Creating R Packages, Using CRAN, R-Forge, And Local R Archive Networks And Subversion (SVN) Repositories

Spencer Graves
PDF Solutions
San José CA
spencer.graves@pdf.com

Sundar Dorai-Raj
Google
Mountain View CA
sdorairaj@google.com

Motivation

- **R is the language of choice for a large and growing proportion of people developing new statistical algorithms**
- **Comprehensive R Archive Network (CRAN) makes it easy to benefit from others' work and to share your work and get feedback on potential improvements**
- **Creating R packages**
  - Provides a system for creating software with documentation including unit tests, and thereby
  - Increases software quality & development productivity
- **Local R Archive Networks can increase your productivity in developing new code and sharing it with coworkers**
- **R-Forge and local Subversion (SVN) repositories make collaboration on joint software development easy & productive**

Outline

- ■ Installing R and R Packages
  - From CRAN
  - From a local package
  - From alternative repositories
  - Getting help
- ■ Obtaining source code
- ■ Creating R packages
- ■ Establishing and Maintaining Local R Archive Networks
- ■ Using Subversion (SVN)

PDF/SOLUTIONS
Yield, Performance, Profitability

Installing R And R Packages

- Installing R from CRAN
- Installing R contributed packages from
  - CRAN
  - local package
  - alternative repositories

PDF/SOLUTIONS
Yield, Performance, Profitability

Installing R

- www.r-project.org
- CRAN
- (select a local repository)
- Download an appropriate precompiled version or package source to suit your operating system
- Configure ...
  - R Installation and Administration manual http://cran.r-project.org/doc/manuals/R-admin.pdf
  - modify default options in "~R/etc/Rprofile.site":
    - default repositories (including local?)
    - max.print
    - ...

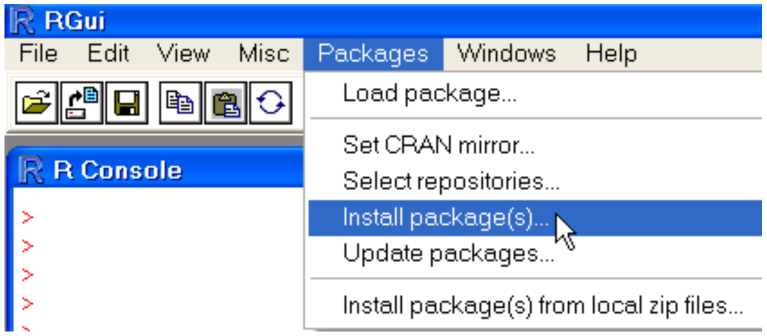
```
options(repos = c(CRAN = "http://cran.cnr.berkeley.edu",
                  CRANextra = "http://www.stats.ox.ac.uk/pub/RWin"),
        max.print=222)
```

PDF SOLUTIONS
Yield, Performance, Profitability

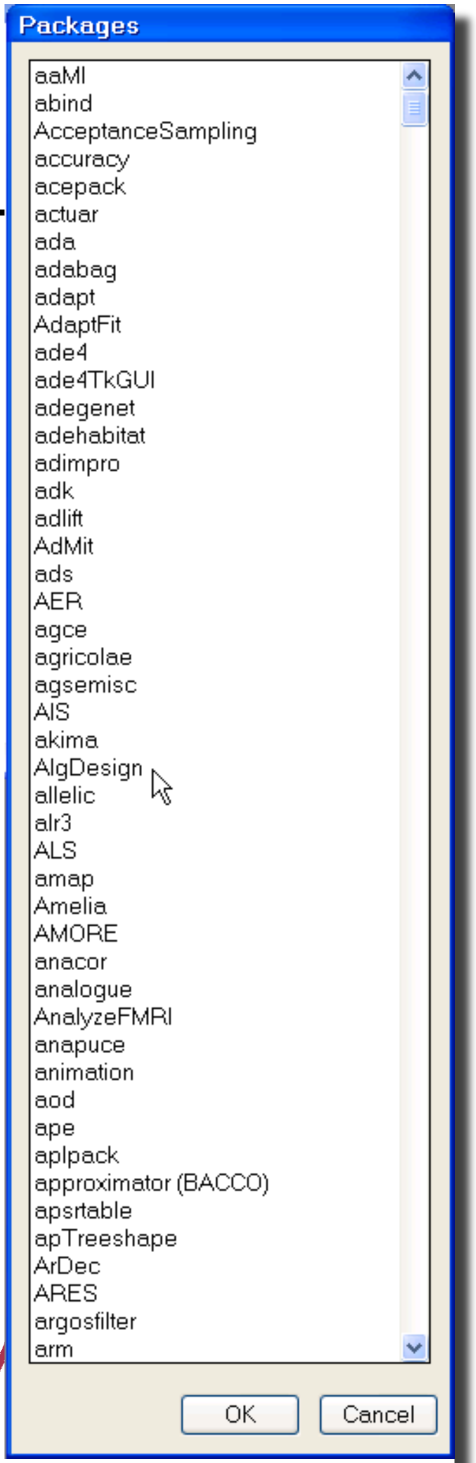
system.file()

Installing R Packages From CRAN

- install.packages('packageName')
- OR in Rgui:



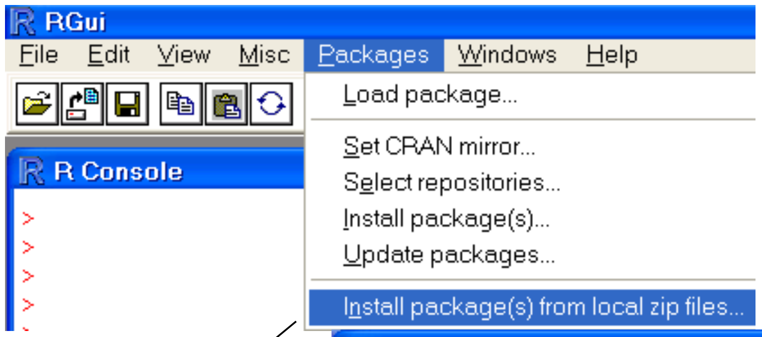
  - select a local repository (if needed)
  - select package(s) from list



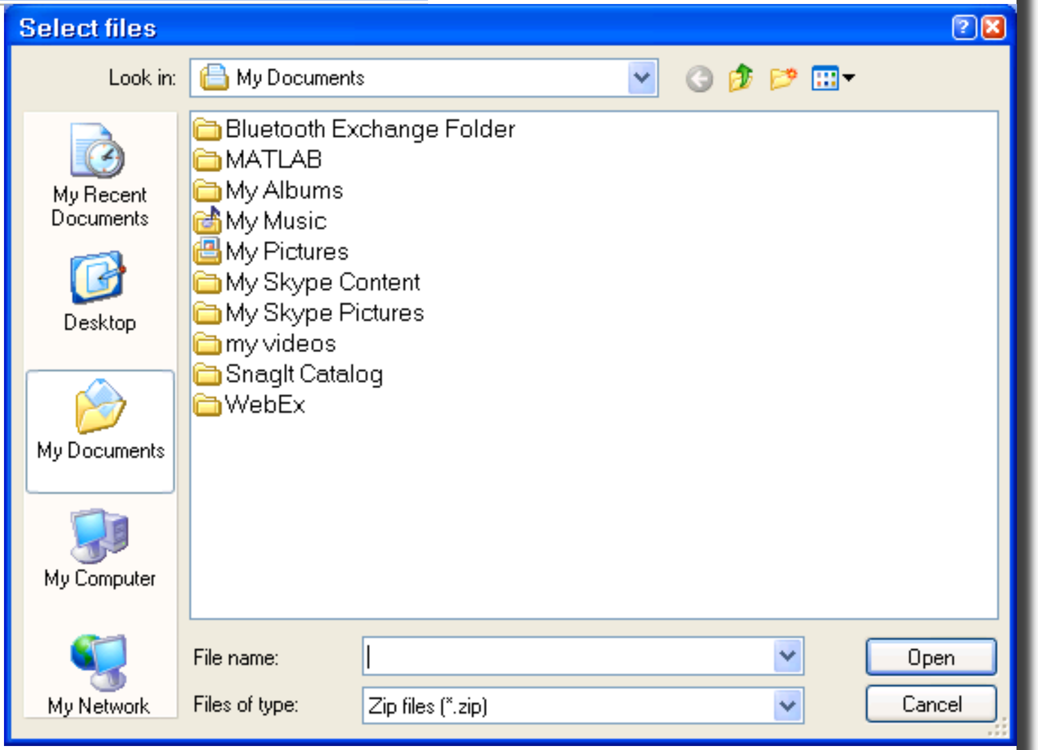
aaMI
abind
AcceptanceSampling
accuracy
acepack
actuar
ada
adabag
adapt
AdaptFit
ade4
ade4TkGUI
adegenet
adehabitat
adimpro
adk
adlift
AdMit
ads
AER
agce
agricolae
agsemisc
AIS
akima
AlgDesign
allelic
alr3
ALS
amap
Amelia
AMORE
anacor
analogue
AnalyzeFMRI
anapuce
animation
aod
ape
aplpack
approximator (BACCO)
apsrtable
apTreeshape
ArDec
ARES
argosfilter
arm

Installing R Packages From Local Zip Files (Windows)

- in Rgui:



- find packageName.zip



PDF SOLUTIONS
Yield, Performance, Profitability

Or From R Command Prompt (Any OS)

- **Windows binary**
  - install.packages(“packageName.zip”, repos = NULL)
- **Any OS provided appropriate tools for compiling source are available**
  - install.packages(“packageName.tar.gz”, repos = NULL)
  - Windows requires “Rtools”
    - http://www.murdoch-sutherland.com/Rtools/
  - Mac requires Xtools
  - For most Linux/UNIX systems the required toolsets are available

Getting Help

- **?functionName**
  - help pages for packages in the search path
- **Fuzzy search**
  - “help.search” function
  - www.r-project.org → search or ‘RSiteSearch’ function
- **Other R search engines and R Wiki**
- **Google**
- **r-help listserve**
  - PLEASE do read the posting guide http://www.R-project.org/posting-guide.html and provide commented, ***minimal, self-contained, reproducible code*.**
  - Reading “r-help”, “r-devel”, “r-sig-____” is like attending a professional meeting a few minutes a day

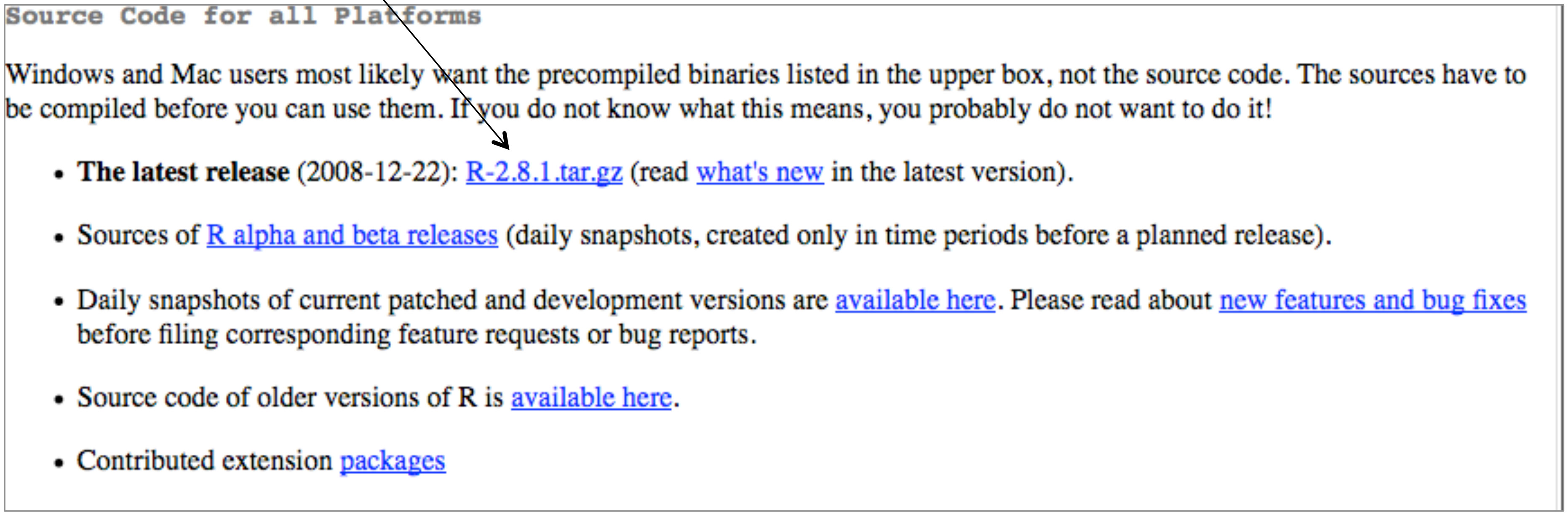
Outline

- Installing R and R Packages
- ➡ Obtaining source code
- Creating R packages
- Establishing and Maintaining Local R Archive Networks
- Using Subversion (SVN)

PDF/SOLUTIONS
Yield, Performance, Profitability

Obtaining Source Code For R

- www.r-project.org → CRAN → (select a repository)
- For R:



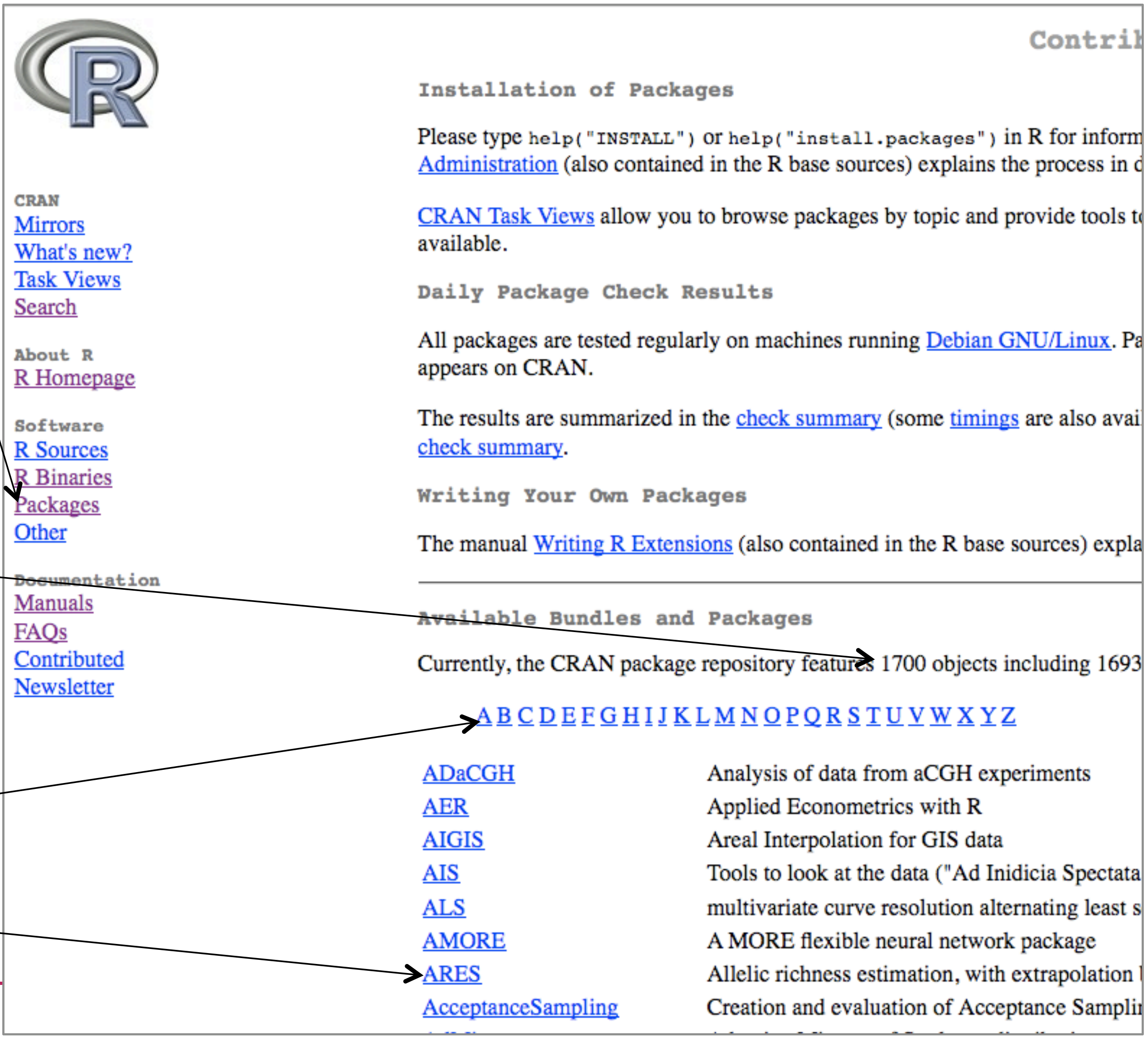
Source Code for all Platforms

Windows and Mac users most likely want the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- **The latest release** (2008-12-22): R-2.8.1.tar.gz (read what's new in the latest version).
- Sources of R alpha and beta releases (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are available here. Please read about new features and bug fixes before filing corresponding feature requests or bug reports.
- Source code of older versions of R is available here.
- Contributed extension packages

Obtaining Source Code For A Package

- **Load CRAN in browser**
  - Click “Packages” link
  - 1700 objects including packages (as of 2009-03-11)
  - Find the package of interest by first letter
  - click name



Contri

CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed
Newsletter

Installation of Packages

Please type `help("INSTALL")` or `help("install.packages")` in R for inform
Administration (also contained in the R base sources) explains the process in d

CRAN Task Views allow you to browse packages by topic and provide tools t
available.

Daily Package Check Results

All packages are tested regularly on machines running Debian GNU/Linux. Pa
appears on CRAN.

The results are summarized in the check summary (some timings are also avai
check summary.

Writing Your Own Packages

The manual Writing R Extensions (also contained in the R base sources) expla

Available Bundles and Packages

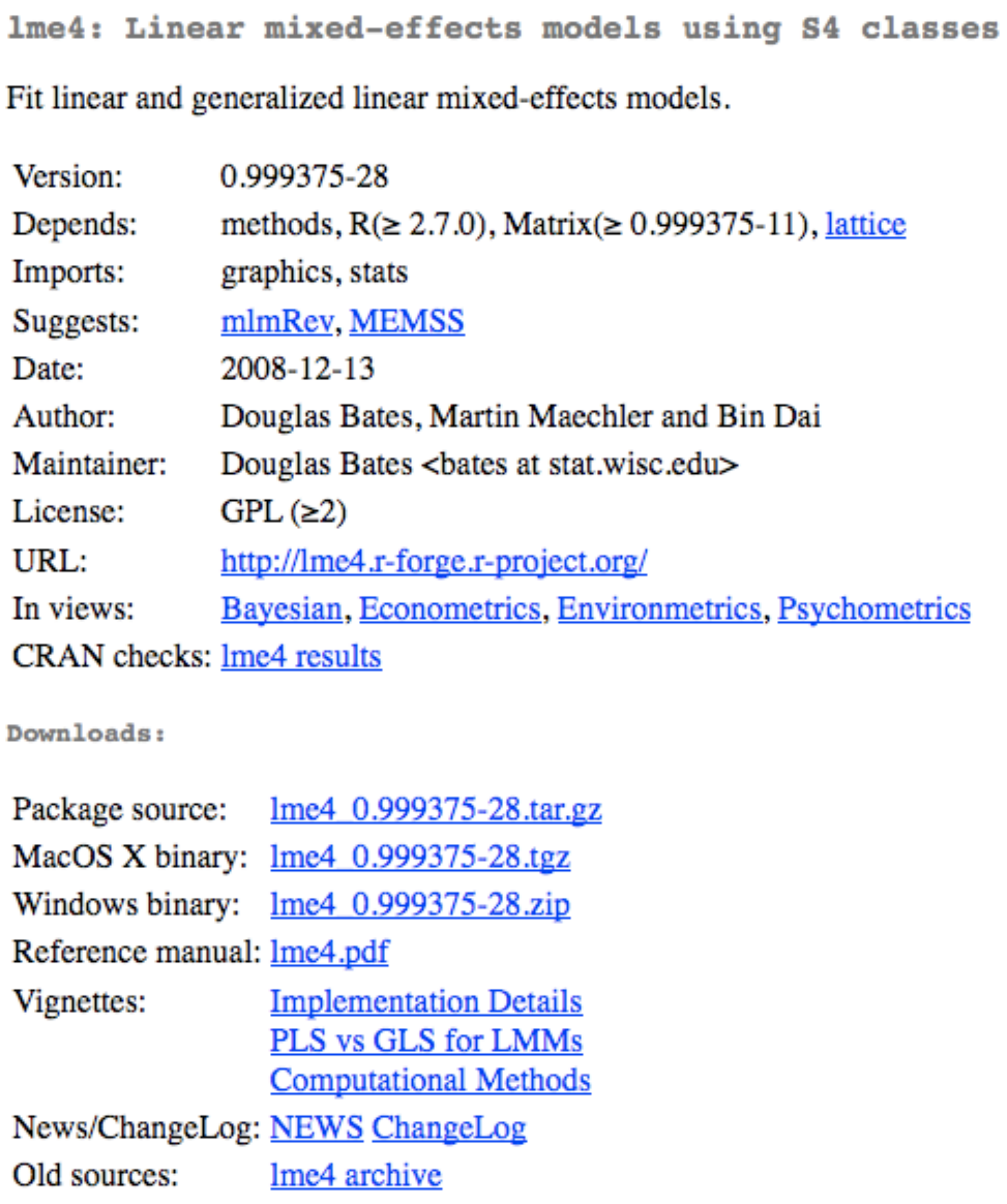
Currently, the CRAN package repository features 1700 objects including 1693

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

ADaCGH	Analysis of data from aCGH experiments
AER	Applied Econometrics with R
AIGIS	Areal Interpolation for GIS data
AIS	Tools to look at the data ("Ad Inidicia Spectata
ALS	multivariate curve resolution alternating least s
AMORE	A MORE flexible neural network package
ARES	Allelic richness estimation, with extrapolation
AcceptanceSampling	Creation and evaluation of Acceptance Sampli

“lme4” Package

- **Package pages contain links to:**
  - Package dependencies
  - Package source
  - Package binaries
  - Reference manual
  - Archives for old source tarballs
  - Maintainer contact info
  - And, if applicable,
    - Project URL
    - Task Views
    - Vignettes



lme4: Linear mixed-effects models using S4 classes

Fit linear and generalized linear mixed-effects models.

Version:	0.999375-28
Depends:	methods, R(≥ 2.7.0), Matrix(≥ 0.999375-11), lattice
Imports:	graphics, stats
Suggests:	mlmRev, MEMSS
Date:	2008-12-13
Author:	Douglas Bates, Martin Maechler and Bin Dai
Maintainer:	Douglas Bates <bates at stat.wisc.edu>
License:	GPL (≥2)
URL:	http://lme4.r-forge.r-project.org/
In views:	Bayesian, Econometrics, Environmetrics, Psychometrics
CRAN checks:	lme4 results

Downloads:

Package source:	lme4_0.999375-28.tar.gz
MacOS X binary:	lme4_0.999375-28.tgz
Windows binary:	lme4_0.999375-28.zip
Reference manual:	lme4.pdf
Vignettes:	Implementation Details PLS vs GLS for LMMs Computational Methods
News/ChangeLog:	NEWS ChangeLog
Old sources:	lme4 archive

Using An Installed Package

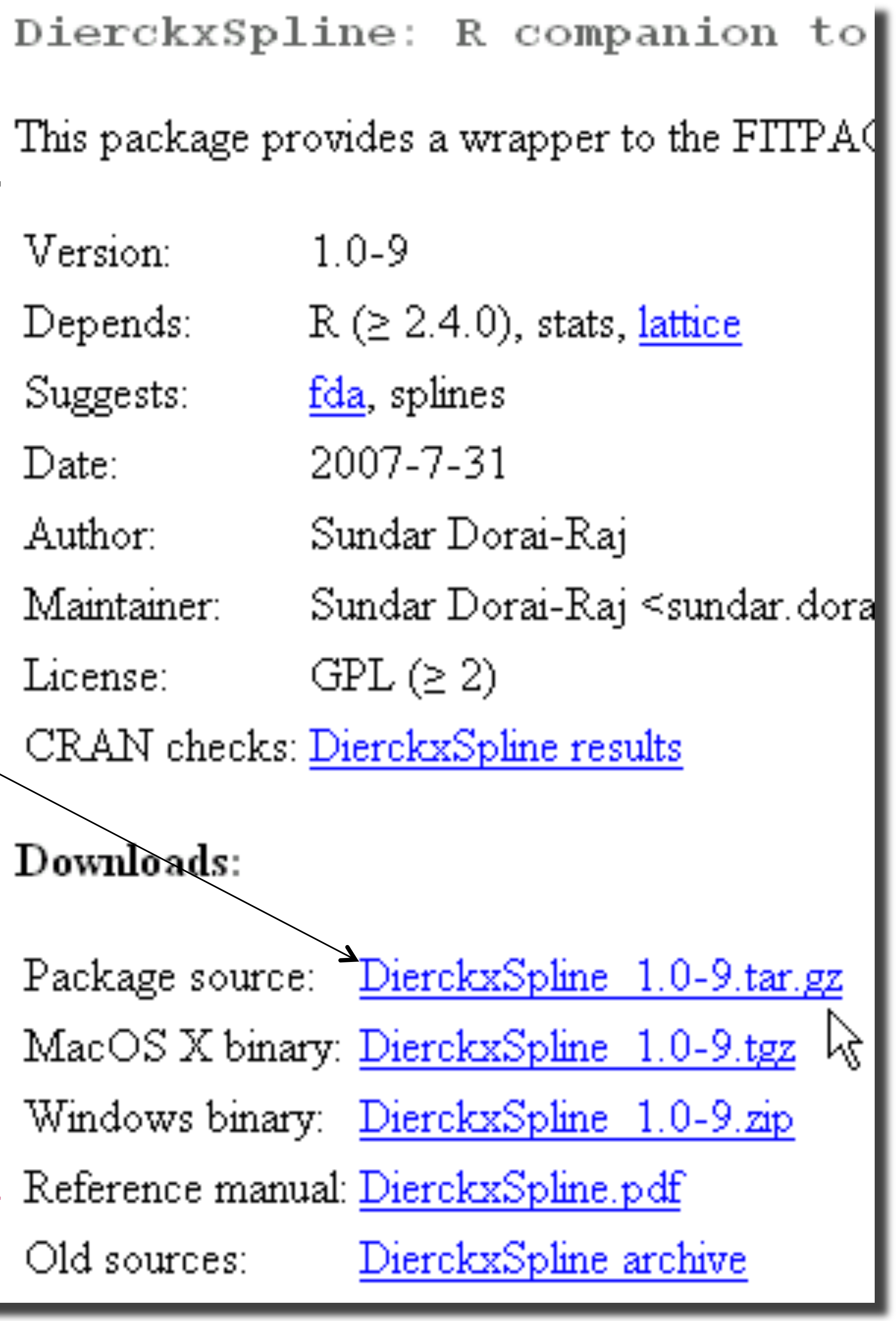
- **help(package = fortunes) or library(help = fortunes)**
  - to get an overview of package capabilities
- **library(fortunes)**
  - to attach it as the second in the search path
- **?fortune**
  - to get ‘help’ on the function ‘fortune’

```
> fortune('RTFM')

This is all documented in TFM. Those who WTFM don't want
to have to WTFM again on the mailing list. RTFM.
   -- Barry Rowlingson
      R-help (October 2003)
```

“DierckxSpline” Package

- **Click**
  - Download to your hard drive
  - Unzip



DierckxSpline: R companion to

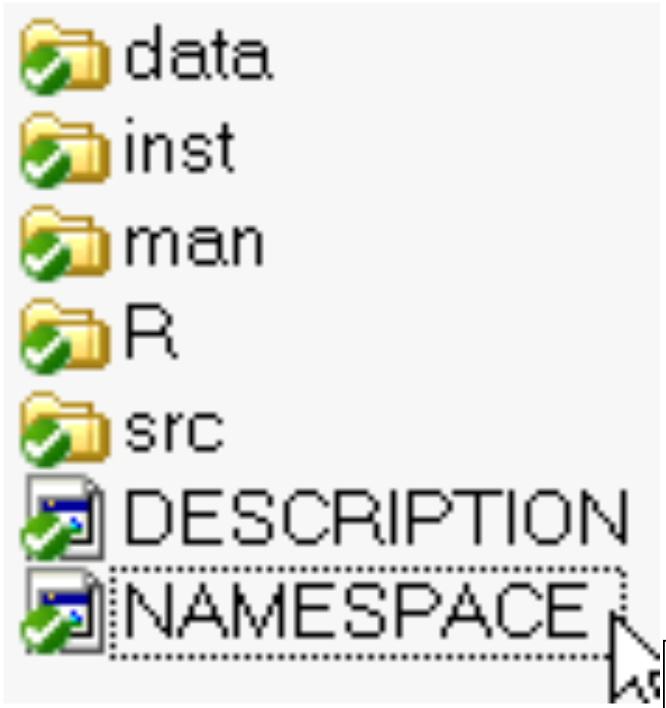
This package provides a wrapper to the FITPA

Version:	1.0-9
Depends:	R (≥ 2.4.0), stats, lattice
Suggests:	fda, splines
Date:	2007-7-31
Author:	Sundar Dorai-Raj
Maintainer:	Sundar Dorai-Raj <sundar.dora
License:	GPL (≥ 2)
CRAN checks:	DierckxSpline results

Downloads:

Package source:	DierckxSpline_1.0-9.tar.gz
MacOS X binary:	DierckxSpline_1.0-9.tgz
Windows binary:	DierckxSpline_1.0-9.zip
Reference manual:	DierckxSpline.pdf
Old sources:	DierckxSpline archive

"DierckxSpline" Package Contents



data

inst

man

R

src

DESCRIPTION

NAMESPACE

- data sets
- files not checked by 'R CMD check'
- Help files
- R function definition files
- source code in Fortran, C, C++, ...
- Package description
- Names to be exported
- Not all packages have all of these
- Some packages have others

- Ultimate documentation = source code
- "debug" function: walk through R code line by line until we understand what it does; "browser" for check points

PDF SOLUTIONS *Yield, Performance, Profitability*

Outline

- Installing R and R Packages
- Obtaining source code
- ➡ Creating R packages
  - Why?
  - How to create?
  - How to check?
  - How to share?
- Establishing and Maintaining Local R Archive Networks
- Using Subversion (SVN)

PDF SOLUTIONS
Yield, Performance, Profitability

Why Create R Packages

- **Productivity**
  - Tripled my software development productivity overnight
  - Help file with examples first; code to these examples
  - "R CMD check" finds when new changes break previous tests
  - Version control
- **Quality:**
  - Examples = unit testing
    - http://en.wikipedia.org/wiki/Unit_test
    - Chambers' "Prime Directive": Trustworthy software
      - (2008) *Software for Data Analysis* (Springer)
  - as well as documentation
- **Easy to share results**
- **Easy to understand what I did a couple of years ago**

How to Create an R Package

- **Copy existing package(s)**
- **“package.skeleton” function**
- **“Writing R Extensions” manual**
  - http://cran.r-project.org/doc/manuals/R-exts.pdf
- **Rossi, Peter (2006) Making R Packages under Windows**
  - (http://faculty.chicagogsb.edu/peter.rossi/research/bayes%20book/bayesm/Making%20R%20Packages%20Under%20Windows.pdf, accessed 2008.11.02)
- **R-devel listserve**
  - r-devel@stat.math.ethz.ch

Rolf Turner: In the middle of a Saturday morning (in my Time Zone!) I send out a plea for help, and in just over 20 minutes my problem is solved!
I don't think you get service like that anywhere else. This R-help list is BLOODY AMAZING!
Spencer Graves: 'The sun never sets on the (former) British Empire.' Today, it never sets on R-Help.
-- Rolf Turner and Spencer Graves
R-help (May 2005)

See section 1.1 in “Writing R Extensions”

Package Directory Structure

- packageName

  - **DESCRIPTION** – describes the package contents
  - **man** – Rd help files
  - **R** – R code files

  Required

  - NAMESPACE – defines the package name space
  - data – contains files with data (txt, csv, rda)
  - inst – contents are copied to installed package
  - src – C, Fortran code to compile with the package
  - tests – R code for testing package functions

  Optional

PDF/SOLUTIONS
Yield, Performance, Profitability

Building Packages On Windows

- **Requires Rtools**
  - Contains all compilers and Unix tools
  - http://www.murdoch-sutherland.com/Rtools
- **LaTeX: http://www.miktex.org**
- **For additional help, see:**
  - Google
  - r-devel mailing list
  - FAQ: http://cran.cnr.berkeley.edu/bin/windows/base/rw-FAQ.html
  - http://faculty.chicagogsb.edu/peter.rossi/research/bayes%20book/bayesm/Making%20R%20Packages%20Under%20Windows.pdf, accessed 2008.11.02

PDF SOLUTIONS *Yield, Performance, Profitability*

Building Packages On Mac

- **Mac tools are usually not loaded "out-of-the-box"**
  - Required tools maybe downloaded or installed from the OSX installation CDs
  - http://developer.apple.com/tools/xcode/
- **Latex: http://www.tug.org/mactex/**
- **Building packages on PPC and Intel Macs slightly different**
  - See the FAQ 5.4 on link below
- **Help**
  - http://cran.cnr.berkeley.edu/bin/macosx/RMacOSX-FAQ.html
  - R-SIG-Mac mailing list

Typical Package Check And Install Sequence

- **R CMD build packageName**
  - (or “R CMD build pkg” with an R-Forge package)
  - Windows: in a “Command Prompt” window with ‘packageName’ in the local directory
  - Creates “packageName_x.y-z.tar.gz”

current package version number

- **R CMD check packageName_x.y-z.tar.gz**
- **R CMD install packageName_x.y-z.tar.gz**
  - Installs it in your local installation of R
- **R CMD install --build packageName_x.y-z.tar.gz**
  - Creates “packageName_x.y-z.zip”, which can be used to install “packageName” on other Windows computers

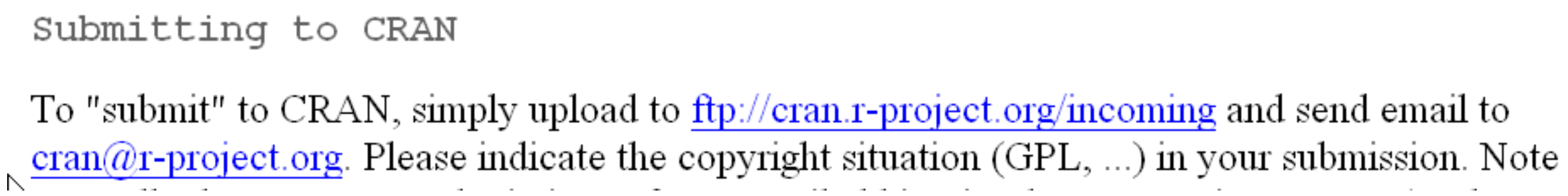
Cryptic Error Message?

- **invisible(lapply(list.files(“~packagepath/R", full = TRUE, pattern="\\.R$"), source))**
  - identifies functions and lines with syntax errors
- **Google**
- **RSiteSearch**
  - www.r-project.org → search
  - function in R
  - R-devel listserve
  - undo recent changes and try again from the last working version

PDF/SOLUTIONS
Yield, Performance, Profitability

Submitting A Package To CRAN

- **www.r-project.org -> CRAN -> (select a local mirror)**



Submitting to CRAN

To "submit" to CRAN, simply upload to ftp://cran.r-project.org/incoming and send email to cran@r-project.org. Please indicate the copyright situation (GPL, ...) in your submission. Note

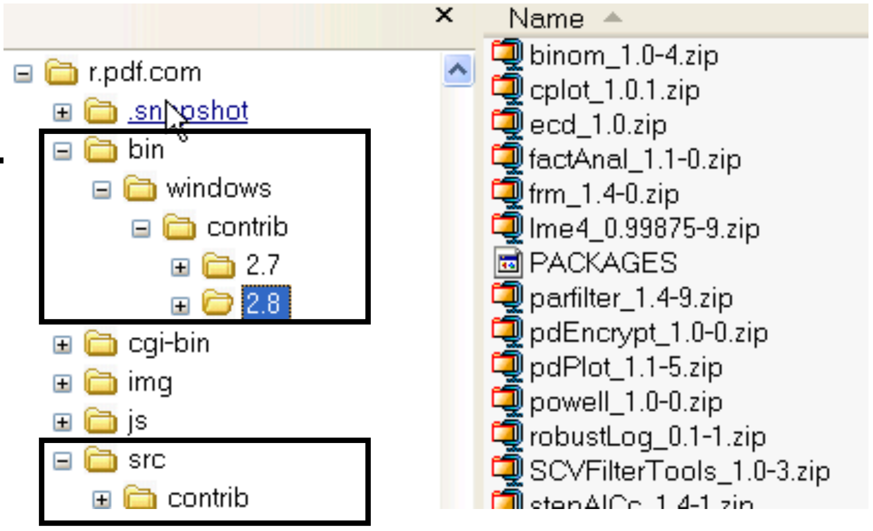
- **Build packageName_x.y-z with the current version of R**
- **Upload to ftp://cran.r-project.org/incoming**
  - packageName_x.y-z.tar.gz
  - (With firewall problems, can you use a different computer?)
- **Email "cran@r-project.org"**
  - subj: packageName_x.y-z.tar.gz now on CRAN
  - text: "uploaded to CRAN\incoming. GPL (>= 2)"

PDF SOLUTIONS
Yield, Performance, Profitability

Outline

- Installing R and R Packages
- Obtaining source code
- Creating R packages
- Establishing and Maintaining Local R Archive Networks
- Using Subversion (SVN)

Local R Archive Networks



```
r.pdf.com
  .snapshot
  bin
    windows
      contrib
        2.7
        2.8
  cgi-bin
  img
  js
  src
    contrib
```

```
Name
binom_1.0-4.zip
cplot_1.0.1.zip
ecd_1.0.zip
factAnal_1.1-0.zip
frm_1.4-0.zip
lme4_0.99875-9.zip
PACKAGES
parfilter_1.4-9.zip
pdEncrypt_1.0-0.zip
pdPlot_1.1-5.zip
powell_1.0-0.zip
robustLog_0.1-1.zip
SCVFilterTools_1.0-3.zip
stepAICc_1.4-1.zip
```

- **Why:**
  - Share work with others that you may not want to share with the world
- **How:**
  - Requires access to a web server
  - Then setting up a very specific directory structure to hold both source and binary packages
  - bin directory contains compiled packages for Windows (*.zip) or Mac (*.tgz)
    - Must contain a subdirectory for every supported version of R
  - src directory contains package source (*.tar.gz)

PDF SOLUTIONS
Yield, Performance, Profitability

Repository Directory Structure

- /www (directory that is visible from web)
  - bin
    - windows
      - contrib
        - 2.7 →
        - 2.8 →
    - macosx
      - contrib
        - 2.7 →
        - 2.8 →
  - src
    - contrib →

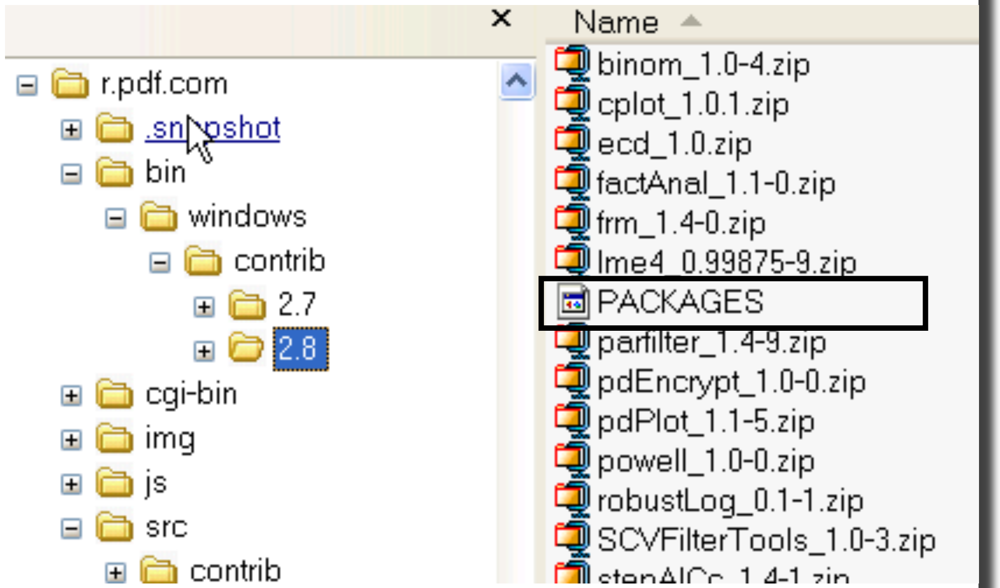
package1_x.y-z.zip
package2_x.y-z.zip
PACKAGES

package1_x.y-z.tgz
package2_x.y-z.tgz
PACKAGES

package1_x.y-z.tar.gz
package2_x.y-z.tar.gz
PACKAGES

Accessing The Repository Via install.packages

- **The PACKAGES file identifies which version to install**
  - Contents of PACKAGES equal DESCRIPTION file from each package



binom_1.0-4.zip
cplot_1.0.1.zip
ecd_1.0.zip
factAnal_1.1-0.zip
frm_1.4-0.zip
lme4_0.99875-9.zip
PACKAGES
parfilter_1.4-9.zip
pdEncrypt_1.0-0.zip
pdPlot_1.1-5.zip
powell_1.0-0.zip
robustLog_0.1-1.zip
SCVFilterTools_1.0-3.zip
stepAICc_1.4-1.zip

- **Installing a package**
  - install.packages(“packageName”, repos = “http://my.Rrepos.com”)
  - Or add to Rprofile.site (in $RHOME/etc)

```
options(repos = c(CRAN = "http://cran.cnr.berkeley.edu",
           myCRAN = "http://my.Rrepos.com",
           CRANextra = "http://www.stats.ox.ac.uk/pub/RWin"),
        max.print=222)
```

PDF SOLUTIONS *Yield, Performance, Profitability*

R.home() # R installation directory

Outline

- Installing R and R Packages
- Obtaining source code
- Creating R packages
- Establishing and Maintaining Local R Archive Networks
- Using Subversion (SVN)
    - Why?
    - Installing and Using Subversion
    - R-Forge
    - a local Subversion (SVN) repository
        - How to use
        - How to establish and maintain

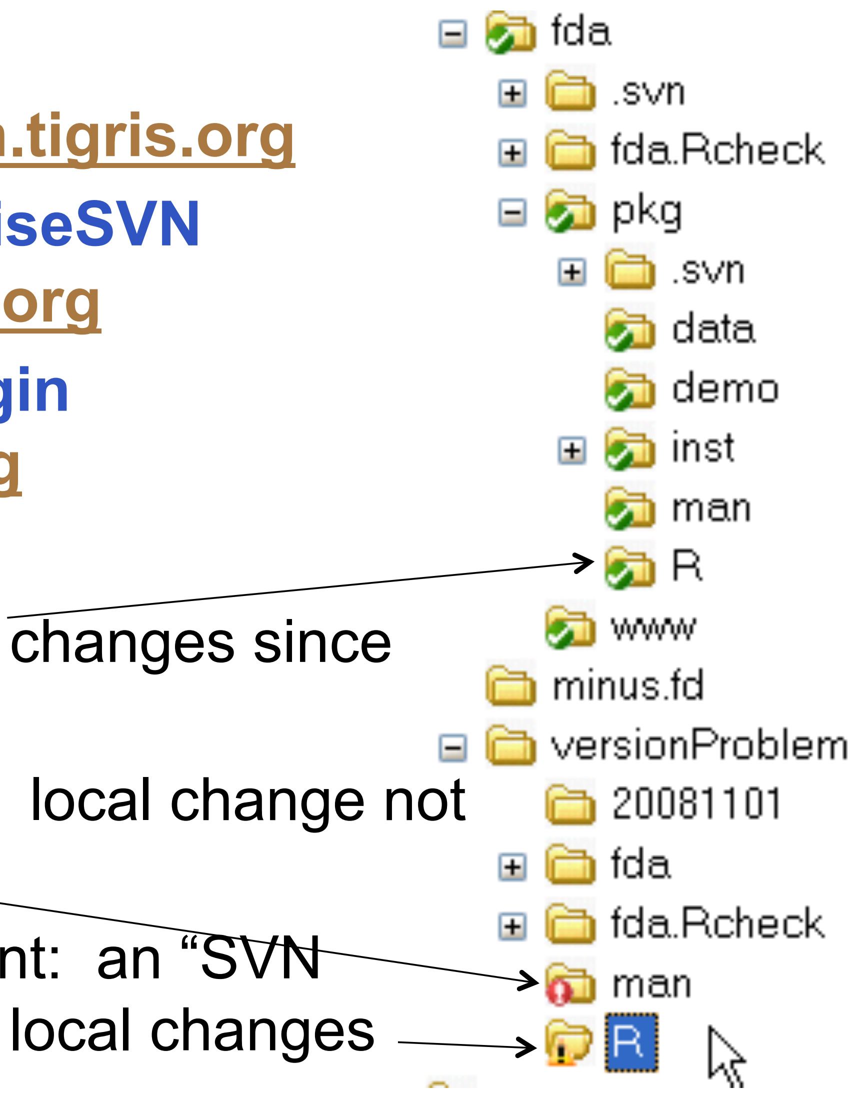
Why Use A Subversion Repository?

- **Easy to collaborate on package development**
- **Help learn R**
  - Find an R package that interests you
  - Make suggestions to the package maintainer
  - A maintainer may ask if you'd like do make those changes in their subversion repository
- **Audit trail on all changes**
  - Relatively easy to identify and reverse changes selectively

- **Creating an SVN repository (e.g. R-Forge) typically requires help from Information Technology**

Installing and Using Subversion (SVN) Client

- SVN – http://subversion.tigris.org
- Windows client – TortoiseSVN http://tortoisesvn.tigris.org
- Mac client – Finder plugin http://scplugin.tigris.org
- Symbols
  - Green check: No local changes since "Commit"
  - Red exclamation point: local change not in the repository
  - Yellow exclamation point: an "SVN Update" conflicted with local changes



PDF/SOLUTIONS
Yield, Performance, Profitability

SVN Checkout, Update, Commit

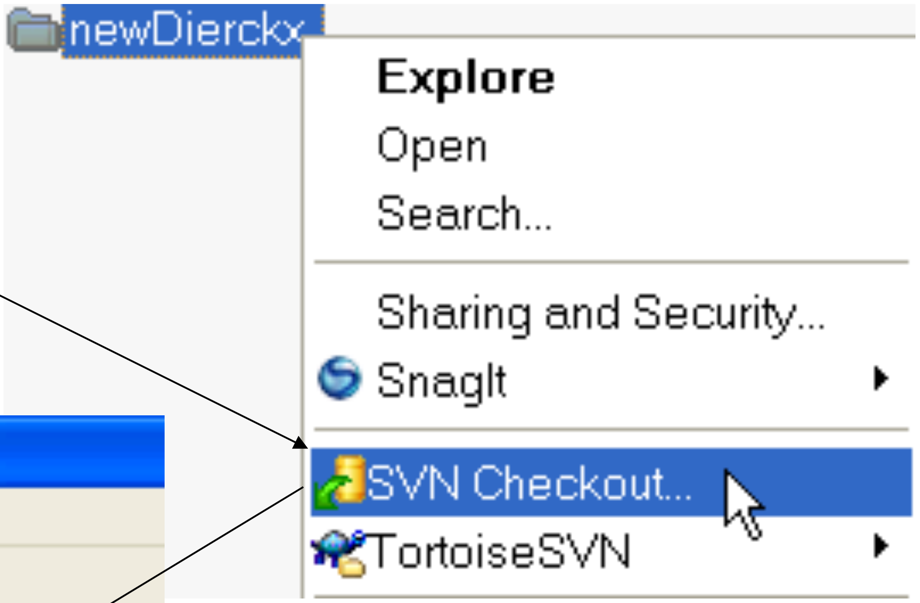
- **SVN Checkout**
  - Creates a local copy of a package on an SVN repository
- **SVN Update**
  - Updates local copies to newer versions on the repository
  - Identifies conflicts between recent changes made locally and elsewhere
- **SVN Commit**
  - Uploads recent changes from the local copy to the repository

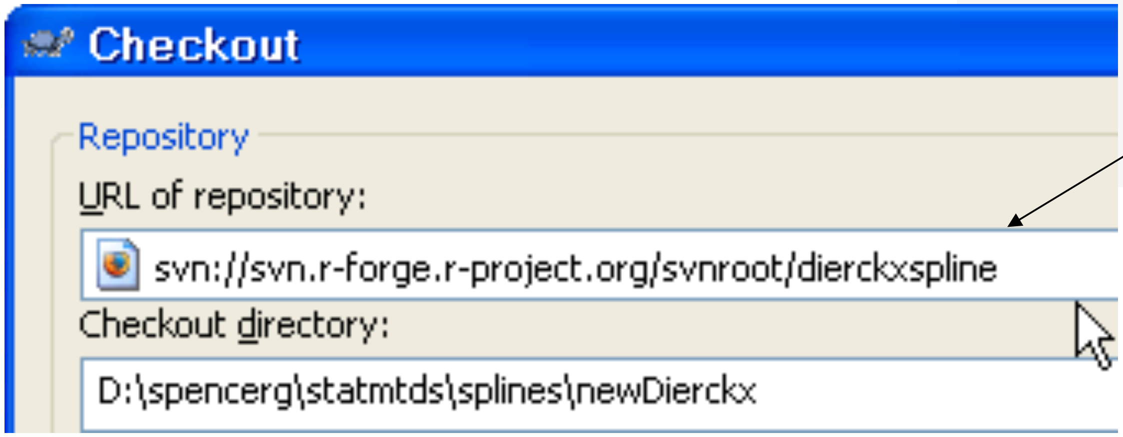
Two Subversion Repositories For R: RForge & R-Forge

- **RForge: www.rforge.net**
  - 37 projects as of 2009-03-11
- **R-Forge: r-forge.r-project.org**
  - 340 projects as of 2009-03-11
  - including DierckxSpline, FinTS, maxLik, fda, Rmetrics, ...
- **Both are free**
- **Installation of Packages in R: If an R-Forge package passed the quality check it can be installed directly via:**
  - install.packages(“DierckxSpline",repos="http://r-forge.r-project.org")

Anonymous Subversion Access From R-Forge

- svn checkout svn://svn.r-forge.r-project.org/svnroot/dierckxspline
- Windows: right-click on a new folder & select SVN Checkout





URL of repository:

svn://svn.r-forge.r-project.org/svnroot/dierckxspline

Checkout directory:

D:\spencerg\statmtds\splines\newDierckx

PDF/SOLUTIONS
Yield, Performance, Profitability

Developer Subversion Access Via SSH

- **Only project developers can access the SVN tree via this method. SSH must be installed on your client machine. Substitute *developername* with the proper values. Enter your site password when prompted.**
- **svn checkout svn+ssh://*developername*@svn.r-forge.r-project.org/svnroot/dierckxspline**

PDF/SOLUTIONS
Yield, Performance, Profitability

A Local Subversion Repository

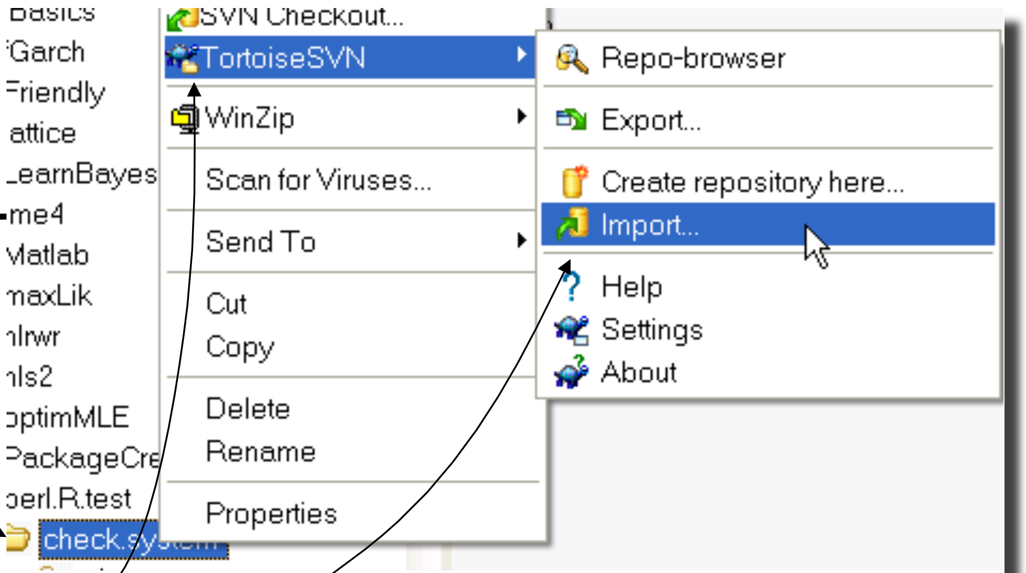
- **Why?**
  - Facilitate collaboration on software development
- **How?**
  - Different people typically work on different functions
  - “SVN Update” downloads recent changes made by others
  - “R CMD check” makes sure everything passes the programmed unit tests
  - “SVN Commit” uploads recent local changes

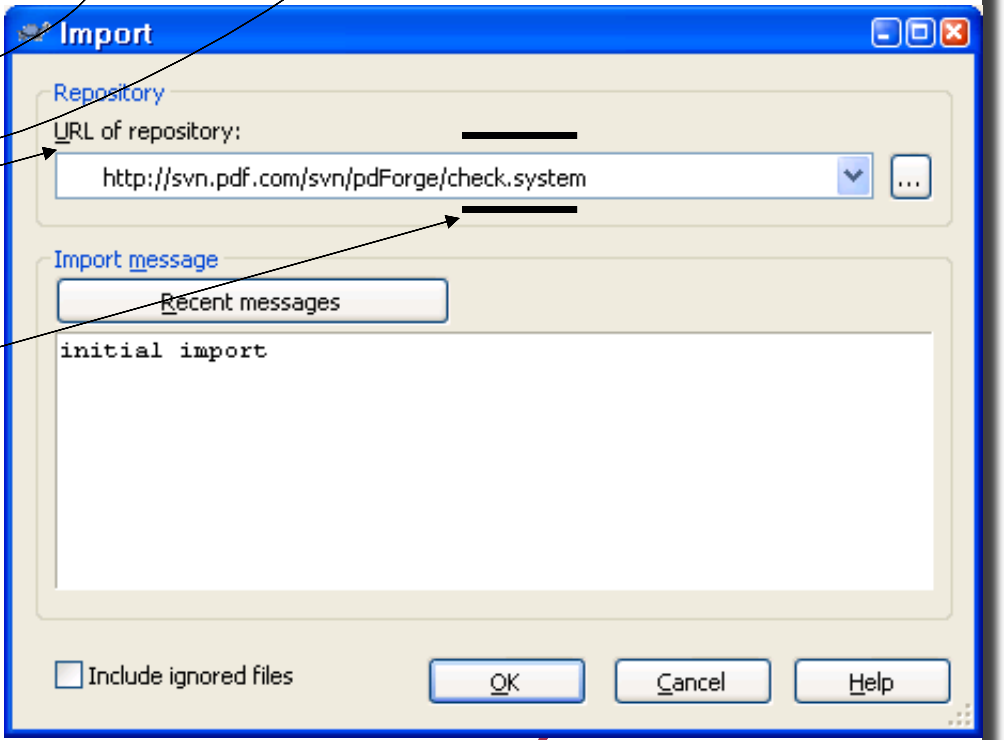
How To Establish/Maintain An SVN Repository

- Creating a repository server typically requires help from your local IT department
  - We won’t discuss that here.
- Once established, TortoiseSVN can be used to create projects.
- To add a new project to the repository:
  - “Import” to the repository
  - “Checkout” an official local copy
    - which contains the bookkeeping SVN requires that is NOT included in your “Import”

“Import” To The Repository

- Click on the folder containing the package (DESCRIPTION, MAN, R, ...)
- Tortoise SVN
- Import
- Enter “URL of repository”
  - with the name of your package





URL of repository: http://svn.pdf.com/svn/pdForge/check.system

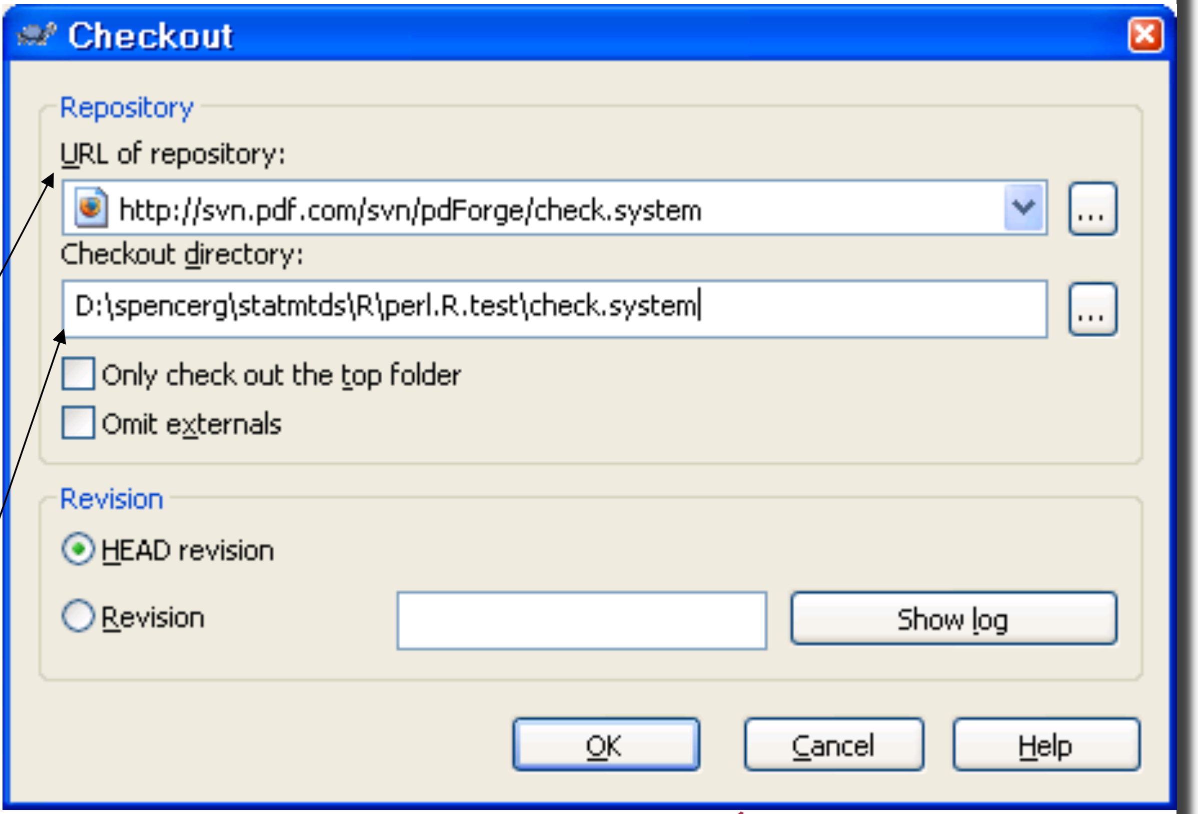
```
initial import
```

☐ Include ignored files

PDF SOLUTIONS
Yield, Performance, Profitability

“Checkout”

- Your original does NOT contain the bookkeeping information required by SVN
- Therefore, you need to “Checkout” an official copy properly configured for SVN
- To do that
  - Create a new folder to contain this version
  - Right-click: TortoiseSVN
  - Checkout
  - Enter “URL of Repository” and “Checkout Directory”



Checkout

Repository

URL of repository: http://svn.pdf.com/svn/pdForge/check.system

Checkout directory: D:\spencerg\statmtds\R\perl.R.test\check.system

[ ] Only check out the top folder

[ ] Omit externals

Revision

[✓ HEAD revision]

Revision

PDF SOLUTIONS Yield, Performance, Profitability

Outline

- Installing R and R Packages
- Obtaining source code
- Creating R packages
- Establishing and Maintaining Local R Archive Networks
- Using Subversion (SVN)

PDF/SOLUTIONS
Yield, Performance, Profitability

Annotated Bibliography

- **Writing R Extensions**
  - http://cran.r-project.org/doc/manuals/R-exts.pdf
  - THE official reference manual for R package development
  - BUT: It IS a reference manual, NOT a tutorial
- **Rossi, Peter (2006) Making R Packages under Windows: A Tutorial**
  - http://faculty.chicagogsb.edu/peter.rossi/research/bayes%20book/bayesm/Making%20R%20Packages%20Under%20Windows.pdf, accessed 2008.11.02
  - ***Excellent overview***

Annotated Bibliography - 2

- **Falcon, Seth (2006) Modeling package dependencies using graphs. *R News*, 6(5):8-12, December 2006.**
  - “pkgDepTools” package for viewing dependencies between packages
- **Gilbert, Paul, R (2004) package maintenance. *R News*, 4(2):21-24, September 2004.**
  - Reviews the “Make” capabilities described more fully in “Writing R Extensions”
- **Ligges, Uwe (2003) R help desk: Package management. *R News*, 3(3):37-39, December 2003.**
  - Managing packages in multiple ‘libraries’

PDF SOLUTIONS
Yield, Performance, Profitability

Annotated Bibliography - 3

- **Ripley, Brian D. (2005) Packages and their management in R 2.1.0. *R News*, 5(1):8-11, May 2005.**
  - Updates Ligges (2003) to R 2.1.0
- **Rougier, Jonathan (2005) Literate programming for creating and maintaining packages. *R News*, 5(1):35-39, May 2005.**
  - “The basic idea of *literate programming* is ... to keep the code and the documentatation ... together, in one file” using the “noweb” literate programming tool.