS, many others |
| Basics | SAS® | SPSS Base™ | R |
| Conjoint analysis | SAS/STAT®: Transreg | SPSS Conjoint™ | homals, psychoR, bayesm |
| Correspondence analysis | SAS/STAT®: Corresp | SPSS Categories™ | homals, MASS, FactoMineR, ade4, PTAk, cocorresp, vegan, made4, PsychoR |
| Custom tables | SAS Base® Report, SQL, Tabulate | SPSS Custom Tables™ | reshape |
| Data access | SAS/ACCESS® | SPSS Data Access Pack™ | DBI, foreign, RODBC |
| Data mining | Enterprise Miner™ | Clementine® | rattle, arules, FactoMineR |
| Data preparation | Various procedures | Various procedures, SPSS Data Preparation™ | dprep, various functions |
| Exact tests | SAS/STAT®: various | SPSS Exact Tests™ | coin, elm, exactLoglinTest, exactmaxsel, exactRankTests, and as options in many others |
| Genetics | SAS/Genetics®, SAS/Microarray® Solution®, JMP Genomics® | None | Bioconductor at http://www.bioconductor.org/ |
| Geographic information systems/mapping | SAS/GIS®, SAS/Graph® | SPSS Maps™ (no full GIS) | maps, mapdata, mapproj, GRASS via spgrass6, RColorBrewer, see Spatial in Task Views link above |
| Graphical user interface | Enterprise Guide® | SPSS Base™ | JGR, R Commander, pmg, SciViews |
| Graphics – interactive with linked windows | SAS/INSIGHT® | None | GGobi via rggobi, iPlots, Mondrian via Rserve |
| Graphics – static | SAS/GRAPH® | SPSS Base™ | ggplot, gplots, graphics, grid, gridBase, hexbin, lattice, plotrix, scatterplot3d, vcd, vioplot, |

Table B.1 (continued)

| Topic | SAS Product | SPSS Product | R Package (some are package- function) |
|-------------------------------|--|---|--|
| Guided analysis | SAS/LAB® | None | None |
| Matrix/linear algebra | SAS/IML®, SAS/STAT Studio® | SPSS Matrix™ | R, matlab, Matrix, sparseM |
| Missing values imputation | SAS/STAT®: MI | SPSS Missing Values Analysis™ | Hmisc - aregImpute, EMV, Design - fit.mult.impute, mice, mitools, mvnmle |
| Operations research | SAS/OR® | None | glpk, linprog, LowRankQP, TSP |
| Power analysis | SAS® Power and Sample Size Application, SAS/STAT: Power, GLM Power | SamplePower™ | asypow, powerpkg, pwr, MBESS |
| Quality control | SAS/QC® | SPSS Base™ | qcc, spc |
| Regression models | SAS/BASE® | SPSS Regression Models™ | R, Hmisc, Design, lasso, VGAM, pda |
| Sampling, complex or survey | SAS/STAT®: surveymeans, etc. | SPSS Complex Samples™ | pps, sampling, sampling, spsurvey, survey |
| Structural equations | SAS/STAT®: Calis | Amos™ | Sem |
| Text analysis | Text Miner | SPSS Text Analysis for Surveys™, Text Miner for Clementine® | Rstem, lsa, tm |
| Time series | SAS/ETS® | SPSS Trends™ Expert Modeler | Over 40 packages that do time series are described at Task View link above under Econometrics. |
| Time series, automated | SAS Forecast Studio® | SPSS Trends, DecisionTime/WhatIf™ | None |
| Trees, decision or regression | Enterprise Miner™ | SPSS Classification Trees™, AnswerTree™ | ada, adabag, BayesTree, boost, caret, GAMboost, gbev, gbm, maptree, mboost, mvpart, party, pinktoe, quantregForest, rpart, rpart. permutation, randomForest, randomForests, tree |

Appendix C

Automating Your Settings

SAS has its `autoexe.sas` file that exists to let you automatically set options and run SAS code. R has a similar file called `.Rprofile`. This file is stored in your initial working directory, which you can locate with the `getwd()` function.

Below is my `.Rprofile`. It sets options just as you would in R. See `enter help (options)` for many more. Let us step through it one command at a time.

First, I set the console width to 64 so my output fits training examples better. I also ask for five significant digits and tell it to mark significant results with stars. The latter is the default, but since many people prefer to turn that feature off, I included it. You would turn them off with a setting of `FALSE`.

```
options (width=64, digits=5, show.signif.stars=TRUE)
```

Setting the random number seed is a good idea if you want to generate numbers that are random but repeatable. That is handy for training examples in which you would like every student to see the same result. Here I set it to the number 1234.

```
set.seed (1234)
```

The `setwd` function sets the working directory, the place all your files will go if you don't specify a path.

```
setwd ("/myRfolder")
```

I also like to define the set of packages that I install whenever I upgrade to a new version of R. With these stored in `myPackages`, I can install them all with a single command. For details, see Chap. 5. This is the list of all packages used in this book.

```
myPackages <- c("car", "hexbin",  
  "ggplot2", "gmodels", "gplots", "Hmisc",  
  "reshape", "Rcmdr")
```

You can have R load your favorite packages automatically too. This is particularly helpful when setting up a computer to run R with a graphical user interface like R Commander.

Loading packages at startup does have some disadvantages though. It slows down your startup time, takes up memory in your workspace, and can create conflicts when different packages have functions with the same name. Therefore, you do not want to load too many. Loading packages at startup requires the use of the `local` function. The `getOption` function gets the names of the original packages to load and stores them in a character vector I named `myOriginal`. I then created a second character vector, `myAutoLoads`, containing the names of the packages I want to add to the list. I then merged them into one character vector, `myBoth`. Finally, I used the `options` function to change the default packages to the combined list of both the original list and my chosen packages:

```
local({
  myOriginal <- getOption("defaultPackages")
  # edit next line to be your list of favorites.
  myAutoLoads <- c("Hmisc","ggplot2")
  myBoth <- c(myOriginal,myAutoLoads)
  options(defaultPackages = myBoth)
})
```

If you want R to run any functions automatically, you create your own single functions that do the required steps. To have R run a function before all others, name it `“.First”`. To have it run the function after all others, name it `“.Last`. Notice that utility functions require a prefix of `"utils:: "` or R will not find them while it is starting up. The `timestamp` function is one of those. It simply returns the time and date. The `cat` function simply prints messages.

```
.First <- function()
{
  cat("\n                Welcome to R!\n")
  utils::timestamp()
  cat("\n")
}
```

You can also have R run any functions before exiting the package. As a Windows user, I would like to save my command history. Below I print a farewell message and then save the history to a file named `myLatest.Rhistory`.

```
.Last <- function()
{
  graphics.off()
  cat("\n\n myCumulative.Rhistory has been saved." )
  cat("\n\n Goodbye!\n\n")
  utils::savehistory(file="myCumulative.Rhistory")
}
```

Warning: Since these functions begin with a period, they are invisible to the `ls` function by default. The command `ls (all.names=TRUE)` will show them to you. Since they are functions, if you save a workspace that contains them, they will continue to operate whenever you load that workspace, even if you delete the `.Rprofile!` As usual, you can display them by typing their names and run them by adding empty parentheses to them: `.First()`. If you need to delete them from the workspace, `rm` will do it with no added arguments:

```
rm(.First, .Last) .
```

Here is the `.Rprofile` with all commands together. You can download it with the practice data sets and programs from <http://RforSASandSPSSusers.com/>

```
# Startup Settings
# Place any R commands below.
options(width=64, digits=5, show.signif.stars=TRUE)
set.seed(1234)
setwd("/myRfolder")
myPackages <- c("car", "hexbin",
               "ggplot2", "gmodels", "gplots", "Hmisc",
               "reshape", "Rcmdr")
utils::loadhistory(file = "myCumulative.Rhistory")
# Load packages automatically below.
local({
  myOriginal <- getOption("defaultPackages")
  # Edit next line to include your favorites.
  myAutoLoads <- c("Hmisc", "ggplot2")
  myBoth <- c(myOriginal, myAutoLoads)
  options(defaultPackages = myBoth)
})
# Things put here are done first.
.First <- function()
{
  cat("\n Welcome to R!\n")
  utils::timestamp()
  cat("\n")
}
# Things put here are done last.
.Last <- function()
{
  graphics.off()
  cat("\n\n myCumulative.Rhistory has been saved.")
  cat("\n\n Goodbye!\n\n")
  utils::savehistory(file="myCumulative.Rhistory")
}
```

Appendix D

Appendix D. A comparison of the major attributes of SAS and SPSS to R

| | SAS and SPSS | R |
|---|--|--|
| Aggregating data | One pass to aggregate, another to merge (if needed, SAS only), a third to use. Few basic statistics are available. | A statement can mix both raw and aggregated values. Can aggregate on all statistics. |
| Choosing data | All the data for an analysis or graph must be in a single dataset. | Analyses and graphs can freely combine variables from different data frames or other structures. |
| Choosing observations | Uses logical conditions in IF, SELECT IF, WHERE | Uses wide variety of selection by index value, variable name, logical condition (same as when selecting variables). |
| Choosing variables | Uses the simple lists of variable names in the form of: x, y, z; a to z; a-z | Uses wide variety of selection by index value, variable name, logical condition (same as when selecting observations). |
| Controlling procedure or function | Statements such as CLASS and MODEL and options control the procedure. | You can control functions by manipulating the data's structure (its class), setting function options (arguments) and using separate <i>apply</i> and <i>extraction</i> functions. |
| Converting data structures to match procedure or function | In general, all procedures accept all variables; you rarely need to convert variable type. | Original data structure plus variable selection method determines structure. You commonly use conversion functions to get data into acceptable form. |
| Cost | Each module has its price. | R and all its packages are free. |
| Data size | Most procedures are limited only by hard disk size. | Most functions must fit the data into the computer's smaller random access memory. |
| Data structure | Rectangular dataset. | Vector, factor, data frame, matrix, list, etc. |
| Graphical user interface | SAS Enterprise Miner uses flowchart approach that provides audit trail and repeatability in that form. SPSS offers well developed menus that control most things. Depends upon its language for repeatability. | R has several. R Commander looks much like SPSS. It offers easy control of the basics but is not as comprehensive as either the SAS or SPSS GUIs. Uses R language for repeatability. |

Appendix D. (continued)

| | SAS and SPSS | R |
|--------------------------|--|---|
| Graphics | SAS' are easy but relatively inflexible. SPSS Graphics Production Language (GPL) is slightly ahead of R. | Traditional graphics are extremely flexible. The <code>ggplot2</code> package provides functionality very close to GPL using a similar programming style. |
| Help and documentation | Aimed at beginner to intermediate users. | Aimed at intermediate to advanced users. |
| Macro language | A separate language used mainly for repetitive tasks or adding new functionality. User-written macros run differently from built-in procedures. | R does not have a macro language as its language is flexible enough to not require one. User-written functions run the same way as built-in ones. |
| Managing datasets | Relies on standard operating system commands to copy, delete, etc. Standard search tools can find datasets since they are in separate files. | Uses internal environments with its own commands to copy, delete, etc. Standard search tools cannot find multiple data frames if you store them in a single file. |
| Matrix language | A separate language used only to add new features. | An integral part of R that you use even when selecting variables or observations. |
| Missing data | When data is missing, procedures use all the data they can. Some procedures offer listwise deletion as an alternative. | When data is missing, functions often provide no results by default; different functions require different missing value options. |
| Output management system | People rarely use output management systems for routine analyses. | People routinely get additional results by passing output through additional functions. |
| Publishing results | See it formatted immediately in any style you choose. Quick cut and paste to word processor maintains fonts, table status, and style. Can also export to a file. | Process output with additional procedures that route formatted output to a file. You do not see it formatted as lined tables with proportional fonts until you import it to a word processor or text formatter. |
| Statistical methods | SAS is slightly ahead of SPSS but both trail well behind R. SPSS can run R programs within SPSS programs. | Most new methods appear in R around five years before SAS and SPSS. |
| Tables | Easy to build and nicely formatted but limited in what they can display. | Can build table of the results of virtually all functions but you need to view them outside R to see them nicely formatted.. |
| Variable labels | Built in. Used by all procedures. | Added on. Used by few procedures. |

Bibliography

1. Chambers, John M. *Software for Data Analysis: Programming with R*. s.l.: Springer Science+Business Media, LLC, 2008. ISBN 978-0-387-75935-7.
2. Team, R Development Core. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: <http://www.R-project.org>, 2007. 3-900051-07-0.
3. Roebuck, P. *The MATLAB Package*. 2006: <http://lib.stat.cmu.edu/R/CRAN/doc/packages/matlab.pdf>.
4. Keeling, Kellie B. and Pavur, Robert J. *A comparative study of the reliability of nine statistical software packages*. 8, May 1, 2007, Computational Statistics & Data Analysis, Vol. 51, pp. 3811–3831.
5. Bolker, Ben. [R] software comparison. *R Help Archive*. [Online] 4 16, 2007. [Cited: 3 12, 2008.]: <http://finzi.psych.upenn.edu/R/Rhelp02a/archive/97802.html>.
6. Frank E. Harrell, Jr. with contributions from many others. *Hmisc: Harrell Miscellaneous*. R package version 3.4-3. s.l.: <http://biostat.mc.vanderbilt.edu/s/Hmisc>, 2007.
7. Lemon, Jim and Grosjean, Phillippe. *prettyR: Pretty descriptive stats*. 2007. R package version 1.1-3.
8. *Bioconductor, open source software for bioinformatics*. [Online]: <http://www.bioconductor.org/whatisit>.
9. *Omegahat Project for Statistical Computing*. [Online]: <http://www.omegahat.org/>.
10. Fox, John. *car: Companion to Applied Regression*. 2007. R package version 1.2-7: <http://www.r-project.org/http://socserv.socsci.mcmaster.ca/jfox/>.
11. Venables, W.N., Smith, D.M., and the R Development Core Team. *An Introduction to R*. 2007. ISBN 3-900051-12-7.
12. SPSS, Inc. *SPSS Statistics-R Integration Package*. Chicago: SPSS, Inc., 2008.
13. Fox, John, Michael Ash, Theophilus Boye, Stefano Calza, Andy Chang, Philippe Grosjean, Richard Heiberger, G. Jay Kerns, Renaud Lancelot, Matthieu Lesnoff, Samir Messad, Martin Maechler, Duncan Murdoch, Erich Neuwirth, Dan Putler, Miroslav Ristic. *Rcmdr: R Commander*. 2008: <http://www.r-project.org/http://socserv.socsci.mcmaster.ca/jvov/Misc/Rcmdr/>.
14. Rattle: Gnome R Data Mining. *Togaware*. [Online]: <http://rattle.togaware.com/>.
15. Helbig, Markus and Urbanek, Simon. *JGR: Java GUI for R*. 2007. Version 1.5-8: <http://cran.r-project.org/doc/packages/JGR.pdf>.
16. JGR: Java GUI for R. *Dept. of Computer Oriented Statistics and Data Analysis*. [Online] 2008: <http://rosuda.org/JGR/>.
17. R-core members and Saikat DebRoy, Roger Iyengar and others. *Foreign: Red Data Stored by Minitab, S, SAS, SPSS, Stata, Systat, dBase*. 2007. R package version 0.8-23.
18. Venables, W.N. and Ripley, B.D. *Modern Applied Statistics with S*. Fourth. New York: Springer Science+Business Media, LLC, 2002. 0-387-95457-0.
19. Spector, Phil. *Data Manipulation with R*. s.l.: Springer Business+Science, 2008. ISBN 978-0-387-74730-9.

20. Baron, Jonathan. R Site Search. *Help for R: A Language and Environment for Statistical Computing and Graphics*. [Online] [Cited: February 12, 2008.]: <http://finzi.psych.upenn.edu/search.html>.
21. Romain, Francois. A firefox extension for R Site Search. [Online] [Cited: 2 12, 2008.]: <http://addictedtor.free.fr/rsitesearch/>.
22. Wickham, Hadley. *reshape: Flexibly reshape data*. 2007. R package version 0.8.0.
23. Therneau, Terry M., Atkinson, Beth and Ripley, Brian (ported). *rpart: Recursive Partitioning*. 2007. R package version 3.1-38.
24. Lumley, Thomas. *The biglm Package*. 2006: <http://cran.r-project.org/doc/packages/biglm.pdf>.
25. iPlots. *Department of Computer Oriented Statistics and Data Analysis*. [Online] 2008. [Cited: 2 11, 2008.]: <http://www.rosuda.org/iplots/>.
26. Swayne, D., et al. *GGobi: XGobi redesigned and extended*. Vols. In Proc. of the 33th Symposium on the Interface: Computing Science and Statistics, 2001.
27. Sarkar, Deepayan. *lattice: Lattice Graphics*. 2008. R package version 0.17-4.
28. Wickham, Hadley. *ggplot*. s.l.: <http://had.co.nz/ggplot2/book.pdf>, 2007.
29. Murrell, Paul. *R Graphics*. Boca Raton, FL: Chapman & Hall/CRC, 2006. ISBN 978-1-58488-486-6.
30. Cleveland, William S. *Visualizing Data*. s.l.: Hobart Press, 1993. ISBN 978-0963488404.
31. Sarkar, Deepayan. *Lattice: Multivariate Data Visualization with R*. s.l.: Springer Science+Business Media, LLC, 2008. ISBN 978-0-387-75968-5.
32. Wilkinson, Leland. *The Grammar of Graphics, Second Edition*. New York: Springer Science+Business Media, Inc, 2005. ISBN 978-0387-24544-8.
33. Friendly, Michael. *Visualizing Categorical Data*. s.l.: SAS Publishing, 2000. ISBN 978-1580256605.
34. Dan Carr, porte by Nicholas Lewin-Koh and Martin Maechler. *hexbin: Hexogonal Binning Routines*. 2006. R package version 2.3.2.
35. Warnes, Gregory R., et al. *gplots: Various R programming tools for plotting data*. R package version 2.3.2.
36. Dalgaard, Peter. *Introductory Statistics with R*. s.l.: Springer Science+Business Media, Inc., 2002. ISBN 978-0387-95475-2.
37. Warnes, Gregory R., et al. *gmodels: Various R Programming Tools for Model Fitting*. 2007. R package version 2.14.1: <http://cran.r-project.org/src/contrib/PACKAGES.htm>;<http://www.sf.net/projects/r-gregmisc>.
38. Pinheiro, Jose C. and Bates, Douglas M. *Mixed Effects Models in S and S-Plus*. New York: Springer Science+Business Media, Inc., 2000. ISBN 978-0387989570.
39. Venables, W.N. *Exegeses on Linear Models*. 1998: <http://www.stats.ox.ac.uk/pub/MASS3/Exegeses.pdf>.
40. Burns, Patrick J. *S Poetry*. 1998: <http://www.burns-stat.com/pages/Spoetry/Spoetry.pdf>.
41. Mitchell, Michael N. *Strategically using General Purpose Statistics Packages: A Look at Stata, SAS and SPSS*. Statistical Consulting Group, UCLA Academic Technology Services. 2007. Technical Report Series: <http://www.ats.ucla.edu/stat/technical-reports/>.
42. Burns, Patrick. *R Relative to Statistical Packages: Comment 1 on Technical Report Number 1 (Version 1.0) Strategically using General Purpose Statistics Packages: A Look at Stata, SAS and SPSS*. Los Angeles: UCLA Academic Technology Services, 2006. Technical Report Series: <http://www.ats.ucla.edu/stat/technicalreports/>.
43. Muenchen, Robert A. *R for SAS & SPSS Users*. 2006: <http://RforSASandSPSSusers.com>.
44. Alzola, Carlos and Harrell, Frank. *An Introduction to S and The Hmisc and Design Libraries. September 24, 2006*. <http://cran.r-project.org/doc/contrib/AlzolatHarrel-Hmisc-Design-Intro.pdf>.

45. Roebuck, P. MATLAB emulation package, version 0.8-1. *MATLAB emulation package, version 0.8-1*. 2006: <http://cran.r-project.org/doc/packages/matlab.pdf>.
46. Lumley, Gregory R. Warnes. Includes R source code and/or documentation contributed by Ben Bolker and Thomas. *gplots: Various R programming tools for plotting data*. R package version 2.3.2.
47. Hadley Wickham, Micheal Lawrence, Duncan Temple Lang, and Deborah F Swayne. An introduction to rggobi. *R-news*, Under revision. <http://ggobi.org/rggobi>.

Index

- SAS operator, 103
- SAS operator, 103
- ~ operator, 65
- !IN SPSS keyword, 111, 133
- %in% operator, 110, 111, 131, 133, 146
- %INCLUDE SAS statement, 27
- .First, 455
- .GlobalEnv, 266
- .Last, 455
- .RData file extension, 270
- .Rhistory file extension, 23, 25, 26, 272
- .Rprofile
 - file, 23, 16, 453
- / operator, 148
- | operator, 109
- : SAS operator, 101
- ^ R operator, 148
- _LAST_ SAS parameter, 113
- + R operator, 169
- <-, 42, 50

- Abline function, 369
- Accuracy of R, 3
- ADD FILES SPSS command, 189
- Add1 function, 421
- Addmargins function, 412
- Aes function, 372
- Aggregate function, 433
- AGGREGATE SPSS procedure, 214, 221
- Aggregating, 203
- AITR
 - An Introduction to R, 25
- ALL SPSS keyword, 104
- Anova function, 6, 432
- Array, 66
- as.data.frame function, 231, 242, 411
- as.logical function, 129
- as.matrix function, 152
- as.table function, 226
- as.vector function, 171
- Assignment operator, 42, 50
- Attach, 131
- Attach function, 120, 149, 264, 266
- Attaching data frames, 264
- Attaching files, 266
- Attribute, 78
- Attributes function, 262, 271, 354
- Axis function, 321

- Bar function, 355
- Barplot function, 289
- Bartlett.test function, 424
- Batch processing, 27
- Biglm package, 259
- Bioconductor*, 17
- Boxplot function, 323
- By function, 206, 208, 214, 324
- BY processing example programs, 208
- BY processing, SAS, 208
- BY SAS statement, 243

- c function, 57, 173
- Car package, 19, 180, 433
- Cases, 58
- Cat function, 454
- cbind function, 57
- CD SPSS command, 269, 330
- chisq.test function, 411
- Class, 73, 82
- Class function, 66, 69, 226, 262
- CLASS SAS statement, 225
- Cleanup.import function, 268
- Coercion, 51, 56
- colClasses argument, 83
- Comments, in programs, 62

- Complete.cases function, 167
- Components, of a list, 60
- Contents function, 44, 263
- CONTENTS SAS procedure, 263
- Conversion
 - class removal, 144
 - data frame to matrix, 144
 - from logical vector to index
 - when selecting variables, 143
 - index to logical, 146
 - list to separate vectors, 144
 - list to vector, 144
 - lists or dataframes into list, 144
 - matrix to data frame, 144
 - matrix to vector, 144
 - variable names to indexes, 115
 - vector to matrix, 144
 - vectors into one long one, 144
 - vectors to columns of a matrix, 144
 - vectors to data frame, 144
 - vectors to rows of a matrix, 144
- coord_polar function, 390
- Coplot function, 310
- Cor function, 37, 38, 58, 416, 439
- coord_flip function, 346
- Correlation
 - Kendall, 413
 - Pearson, 434
 - Spearman, 439
- CRAN, 17, 29
- CROSSTABS SPSS procedure, 200
- Crosstabulation
 - column proportions, 411
 - row proportions, 411
- CTABLES SPSS procedure, 200
- cut2 function, 180
- Data
 - generating a data frame, 251
 - generating continuous measures, 249
 - generating factors, 246
 - generating integer measures, 248
 - generating numeric sequences, 245
 - generating repetitive patterns, 247
 - generation, 251
 - storage and memory considerations, 263
- Data argument, 65
- Data editor, 86
- Data frame, 67, 81
- Data function, 18
- Datasets
 - reshaping, example programs, 219
 - adding, 186
 - concatenating, 186
 - joining, 190
 - listing in all packages, 19
 - merging, 190
 - stacking, 186
- data.frame function, 56, 116
- DATASET NAME SPSS command, 103
- DELETE VARIABLES SPSS command, 185
- Deleting objects. *See* removing objects,
- Demo function, 276
- Describe function, 14, 407
- DESIGN SPSS keyword, 65
- Detach function, 14, 114
- Dim function, 57
- Directory, setting. *See* working directory
- do.call function, 207, 215
- DROP SAS statement, 185
- drop1 function, 421
- Duplicate observations, 210
- Edit function, 79
- Editor
 - JGR data editor, 37, 40
 - R Editor, 22, 24, 37, 173
 - SAS Program Editor, 22, 24
 - SPSS Syntax Editor, 22, 24
- Elements, of a vector, 51
- Exiting R. *See* quitting R
- Exp function, 148
- EXPLORE SPSS procedure, 406
- Exporting data
 - example programs, 95
- Exporting data, 97
- Expression function, 325
- Extracting parts of objects
 - getting help on, 42
- Extractor functions, 67–69, 418, 444
- Factor, 82, 56, 66, 266
 - character, 226
 - converting many variables to, 232
 - converting to variables, 228
 - dropping unused levels, 233
 - numeric, 247
 - ordered, 229
- file.show function, 101
- FILTER SPSS command, 124
- First observation per group, 214
- FIRST SPSS keyword, 214
- First.variable SAS variable, 214
- Fix function, 79
- Foreign package, 98

- FORMAT SAS procedure, 225
- FORMAT SAS statement, 225
- Formats, 225
- Formulas, 64
- FREQ SAS procedure, 200
- Function, 63
 - arguments, 63
 - controlling with arguments, 62
 - controlling with class, 65
 - controlling with formulas, 64
 - generic, 41, 44, 66, 67, 143, 285, 319, 444, 445, 446
 - n, SAS or SPSS, 156
 - visible vs. non-visible, 67
 - writing your own, 73–75
- Function (drop “s”), 14

- Generating data. *See* data generation
- Generic function. *See* function, generic
- Geom_bar function, 355
- Geom_jitter function, 378
- Geom_segment function, 363
- GET FILE SPSS command, 113
- getOption function, 454
- getwd function, 61, 268, 453
- getwd function, 453
- ggplot function, 374
- ggplot2 package, 341, 386
- ggsave function, 278, 385
- gl function, 246, 253, 254, 255
- gmodels package, 409, 437
- GOPTIONS SAS statement, 277, 290
- GPL. *See* graphics:Graphics Production Language
- Graphics
 - comparing R’s packages, 278
 - density countours, 366
 - devices, 277
 - File> Export SPSS menu, 277
 - GGobi, 274
 - ggplot2 package, 274, 278
 - ggplot2, example programs, 387
 - Grammar of Graphics, 275–276, 341, 342, 347, 387, 389
 - example programs, 387
 - grid graphics system, 307
 - history recording, 278
 - lattice, 310
 - lattice package, 275, 310
 - overview, 273
 - procedures vs. systems, 277
 - SAS/GRAPH, 273
 - SAS/INSIGHT, 274
 - SPSS, 274
 - traditional, 274, 281–339
 - traditional graphics system, 277
 - traditional, example programs, 331
- Graphics Production Language, 274, 341, 374, 458
- Graphics, ggplot2
 - aesthetics, defined, 342
 - aspect ratio, 382
 - axes, logarithmic, 381
 - bar charts, 344
 - bar charts with subgroups, 348
 - box plots, 376
 - box plots with jittered points, 378
 - coordinate system, defined, 342
 - density curve, 355
 - dot charts, 352
 - elements and parameters, summary, 386
 - error bar plots, 337
 - facets, defined, 342
 - geoms, defined, 342
 - histograms, 354
 - labels, 353
 - linear fit by groups, 373
 - multiple plots on a page, 382
 - normal QQ plots, 299
 - overview of qplot and ggplot, 342
 - pie charts, 347
 - plot symbols, setting by groups, 372
 - plots by group or level, 309
 - point display variations, 361
 - presummarized data, 351
 - scales, defined, 342
 - scatterplot matrix, 374
 - scatterplot with density contours, 366
 - scatterplots, 361
 - scatterplots faceted by group, 374
 - scatterplots with fit lines, 367
 - scatterplots with jitter, 363
 - scatterplots with large datasets, 364
 - scatterplots with reference lines, 368
 - statistics, defined, 342
 - strip plots, 360
 - titles, 353
- Graphics, traditional
 - abline function, 307, 308, 309, 369
 - adding titles, labels, colors, legends, 288
 - adj parameter, 327
 - arrows function, 327
 - ask parameter, 291
 - axis function, 321
 - bar plots, 337
 - barplot, 281

Graphics, traditional (cont.)

- barplots of counts with subgroups, 281
- barplots of means, 286
- box function, 327
- box plots, 322
- cex parameter, 327
- col parameter, 327
- coplots, 309
- demonstration plot, 147
- density, 366
- dot charts, 352
- dual-axes plots, 320
- error bar plots, 380
- family parameter, 326
- font parameter, 327
- formula and symbol display, 324
- graphics parameters, 325
- grid function, 350
- histograms, 293
- histograms overlaid, 297
- identifying points, 311
- interaction plots, 324
- jitter, 304
- las parameter, 327
- linear fit, 308
- linear fit by group, 308
- lines function, 294
- lty parameter, 327
- lwd argument, 327
- main sub title argument, 324
- main title argument, 326
- mar parameter, 326
- mfcoll parameter, 326
- mfrow parameter, 326
- mosaic plot, 275, 286
- mtxt function, 328
- multiple plots on a page, 290
- new parameter, 326
- normal QQ plots, 299
- options and elements table, 325
- par function, 297
- pch parameter, 327
- pie charts, 347
- ps parameter, 326
- scatterplot matrices, 318
- scatterplots, 303
- scatterplots with confidence and prediction intervals, 312
- scatterplots with confidence ellipse, 311
- scatterplots with jitter, 304
- scatterplots with large datasets, 305
- scatterplots with lines, 307

- spine plot, 348
- srt parameter, 327
- strip charts, 301
- text function, 321
- types of point displays, 304
- usr parameter, 289, 326
- xlab argument, 326
- xlog parameter, 326
- ylab argument, 326
- ylog parameter, 326

Grep function, 110, 111, 132

Hat function, 325, 328, 338
 Head function, 18, 214, 264, 290

Help

- examples, 42
- extracting and replacing elements, 42
- for datasets, 45
- for generic functions, 44
- for packages, 44
- help files, 41
- running help examples, 32
- searching, 46
- via mailing lists, 45
- via web searches, 46
- vignettes, 47

help.search function, 42
 help.start function, 41
 Hexbin package, 307
 Hist function, 297
 Hmisc package, 13, 43, 45, 95, 180, 219, 239, 263, 326

ID variable. *See* row names

Identify function, 301, 303, 304

IML SAS product, 6

Importing data

- from SAS, 95
- from SAS, example programs, 95
- from SPSS, 96
- from SPSS, example programs, 96

IN SAS operator, 111, 133

INCLUDE SPSS command, 27

Inputting text files. *See* reading text files

install.packages function, 13, 16, 33

Installation

- of packages, 15
- of R, 12

interaction.plot function, 324

is.na function, 166

JGR user interface, 36

KEEP SAS statement, 185

- Keywords, SPSS, 51
- kruskal.test function, 432, 433
- Label function, 239
- LABEL SAS statement, 239
- Lapply function, 154, 169, 207
 - compared to do.call, 207, 215
- Last observation per group, 214
- LAST SPSS keyword, 214
- last.variable SAS variable, 214
- Lattice package, 275, 277, 290, 307, 341
- Layout function, 290
- Length function, 155, 157
- LENGTH SAS statement, 95
- levene.test function, 424
- LIBNAME SAS statement, 269
- Library function, 13, 14
- Library, for SAS formats, 225
- Lines function, 294, 308, 316, 319
- List, 41, 45, 59
 - created by lm, 68
 - created in your function, 66
- List function, 59, 75, 92, 145, 287
- lm function
- lm function, 5, 64, 68, 262, 308, 314, 417, 418
- Load function, 23, 25, 26, 61, 261, 270
- Loadhistory function, 26, 272
- Loading a package, 14, 33
- Local function, 454
- Log function, 147
- log10 function, 381
- Logical comparisons
 - to missing values, 89, 129, 159, 165, 166,
- Logical operators, 159
- Long dataset format, 190
- Lowess function, 308, 367
- ls function, 60, 61, 261, 262, 263, 264, 455
- ls.str function, 263, 271
- Macro. *See* function
- Macro language, 5, 90, 458
- Macro substitution, 90–91, 92
- Mailing lists, 45, 103
- Mapply function, 157
- Masking functions, 432
- MASS
 - Modern Applied Statistics in S, 45, 403
- MATCH FILES SPSS procedure, 214
- MATLAB, 2, 451
- Matlines function, 313, 316
- Matrix, 67, 68
 - character, 68
 - numeric, 68
- Matrix language, 5, 6, 441, 458
- Matrix, SPSS product, 5
- Mean function, 42, 58, 63, 67, 68, 147, 152, 195, 197, 204, 286
- MEANS SAS procedure, 153
- Median function, 408, 427, 433
- Merge function, 191, 192, 193, 199
- Methods function, 66, 319, 441
- Missing values. *See* NA
 - example programs to assign, 169
- MISSOVER SAS option, 89
- Mode, 21, 27, 51, 57, 66, 68, 86, 144, 196, 200, 205, 300, 326, 418, 446
- Model object, 68, 70
- MODEL SAS statement, 65
- Mosaic function, 286
- mtext function, 321, 328
- N function, SAS and SPSS, 155, 156, 158
- NA, missing value, 30, 52, 53, 54, 56, 63, 79, 80, 82, 146, 153, 167
- na.omit function, 167, 421
- na.rm argument, 63, 153
- Names function, 56, 70, 106, 115, 127, 128, 132, 134, 172, 174, 262, 298, 317, 415
- Naming objects, 60
- ncol function, 107
- NODUPKEY SAS option, 210
- Noit SAS option, 65
- Not run, 43
- nrow function, 127
- NULL object, 185
- NVALID SPSS function, 155
- Object, 61
- Objects function, 68, 261
- Observations, 56
 - converting to variables, 232
 - selecting, 42, 103, 123–140, 295
 - selecting all, 124
 - selecting by index number, 124
 - selecting by row name, 127
 - selecting by string search, 132
 - selecting in SAS and SPSS, 123
 - selecting using logic, 108–110
 - selecting with subset function, 114
- ODS. *See* Output Delivery System
- OMS. *See* Output Management System
- Operators, mathematical, 148

- Options, 486
 - scipen, 404, 415
- Options
 - digits, 403, 415
 - number of significant digits, 403, 453
 - SAS, linesize, 403, 437
 - SAS, probsig, 403
 - setting automatically, 444
 - SPSS, small, 5, 222, 403
 - SPSS, width, 50
 - width, 50
- Options function, 50
- Order function, 221, 222
- Ordered factor, 225, 229
- Ordinal data, 225, 229
- Output Delivery System, SAS, 5, 68, 417
- Output management, 5, 6, 68, 417, 441
- Output Management System, SPSS, 5, 68, 417, 458
- Packages, 18
- Pairs function, 67, 319, 320
- Pairwise.t.test function, 429, 433
- Pairwise.wilcoxon.test function, 433
- par function, 290, 302, 307, 322, 326, 341
- Paste function, 108, 176
- Plot function, 5, 67, 281
- plotmeans function, 324
- Points function, 294
- Predict function, 315, 316
- prettyR package, 14, 156
- Print function, 27, 31, 51, 69, 70, 74, 116, 135, 145, 262, 383
 - as related to SPSS, 31, 52
- PROC SORT NODUPKEY, 212
- Procedures, SAS or SPSS, 73
- Production Facility, SPSS, 28
- Programmability Extension, SPSS, 23
- Programming syntax, 24
- prop.table function, 407, 411, 412
- PRX SAS function, 110, 132

- qplot function, 341, 342, 343
- qq.plot function, 300
- qqnorm function, 301
- quit function, 23, 26, 32
 - in SPSS-R programs, 28
- Quitting R, 26

- R Commander user interface, 33
- R, running
 - from JGR interface, 36
 - from R Commander, 33, 34, 35
 - from Rattle interface, 34
 - from within SPSS, 27
 - in batch mode, 27
 - in standard R interface, 31, 343
 - programs that include programs, 31
- Rattle package, 34
- Rattle user interface, 34
- rbind function, 187
 - used with do.call, 207
- rcorr function, 413, 414
- read.fwf function, 89, 93
- read.table function, 79, 81, 82, 89, 165, 166, 226
- Reading data from keyboard, 86
- Reading data within a program
 - example programs, 85
- Reading text files
 - 2 records per case
 - example programs, 94
 - comma separated values, 80
 - delimited, 79
 - example programs, 80
 - one record per case, 87
 - skipping columns, 82
 - two or more records per case, 92
 - within a program, 84
- Recode function, 180, 432
- Records, 55
- Regression, linear, 31, 64, 68
- Regular expressions, 110, 112, 132, 133, 262
- Removing objects, 185, 216
- Rename function, 171, 199
- Renaming
 - columns, 37, 175
 - rows, 177
 - variables, 37, 174
 - variables, example programs, 177
- rep function, 247, 254
- Replacing parts of objects
 - getting help on, 43
- Repositories, 17
- Reshape package, 201, 217
- rm function, 61, 267, 268, 313
 - regarding First. and Last., 455
- Round function, 407, 412
- row names, 56
- rownames function, 449
- row.names argument, 89, 177, 190
- row.names function, 56, 127, 128, 132, 134, 317
- rpart function, 180
- rpart package, 180
- Running R, *See* R, running

- Sample function, 248
- Sapply function, 155, 182, 231, 408
- sasxport.get function, 95
- SAS/IML SAS Product, 6
- Save function, 270, 278, 385
- save.image function, 22, 26, 150, 270
- savehistory function, 272
- Saving
 - data and functions, 22, 24, 26
 - history (journal), 271
 - output, 271
 - program and output, 22, 24, 26
 - programs, 271
 - workspace, 269
- Scale function, 198
- Scan function, 87
- Scientific notation, 403
- sd function, 155
- Search, 14
- Search function, 14, 264
- Search path, 264
- SELECT IF SPSS Command, 124
- Selecting observations
 - example programs, 135
- seq function, 246
- SET SAS statement, 186
- SET SPSS command, 404
- setRepositories() function, 449
- setwd function, 61, 269, 453
- sink UNIX command, 2, 25
- slashes, in filenames, 61
- Sorting
 - data, 221
 - example programs, 223
- Source function, 25, 27, 31
 - as related to SPSS, 31
- SPLIT FILE processing example
 - programs, 208
- SPLIT FILE SPSS command, 204
- Split function, 186
- split.screen function, 290
- spss.get function, 96
- spsspivotable.Display function, 31
- SPSS-R Integration Package, 28, 29
- sqrt function, 148
- Statements, SAS, 51
- Statistics
 - nested, 65
 - analysis of covariance, 65
 - Analysis of variance, 427
 - post hoc tests, 436
 - specifying interactions, 430
 - sums of squares, 431
 - Chi-squared test, 408
 - Correlations, 413
 - Crosstabulation, 438
 - Descriptive, 404
 - Example programs, 433, 434
 - Kruskal-Wallis test, 432
 - Mann-Whitney U test, 425
 - Models, comparing, 421
 - Overview, 481
 - percentages, 407, 412
 - Predictions on new data, 422
 - Proportions, 406, 411
 - Regression, linear, 417
 - interaction, 65
 - partial tests, 419
 - QQ plot, 421
 - residual plot, 421
 - residuals-leverage plot, 420, 421
 - scale-location plot, 421
 - t-tests on parameters, 419
 - without intercept, 65
 - diagnostics, 420
 - sequential tests, 420
 - t-tests
 - for independent groups, 422
 - paired, 424
 - Variance, equality testing, 424
 - Wilcoxon rank sum test, 425
 - Wilcoxon signed rank test for paired
 - groups, 426
- Step function, 421
- stepAIC function, 421
- str function, 262, 263
- stringsAsFactors argument, 56
- stripchart function, 301, 303
- subset function, 114
- Summarizing, 157
- Summary function, 406, 407
 - from within SPSS, 31
 - regarding variable labels, 239
- SUMMARY SAS procedure, 201, 221
- t.test function, 424, 426
- Table function, 58, 81, 282, 286
- Table object, 65
- TABULATE SAS procedure, 200
- Tail function, 18, 262
- Tapply function, 196
- TEMPORARY SPSS command, 124
- Text files, viewing, 87
- Text files, reading. See reading text files
- Text function, 317, 325
- textConnection function, 84, 85

- Timestamp function, 454
- TO SPSS keyword, 100
- Transcript of code and results, 26
- Transform function, 148
- Tree function, 180
- TukeyHSD function, 430
- type. *See* mode

- Unclass function, 70, 144
- Uninstalling
 - packages, 16
 - R, 16
- UNIVARIATE SAS procedure, 406
- Unique function, 449
- Unlist function, 144
- Unload packages. *See* detach function
- update.packages function, 15
- Updating packages, 15
- USE ALL SPSS command, 124

- VALUE LABEL SPSS command, 150
- Value labels, 225
 - example programs, 234
- var function, 408, 428
- var.test function, 424
- VARIABLE LABELS SPSS command, 239
- VARIABLE LEVEL SPSS command, 225
- Variables, 65
 - converting to observations, 217
 - dropping, 185
 - keeping, 185
 - labels, 239
 - recoding, 180
 - renaming, 56, 171
 - selecting, 103
 - selecting all variables, 104
 - selecting by column name, 125107
 - selecting by index number, 104
 - selecting by list index, 115
 - selecting by simple name, 113
 - selecting by string search, 110
 - selecting in SAS and SPSS, 103
 - selecting to save in new dataset, 116
 - selecting using \$ notation, 112
 - selecting using logic, 108
 - selecting with subset function, 114
 - selection example programs, 116
- Vector, 51
 - character, 68
 - character or string, 51
 - numeric, 63, 73

- Where function, 130
- WHERE SAS statement, 103, 123
- Which function, 162
- Wide dataset format, 217
- wilcox.test function, 426, 427
- with function, 51, 84, 114, 120, 147, 196, 208, 404
- within function, 147
- Working directory, 61, 272
- Workspace
 - managing, 261
 - minimizing, 268
 - saving, 269
- write.foreign function, 98
- write.table function, 97
- writing text files, 99

- Z score, 198