R FAQ Windows

KRUG

http://cran.r-project.org/bin/windows/rw-FAQ.html

목차

Introduction

This FAQ is for the Windows port of R: it describes features specific to that version. The main R FAQ can be found at http://CRAN.R-project.org/doc/FAQ/R-FAQ.html. The information here applies only to recent versions of R for Windows, ('2.14.0' or later). It is biased towards users of 64-bit Windows. 이 FAQ는 R의 Windows 포트를 위한 것 입니다. 이것은 버전에 대한 특징들을 설명합니다. 중요한 R FAQ는 http://CRAN.R-project.org/doc/FAQ/R-FAQ.html에서 찾을수 있습니다. 여기에 있는 정보는 Windows에 대한 최신버전(2.14.0 이상)의 R만을 지원합니다. 이것은 64bit Windows 유저들에게 편향되어집니다.

Installation and Usage

Where can I find the latest version?

Go to any CRAN site (see http://cran.r-project.org/mirrors.html for a list), navigate to the bin/windows/base directory and collect the file(s) you need. The current release is distributed as an installer `R-2.15.0-win.exe' of about 47Mb. This contains all the components and allows as complete as installation as you choose.

아무 CRAN 사이트로 가십시오(http://cran.r-progect.org/mirrors.html 을 보십시오

There are also links on that page to the `r-patched' and `r-devel' snapshots. These are frequently updated builds of development versions of R. The `r-patched' build includes bug fixes to the current release, and `r-devel' contains these as well as changes that will eventually make it into the next `x.y.0' release.

How do I install R for Windows?

Current binary versions of R run on Windows XP or later, including on 64-bit versions: See Can I use R on 64-bit Windows?. The last version known to run on Windows 2000 was 2.12.2 and the last version supporting 95/98/ME/NT4 was 2.6.1.

현재 R binary 버전들은 윈도우 xp 버전(64비트 버전 포함) 이후에 사용할 수 있습니다. "Can I use R on 64-bit Windows?"를 참고 하십시오. 윈도우즈 2000에 대한 최신 버전은 2.12.2 였고, 95/98/ME/NT4버전은 2.6.1 버전이 지원합니다.

We only test on versions of Windows currently supported by Microsoft, mainly 64-bit Windows 7 and Server 2008, but to a limited extent on 32-bit XP SP3.

저희는 현재 Microsoft가 지원하는 현재 윈도우 버전(주요 64비트 윈도우즈7) 에서 테스트하고 있지만, 제한적으로 32비트 XP SP3도 확장하고 있습니다.

Your file system must allow case-honouring long file names (as is likely except perhaps for some network-mounted systems). A installation for one architecture takes about 75Mb of disk space, for both about 90Mb.

사용자의 파일 시스템은 case-honouring long file names(몇몇의 네트워드가 설치된 시스템은 어쩌면 제외되기 때문에) 지원해야만 합니다. A installation for one architecture 은 75Mb의 디스크 공간을 차지고, 두개의 architecture는 95Mb를 차지합니다.

If you want to be able to build packages from sources, we recommend that you choose an installation path not containing spaces. (Using a path with spaces in will probably work, but is little-tested.) Users of Vista/Windows 7/Server 2008 installing for a single user using an account with administrator rights1 should consider installing into a non-system area (such as C:₩R). Installing to a network share (a filepath starting with ₩₩machine₩...) is not supported: such paths will need to mapped to a network drive.

만약 사용자가 소스들로부터 패키지 설치하기를 원한다면, 우리는 사용자가 an installation path를 포함하지않는 공간 선택을 추천합니다. (하나의 공간안의 path를 사용하는, 대체적으로 잘 작동하지만, 많은 테스트를 거치지 않았습니다) Vista/Windows 7/Server 2008 를 설치한 유져(administrator rights1과 함께 하나의 계정만을 사용) 들은 non-system area에 설치하는 것을 심사숙고해야만 합니다(C:/R와 같이).

network share 설치는 제공되지 않습니다(//machin/....시작하는 파일경로) such paths will need to mapped to a network drive.

To install use `R-2.15.0-win.exe'. Just double-click on the icon and follow the instructions. If you have an account with Administrator privileges you will be able to install R in the Program Files area and to set all the optional registry entries; otherwise you will only be able to install R in your own file area. You may need to confirm that you want to proceed with installing a program from an `unknown' or `unidentified' publisher.

설치하기 위해 'R-2.15.0-win.exe'를 사용하세요. 그 exe 파일을 더블클릭하고, 설명에 따라 진행하세요. 만약 사용자가 Administrator preivileges 계정을 가지고 있다면, 프로그램 파일안에 R을 설치할수 있고, 모든 optional registry entries들을 설정할 수 있습니다. 그게 아니라면, 사용자는 오직 자신만의 파일 공간에 R을 설치 할 수 있습니다. 사용자는 어쩌면 'unkbown' 혹은 'unidentified' publisher로 부터의 프로그램을 설치하는 과정을 확인할 필요가 있습니다.

After installation you should choose a working directory for R. You will have a shortcut to Rgui.exe on your desktop and/or somewhere on the Start menu file tree, and perhaps also in the Quick Launch part of the taskbar (Vista and earlier). Right-click each shortcut, select Properties... and change the `Start in' field to your working directory.

설치이후에, 사용자는 R의 working directory를 선택해야만 합니다. Rgui.exe(바로가기)가 바탕화면, 스타트 메뉴 파일 트리, 혹은 taskbar의 퀵 런처파트(vista 와 이전 버전들)에 생성될 것 입니다. 각각의 바로가기를 마우스 우클릭하고, properties선택, 그리고 'start in' 를 사용자의 working directory로 변경하십시오

On some systems you will have two shortcuts, one for 32-bit with a label starting R and one for 64-bit starting R x64 (see Should I run 32-bit or 64-bit R?) 몇몇 시스템을 가진 사용자들은 두개의 바로가기를 갖게 될 것입니다. 하나는 starting R 라벨을 가진 32bit를 , 다른 하나는 64bit를 위한 것 입니다. ('Should I run 32-bit or 64-bit R?' 를 참고하십시오)

You may also want to add command-line arguments at the end of the Target field (after any final double quote, and separated by a space), for example --sdi --max-mem-size=1G. You can also set environment variables at the end of the Target field, for example R_LIBS=p:/myRlib, and if you want to ensure that menus and messages are in (American) English, LANGUAGE=en.

사용자는 command line arguments를 Target Field의 마지막에 추가하고 싶을지도 모릅니다(any final double quote와 공간에 의한 구분 이후에) 예를드면 다음과 같습니다 "--sdi --max-mem-size=1G". 사용자는 또한 Target field의 마직막에 변수 환경을 설정할수 있습니다. 예를 들면 'for example R_LIBS=p:/myRlib' 입니다. 그리고 만약 사용자가 menus와 messages를 영문으로 등록하고 싶으면, LANGUAGE=en.

It is also possible to install from an MSI file, which will be of interest only for system administrators. For how to build the MSI file, see the `R Installation and Administration Manual'.

MSI 파일(시스템 관리자들에게는 흥미로울 것입니다)을 이용하여 설치하는 것 또한 가능합니다. 어떻게 MSI 파일을 만드는지에 대해서는 'R Installation and Administration Manual'를 참고 하십시오.

How do I check an installation is not corrupt?

Relates to earlier installers, removed in R 2.11.0.

R 2.11.0에서 제거된 설치파일들과 관련되어 있습니다.

Can I customize the installation?

The normal way to customize the installation is by selecting components from the wizards shown by the installer. However, sysadmins might like to install R from scripts, and the following command-line flags are available for use with the installer.

사용자가 원하는 대로 설치하는 일반적인 방법은 installer 마법사로 부터 콤포넨트를 선택하는 것 입니다. 하지만 sysadmin들은 스크립트를 이용하여 R 설치하는 것을 좋아하고, installer의 사용은 following command-line flags를 이용할 수 있습니다.

`/SILENT' only show the installation progress window and error messages.

오직 설치 과정 윈도우와 에러 메세지만을 보여줍니다.

`/VERYSILENT' only show error messages.

오직 에러 메세지만 보여줍니다. `/DIR="x:₩dirname"' set the default installation directory

기본 설치 directory를 설정합니다

`/GROUP="folder name"' set the default Start-menu group name

기본 Start-menu group name을 설정합니다.

`/COMPONENTS="comma separated list of component names"' set the initial list of components: Components are named `main', `i386' and `x64' (there were many more before R 2.15.0). It is also possible to save the settings used to a file and later reload those settings using

콤포넨트의 이니셜 리스트를 설정합니다. 콤포넨트들의 이름은 `main', `i386' , `x64' 입니다. R 2.15.0 버전 이전에는 더욱 많은 이름들이 있었습니다. 파일에 사용되었던 설정들은 저장하는 것과 다시 불러오는 것 또한 가능합니다.

`/SAVEINF="filename"' save the settings to the specified file. Don't forget to use the quotes if the filename contains spaces.

특정한 파일에 대한 설정을 저장합니다. 만약 파일이름에 공백이 있다면, quotes를 사용해 주시기 바랍니다.

`/LOADINF="filename"' instructs the installer to load the settings from the specified file after having checked the command line. A successful installation has exit code 0: unsuccessful ones may give 1, 2, 3, 4 or 5. See the help for Inno Setup (http://jrsoftware.org/ishelp.php) for details.

command line을 확인한 이후,installer에게 특정한 파일로부터 설정들을 불러오도록 지시합니다.

We have some facilities for building a customized installer, in particular to add packages to the installer. See the `R Installation and Administration' manual in the subsection `Building the installers'.

저희는 사용자지정 installer 설계를 위한 설비들을 가지고 있습니다 (특별히 installer에 패키지 더하는 것). 'Building the installers'안의 'R Installation and Administration' 메뉴얼을 보십시오.

How do I run it?

Just double-click on the shortcut you prepared at installation. If you want to set up another project, make a new shortcut or use the existing one and change the `Start in' field of the Properties.

사용자가 준비했었던 installation의 바로가기를 더블클릭하십시오. 만약 사용자가 다른 프로젝트를 셋업하기 원한다면, 새로운 바로가기를 만들거나, 혹은 이미 만들어진 것을 사용하시고, 'Start in'(field of the Properties) 를 변경하십시오.

You may if you prefer run R from the command line of any shell you use, for example a `Command Prompt' or a port of a Unix shell such as tcsh or bash. (The command line can be anything you would put in the Target field of a shortcut, and the starting directory will be the current working directory of the shell.)

만약 사용자가 사용하는 shell의 command line으로 부터 R을 실행하기를 선호한다면, 사용하십시오. 예를 들면 'Command Prompt'혹은 tcsh, bash와 같은 Unix shell의 포트를 사용하시면 됩니다.(사용자는 cammand line에 자신이 원하는 어떠한 것이든 바로가기의 Target field안에 놓을 수 있습니다. directory를 시작하는 것은 현재 shell의 working dircetory 가 될 것 입니다)

Can I run R from a CD or USB drive?

Yes, with care. A basic R installation is relocatable, so you can burn an image of the R installation on your hard disc or install directly onto a removable storage device such as a flash-memory USB drive.

가능합니다. 기본적인 R installation 은 재배치 되어 질 수 있으며, 그로인해 사용자는 R installation 이미지를 하드디스크 혹은 외부 저장 기기(USB 메모리 등)에 구울 수 있습니다

Running R does need access to a writable temporary directory and to a home directory, and in the last resort these are taken to be the current directory. This should be no problem on a properly configured version of Windows, but otherwise does mean that it may not be possible to run R without creating a shortcut starting in a writable folder.

R 을 실행시키는 것은 쓰기가 가능한 일시적인 directory와 home directory에 대한 엑세스를 필요로 하고, and in the last resort these are taken to be the current directory. 적절하게 환경설정된 윈도우즈의 버전을 문제가 없어야만 하지만, 만약 아니라면, 쓰기가능한 폴더안에 바로가기 생성하는것 없이는 R을 실행하는 것을 아마 불가능 할 것입니다.

How do I UNinstall R?

Normally you can do this from the `Add/Remove Programs' (XP) or `Programs and Features' (Vista/7) group in the Control Panel. If it does not appear there, run unins000.exe in the top-level installation directory. On recent versions of Windows you may well be asked to confirm that you wish to run a program from an `unknown' or `unidentified' publisher.

일반적으로 사용자는 제어판 안의 'Add/Remove Programs'(XP)혹은 'Programs and Features'(Vista/7)에서 이것을 할 수 있습니다. 만약 이것이 제어판에 나타나지 않는다면 unis000.exe를 top-level installation

directory 안에서 실행하십시오. 최신의 윈도우즈 버전들 위에서, 사용자는 'unknown' 혹은 'unidentified' publisher 로부터 그 프로그램은 실행 할 것인지에 대해 질문 받을것입니다

Uninstalling R only removes files from the initial installation, not (for example) packages you have installed or updated. If all else fails, you can just delete the whole directory in which R was installed.
Uninstalling R은 오직 첫번째로 설치된 파일들만 제거합니다. 이것은 사용자가 지금까지 설치하고 업데이트한 패키지들은 삭제하지 않는다는 말과 같습니다. 만약 모든 다른것들이 실패한다면, 사용자는 R이 설치된 whole directory를 삭제하십시오

What's the best way to upgrade?

That's a matter of taste. For most people the best thing to do is to uninstall R (see the previous Q), install the new version, copy any installed packages to the library folder in the new installation, run update.packages(checkBuilt=TRUE, ask=FALSE) in the new R and then delete anything left of the old installation. Different versions of R are quite deliberately installed in parallel folders so you can keep old versions around if you wish.

이것은 사람의 기호에 따라 다릅니다. 대부분의 사람들에게 최고의 방법은 R을 삭제한 후, 새로운 버전을 설치하고, 새롭게 설치된 library 폴더로 설치된 패키지들을 복사하는 것 입니다. 새로운 R의 update.packages(checkBuilt=TRUE, ask=FALSE)를 실행한 다음 이전에 설치되었던 것들을 삭제하십시오. 다양한 R 버전들은 계획적으로 평행적인 폴더에 설치되어질수 있고, 이로 인해 사용자는 오래된 버전들을 유지할 수 있습니다.

For those with a personal library (folder R₩win-library₩x.y of your home directory, R₩win64-library₩x.y on 64-bit builds), you will need to update that too when the minor version of R changes (e.g. from 2.14.2 to 2.15.0). A simple way to do so is to copy (say) R₩win-library₩2.14 to R₩win-library₩2.15 before running update.packages(checkBuilt=TRUE, ask=FALSE).

There seems to be a limit on the memory it uses!

Indeed there is. It is set by the command-line flag --max-mem-size (see How do I install R for Windows?). For a 64-bit build of R it defaults to the amount of RAM.

command-line flag --max-mem-size 에 의해서 그 것은 설정되었습니다(How do I install R for Windows?를 보십시오). 64 비트 R 에 대해서, 그 것은 RAM 용량 만큼은 기본값으로 합니다

For a 32-bit build of R it defaults to the smaller of the amount of physical RAM in the machine and 0.5GB less than the limit on user virtual memory for a process (most often 2GB when using a 32-bit edition of Windows).

32비트 R에 대해서는, 실제 RAM의 양보다 적은 양이 기본값으로 설정되고, 프로세스에 대한 유저 가상메모리의 제한보다 0.5 GB 적습니다(32비트 윈도우즈는 대부분 2GB를 사용합니다)

Use ?Memory and ?memory.size for information about memory usage. The limit can be raised (if

possible) by calling memory.limit within a running R session.

Use ?Memory and ?memory.size for information about memory usage. 만약 가능하다면, 사용자는 실행되고있는 R session안의 memory.limit를 불러옴으로써 용량제한을 증가시킬수 있습니다

The 32-bit executables support up to 3GB of user address space per process under suitably enabled versions of 32-bit Windows (see http://www.microsoft.com/whdc/system/platform/server/PAE/PAEmem.mspx, http://msdn.microsoft.com/en-us/library/bb613473%28VS.85%29.aspx; this is not enabled by default). On such systems, the default for --max-mem-size is the smaller of the amount of RAM and 2.5GB. On all but the earliest 64-bit versions of Windows the user address space for a 32-bit process is 4GB, and there the default for --max-mem-size is the smaller of the amount of RAM and 3.5GB.

32비트로 실행가능한 것들은 최고 32비트 윈도우버전들이 안정적으로 이용되어지는 프로세스 마다 user address space 의 3GB까지 지원합니다.(see http://www.microsoft.com/whdc/system/platform/server/PAE/PAEmem.mspx, http://msdn.microsoft.com/en-us/library/bb613473%28VS.85%29.aspx; this is not enabled by default). --max-mem-size에 대한 기본설정들은 RAM 과 2.5GB 용량보다 작습니다.초기 64비트 윈도우 버전을 제외한 모든 버전들의 32비트 프로세스에 대한 user address space는 4GB 이고, --max-mem-size에 대한 기본설정을 RAM과 3.5GB의 양보타 적습니다.

How can I keep workspaces for different projects in different directories?

Create a separate shortcut for each project: see Q2.5. All the paths to files used by R are relative to the starting directory, so setting the `Start in' field automatically helps separate projects.

각각의 프로젝트에 대한 분활된 바로가기을 생성하십시오: Q2.5를 보십시오. 모든 R에 의해 이용되어지는 파일 경로들은 starting directory와 연관됩니다. 그러므로, 'Start in' 필드를 세팅하는 것은 자동적을 separate progect 들을 도와줍니다.

Alternatively, start R by double-clicking on a saved .RData file in the directory for the project you want to use, or drag-and-drop a file with extension .RData onto an R shortcut. In either case, the working directory will be set to that containing the file.

다른 방법으로는, 사용자가 사용하기 원하는 프로젝트에 대한 directory안의 Rdata file 클릭함으로써 R을 시작하거나, extension .RData 파일을 R 바로가기 위에 드래그 앤 드랍함으로써 R을 시작하십시오

How do I print from R?

It depends what you want to print.

이것은 사용자가 어떤 것을 프린터하느냐에 의해서 달라집니다

You can print the graphics window from its menu or by using dev.print with suitable arguments (see its help page: most likely dev.print(win.graph) will work). You can print from the R console or pager by `File | Print'. (This will print the selection if there is one, otherwise the whole console or pager contents.) You can print help files from the pager or HTML browser. If you have LaTeX installed and a PostScript or PDF printing system you can print help files by help(fn_name, help_type="postscript")

(or "PDF").

사용자는 메뉴의 그래픽 윈도우를 통하거나 dev.print(with suitable arguments) 를 사용함으로써 프린트 할수있습니다(도움말 페이지를 보십시오: most likely dev.print(win.graph) will work). 사용자는 R 콘솔 혹은 `File | Print'에 의한 pager로 부터 프린트 할수있습니다.(만약 하나만 있다면,이것은 selection 을 프린트할 것입니다. 만약 아니라면, 전체 콘솔 혹은 pager contents 프린트 합니다). 사용자는 pager 나 HTML 브라우저 로 부터 help 파일들을 프리트 할 수 있습니다. 만약 사용자가 설치된 LaTeX와 PostScript 혹은 PDF 프린팅 시스템를 가지고 있다면, help(fn_name, help_type="postscript") (or "PDF")에 의해서 도움 파일은 프린트 할수 있습니다.

Can I use R CMD BATCH?

Yes: use R CMD BATCH --help or ?BATCH for full details. You can also set up a batch file using Rterm.exe. A sample batch file might contain (as one line) 할수있습니다. 모든 세부사항을 보려면R CMD BATCH --help or ?BATCH 를 사용하십시오. 사용자는 Rterm.exe를 사용하면 batch 파일을 설정할 수있습니다. batch 파일 샘플은 포함할지도 모릅니다.

```
path_to_R\bin\x64\Rterm.exe --no-restore --no-save < %1 > %1.out 2>&1
```

The purpose of 2>&1 is to redirect warnings and errors to the same file as normal output.

Can I use R-2.15.0 with ESS and (X)emacs?

Yes. ESS has for a long time supported R under Windows: it does so by running Rterm.exe without a visible console. For help with ESS, please send email to ESS-help@stat.ethz.ch, not the R mailing lists.

가능합니다. ESS는 윈도우 운영체제에 대한 R LTS(long time supported R)을 가지고 있습니다. 이것은 콘솔없이 Rterm.exe를 통해서 실행됩니다. ESS에 대한 도움말을 위해서 R mailing lists가 아닌 ESS-help@stat.ethz.ch 로 이메일 하시기 바랍니다.

What are HOME and working directories?

Several places in the documentation use these terms.

documentation안의 명명 장소들은 이것들과 같은 용어(HOME, working directories)들을 사용합니다.

The working directory is the directory from which Rgui or Rterm was launched, unless a shortcut was used when it is given by the `Start in' field of the shortcut's properties. You can find this from R code by the call getwd().

working directory는 Rgui 혹은 Rterm이 런치된 directory 입니다(바로가기 설정의 'Start in' 필드에 의해서 그것이 주어졌을 때, 바로가기를 사용하지 않는다면). 사용자는 getwd()코드를 사용함으로써 이것을 찾아낼 수 있습니다.

The home directory is set as follows: If environment variable R_USER is set, its value is used. Otherwise if environment variable HOME is set, its value is used. After those two user-controllable settings, R tries to find system-defined home directories. It first tries to use the Windows "personal" directory (typically C:₩Documents and Settings₩username₩My Documents on Windows XP and C:₩Users₩username₩Documents on Vista/Windows 7). If that fails, if both environment variables HOMEDRIVE and HOMEPATH are set (and they normally are), the value is ${HOMEDRIVE}${HOMEPATH}. If all of these fail, the current working directory is used.

home directory는 다음과 같이 설정되어 있습니다 : 만약, environment variable R_USER가 설정되어 있다면, 그 것의 값들은 사용되어집니다. environment variable HOME으로 설정되어 있지만 않다면, 그 것의 값은 사용되어집니다. 이러한 두 user-controllable 설정을 한 이후, R은 home directories들로 정의된 시스템 찾기를 시작합니다. 그 것을 먼저 윈도우즈 "personal" directory(일반적으로 윈도우즈 XP의 directory는 C:₩Documents and Settings₩username₩My Documents 이고, 윈도우즈 vista와 7의 diretory는 C:₩Users₩username₩Documents 입니다) 사용을 시도합니다. 만약 이것이 실패했을때, 두 environment variables HOMEDRIVE, HOMEPATH가 설정되어있기만 한다면, 그 값은 ${HOMEDRIVE}${HOMEPATH}입니다. 위의 모든것들이 실패했다면, 현제 working directory가 사용됩니다.

You can find this from R code by Sys.getenv("R_USER") or normalizePath("~"), `~' being Unix notation for the home directory.

사용자는 R 코드 Sys.getenv("R_USER) 혹은 normalizePath("~")를 이용하여 이것을 찾을수 있습니다. home directory에 대하여 '~'는 Unix notation 입니다.

How do I set environment variables?

Environment variables can be set for Rgui.exe and Rterm.exe in three different ways. Rgui와 Rterm에 대한 환경변수는 세가지 다른 방법들을 통해 설정될 수 있습니다 On the command line as name=value pairs. For example in the shortcut to Rgui you could have

```
"path_to_R\bin\x64\Rgui.exe" HOME=p:/ R_LIBS=p:/myRlib
```

In an environment file .Renviron in the working directory or in your home directory, for example containing the line

```
R_LIBS=p:/myRlib
```

If you have permission to do so, you can also create an environment file etc₩Renviron.site and set environmental variables in that file in the same way. This is useful for variables which should be set for all users and all usages of this R installation. (Their values can be overridden in a .Renviron file or on the command line.)

만약 사용자가 이것을 하기 위한 permission을 가지고 있다면, environment file etc/Renviron.site를 생성

할 수 있고, 그 파일 안의 환경변수들도 같은 방법으로 설정할 수 있다. 이 것은 모든 유저에 대한 설정과 R installation를 위한 변수들에 대해 유용합니다. 그것들에 값들은 .Renviron파일 혹은 command line에서 중단 될 수 있다.

See ?Startup for more details of environment files.

environment 파일들에 대한 세부 정보들은 ?Startup에서 볼수 있습니다.

For all applications via Windows. How you set an environment variable is system specific: under Windows XP/Server 2003 you can use `System' in the control panel or the properties of `My Computer' (under the `Advanced' tab). Under Vista/7/Server 2008, go to `User Accounts' in the Control Panel, and select your account and then `Change my environment variables'. The order of precedence for environmental variables is the order in which these options are listed, that is the command line then .Renviron then the inherited environment.

For all applications via Windows. How you set an environment variable is system specific: 윈도우즈 XP/server 2003 에서의 사용자는 제어판 혹은 내컴퓨터 설정에서 System을 사용할 수 있습니다. Vista/7 /Server 2008에서는 제어판 안의 'User Accounts'로 가서, 사용자의 계정을 선택한 다음, change my environment variables 를 선택하십시오. 환경 변수에 대한 우선 순위는 옵션(이것은 command line입니다)들이 나열되어 있는 순서입니다. comman line 다음에 .Renviron, 그 다음 inherited environment 등의 순서입니다.

R can't find my file, but I know it is there!

How did you specify it? Backslashes have to be doubled in R character strings, so for example one needs `"d:\\R-2.15.0\\library\\xgobi\\scripts\\xgobi.bat"'. You can make life easier for yourself by using forward slashes as path separators: they do work under Windows. You should include the file extension (e.g. `"xgobi.bat"' rather than just `"xgobi"'); sometimes this isn't shown in Windows Explorer, but it is necessary in R.

사용자는 어떻게 그것을 구체적으로 명시하나요? 두개의 Backslashe가 R character strings에 있어야합니다. 그래서 예를 들면 하나는 `"d:\\R-2.15.0\\library\\xgobi\\scripts\\xgobi.bat"'을 필요로 합니다. 사용자는 앞의 슬래쉬들을 path separators로 사용함으로써 보다 쉽게 작업할 수 있습니다. 그들을 윈도우 운영체제에서 작동됩니다. 사용자는 file extension(예를 들면, 그냥 "xgobi" 보다는 "xgobi.bat"이 좋습니다)을 포함해야만 합니다. 어쩌다가 이것은 윈도우즈 익스플로러에서 볼수 없을수도 있지만, R은 이것을 필요로 합니다.

A simple way to avoid these problems is to use the function file.choose() to invoke the standard Windows file selection dialog. If you select a file there, the name will be passed to R in the correct format.

이러한 문제점들을 피하기 위한 간단한 방법으로는 standard Windows file selection dialog 를 실행하기 위한 function, file.choose()를 사용하십시오. 만약 사용자가 저곳에 있는 파일 하나를 선택하였다면, 그 이

름은 R이 요구하는 정확한 포맷이 될것입니다.

Another possible source of grief is spaces in folder names. We have tried to make R work on paths with spaces in, but many people writing packages for Unix do not bother. So it is worth trying the alternative short name (something like `PROGRA~1'; you can get it as the `MS-DOS name' from the Properties of the file on some versions of Windows, and from dir /X in a `Command Prompt' window), and using the function shortPathName from R code.

또다른 문제점으로는 폴더이름의 공간입니다. 우리는 R이 space를 가진 paths에서 실행하는 것을 시도해 오고 있지만, Unix의 패키지를 작성하는 많은 사람들을 우리의 노력을 신경쓰지 않습니다. 그러므로, 대체 가능한 짧은 이름과 R 코드인 function shortPathName을 사용하는 것은 의미있습니다. 짧은 이름의 예로는 'PROGRA~1' 있습니다. 사용자는 이것을 'MS-DOS name'으로써 몇몇 버전의 윈도우즈에 있는 파일의 Properties와 'Command Prompt' 창의 dir /X으로부터 얻을수 있습니다.

Does R use the Registry?

Not when R itself is running.

R 자신혼자 실행되고 있을때는 Registry를 사용하진 않습니다.

When you run the R installer, there are options (under `Select Additional Tasks') to `Save version number in registry' and (for Administrator installs) `Associate R with .RData files'.

사용자가 R installer를 실행 시킬때, Select Additional Tasks아래 'Save version number in registry'와 'Associate R with .RData files' 있습니다.

If you tick the first option, the following string entries are added to the Windows registry:

만약 사용자가 첫번째 옵션을 선택한다면, 이후의 string entries들을 Windows registry 에 더해집니다.

HKEY_LOCAL_MACHINE₩Software₩R-core₩R₩Current Version contains the version number, currently 2.15.0. HKEY_LOCAL_MACHINE₩Software₩R-core₩R₩[version]₩InstallPath (where [version] is currently 2.15.0) contains the path to the R home directory. If you do not have administrative privileges on the machine while running the installer, then the entries are created under HKEY_CURRENT_USER. The same entries are also created under Software₩R-core₩R32 or Software₩R-core₩R64, for 32- and 64-bit R respectively. If you tick the second option (shown with administrative privileges only) (`Associate R with .RData files') then entries are created under HKEY_CLASSES_ROOT₩.RData and HKEY_CLASSES_ROOT₩RWorkspace.

HKEY_LOCAL_MACHINE₩Software₩R-core₩R₩Current Version은 현재버전의 넘버(2.15.0)을 포함하고 있습니다. HKEY_LOCAL_MACHINE₩Software₩R-core₩R₩[version]₩InstallPath (where [version] is currently 2.15.0)은 R home directory의 경로를 포함합니다. 만약, 사용자가 installer를 실행하고 있는 동안 관리자 권한을 가지고 있지 않다면, entries는 HKEY_CURRENT_USER에 생성됨

니다. 동일한 entries는 32 와 64 비트 R 각각 Software₩R-core₩R32 혹은 Software₩R-core₩R64에 생성됩니다. 만약 두번째 옵션을 (shown with administrative privileges only) ('Associate R with .RData files')을 체크표시 했다면, entries들은 HKEY_CLASSES_ROOT₩.RData dhk HKEY_CLASSES_ROOT₩RWorkspace에 생성됩니다.

After installation you can add the Registry entries by running RSetReg.exe in a sub-folder of the bin folder, and remove them by running this with argument /U. Note that this requires administrative privileges unless run with argument /Personal and neither sets up nor removes the file associations.

사용자는 installation 이후에 sub-folder(bin 폴더안의)에 있는 RSetReg.exe를 실행함으로써 Registry entries들을 추가할수 있고, argument /U를 실행함으로 제거할 수 있습니다. 중요한 것은 만약 argument Personal과 함께 실행하지 않거나, file associations을 제거혹은 설정하지만 않는 다면, 이것은 관리자 권한을 요구한다는 것입니다.

Does R support automation (OLE, COM)?

Directly, no. See CRAN packages rscproxy and rcom as well as RDCOMServer, RDCOMClient, RDCOMEvents and SWinTypeLibs from http://www.omegahat.org/. 지원하지 않습니다. http://www.omegahat.org/에서 CRAN packages rscproxy와 rcom 뿐만 아니라 RDCOMClient, RDCOMEvents and SWinTypeLibs를 보십시오

The Internet download functions fail.

for example update.packages() and the menu items on the Packages menu.

예를 들면, update.packages()와 packages menu위의 menu items 등이 있습니다.

We have had several reports of this, although they do work for us on all of our machines. There are two known possible fixes.

그들이 우리의 모든 기계들을 위해 일함에도 불구하고, 우리는 대한 몇개의 보고서들을 가지고 있습니다. 다음은 두개의 가능한 개선책입니다.

(a) Use the alternative internet2.dll by starting R with the flag --internet2 (see How do I install R for Windows?) or calling setInternet2(TRUE). These cause R to use the Internet Explorer internals, which may already be configured for use with proxies. Note that this does not work with proxies that need authentication.

R을 the flag --internet2함께 실행하거나 혹은 setInternet2(TRUE)를 불러와서, 대체 가능한 internet2.dll을 사용하십시오. 이것들은 R이 인터넷 익스플로러를 internals를 사용하게 합니다. Internet Explorer internals은 이미 프록시와 함께 사용되도록 설정되어 있을지도 모릅니다. 여기서 중요한 점은, 권한이 필요한 프록시와는 실행되지 않습니다.

(b) A proxy needs to be set up: see ?download.file. Here are two versions of an example (a real one,

but from a machine that is only available locally) of a command-line in a short cut: 프록시는 설정해주어야지만 사용할수 있습니다. ?download.file을 보십시오. 여기 바로가기에 있는 command-line에 대한 예들의 두가지 버전 있습니다.

```
"path_to_R\bin\x64\Rgui.exe" http_proxy=http://user:pass@gannet:80/

"path_to_R\bin\x64\Rgui.exe" http_proxy=http://gannet/ http_proxy_user=ask
```

The second version will prompt the user for the proxy username and password when HTTP downloads are first used.

두번째 버전은 HTTP downloads가 처음으로 실행 될 때, 사용자가 프록시 유저네임과 비밀번호를 갖도록 유도합니다.

Another possibility is that firewall settings are blocking the R executables from contacting the Internet, but this should result in informative error messages from the firewall program.

다른 가능성들은 방화벽 설정이 R executables를 contacting과 인터넷으로 부터 차단을 할 수 있다는 것입니다. 하지만, 이것은 방화벽 프로그램으로부터 비공식적인 에러 메시지를 발생시킵니다.

2.20 Entering certain characters crashes Rgui.

This has not been reported for a few years, but used to happen regularly. All the occurrences we have solved have been traced to faulty versions of `msvcrt.dll': we have installed a workaround that seems to avoid this. A few other people have discovered this was caused by desktop switcher and keyboard macro programs, for example `Macro Magic' and `JS Pager'.

2.21 What does 'DLL attempted to change FPU control word' mean?

This is a warning which indicates that R has taken action to correct the action of some (non-R) DLL which has just been loaded and has changed the floating point control word (in its initialization code) to a setting incompatible with that needed for R. This is not good practice on the part of the DLL, and often indicates that it needs to be updated.

Unfortunately, because DLLs may themselves load other DLLs it is not possible for R to track which DLL caused the problem.

See also ?dyn.load.

2.22 Other strange crashes.

Some users have found that Rgui.exe fails to start, exiting with a “Floating-point invalid operation” or other low level error. This error may also happen in the middle of a session. In some cases where we have tracked this down, it was due to bugs in the video driver on the system in question: it makes changes to the floating point control word which are incompatible with R. (Good practice would restore the control word to the state it was in when the driver code was called, and R tries hard to

correct this before running its own code.) For example, one user reported that the virtual screen manager JSP2 caused this crash.

These errors are essentially impossible for us to fix or work around beyond the measures already taken. The only solution we know of is for the user to replace the buggy system component that is causing the error.

2.23 Why does R never use more than 50% of my CPU?

This is a misreading of Windows' confusing Task Manager. R's computation is single-threaded, and so it cannot use more than one CPU. What the task manager shows is not the usage in CPUs but the usage as a percentage of the apparent total number of CPUs. We say `apparent' as it treats so-called `hyper-threaded' CPUs such as many Pentium 4s as two CPUs even though there is only one physical CPU.

Hyper-threading has been re-introduced for some Intel i3/i5/i7, Atom and Xeon CPUs: these will usually be reported as 4 or more CPUs and so R will be shown as using 25% or less. You can see how many `CPU's are assumed by looking at the number of graphs of `CPU Usage History' on the `Performance' tab of the Windows Task manager.

2.24 Does R run under Windows Vista/7/Server 2008?

It does. A few issues have been reported that are related to the way accounts and file permissions work. (These are not specifically R issues, but changes in user experiences.)

Earlier versions of Windows had user and Administrator accounts, and user accounts could be give administrative privileges (by being added to the local Administrators group) and so write permission over system areas such as c:₩Program Files. R would be installed either by a user in his own file space or by an account with administrator privileges into a system area. Sysadmins could set policies for user accounts, and you might for example have needed to be a `Power User' to install software at all.

Vista and later normally disable the Administrator account and expects software installation to done by an account which is in the local Administrator group with `admin approval mode' turned on. (The Administrator account by default has it turned off.) Unlike (say) Windows XP, such accounts do not run programs with full administrator privileges, and this is where the issues arise. (For background information consult e.g. http://windowsvistablog.com/blogs/windowsvista/archive/2007/01/23/security-features-vs-convenience.aspx.) These OSes have the concept of `over-the-shoulder' credentials: if you are running without full administrator privileges and do something which needs them you may be prompted with one or more security-check dialog boxes, and may be required to provide administrator credentials or confirm that you really want to take that action.

Vista will report that the R installer has an `unidentified publisher' (Windows 7 uses `unknown publisher') and ask if it should be run. System administrators can disable installing applications from non-trusted sources, in which case you will have to persuade them that R is trustworthy, or digitally sign the R installer yourself, or (unless this is also disabled) run the installer from a standard account and install into your own file area. (The same issues apply to the .msi version of the installer.)

If you install R as a standard user into your own file space and use it under the same account, there are no known permission issues.

If you use the default Administrator account (without `admin approval mode' being turned on) and install/update packages (in the system area or elsewhere), no issues are known.

If you use an account in the local Administrators group in `admin approval mode' (which is the intended norm under these OSes), installation will make use of `over-the-shoulder' credentials. You will run into problems if you try installing (including updating) packages in the main R library. (It would be nice if at that point R could use over-the-shoulder credentials, but they apply to processes as a whole. Vista and later disallow creating .dll files in the system area without credentials.) There are several ways around this.

Run R with Administrator privileges in sessions where you want to install packages. (Do so by right-clicking on the R shortcut and selecting 'Run as Administrator'.) Transfer ownership of the R installation to the user which installed R. To do so, use the security tab on the `Properties' of the top-level R folder and give `Full Control' over this directory to the user (not just the Administrator group). Install packages into a different library tree owned by the account used to install R. For an installation to be used by a single user, the simplest way is to make use of a `personal library': See I don't have permission to write to the R-2.15.0₩library directory.

For a site installation, you can create a site-wide library directory anywhere convenient, and add it to the default package search path for all users via R_LIBS_SITE in etc₩Renviron.site. See What are HOME and working directories?. There is a standard location for a site library, the site-library directory in the top-level R folder (which you would need to create with full control for the R installation account). This will be used for installation in preference to the main library folder if it exists.

This approach will not allow you to update the recommended packages unless you `Run as administrator': we suggest you use an R session running under Administrator privileges when updating those.

Another issue with Vista was that the standard POSIX ways that R uses (e.g. in file.info and file.access) to look at file permissions no longer work reliably. file.access was re-written to work with Windows NT-based security and the new version seems much more reliable with these OSes (but still not 100% correct).

On suitably recent hardware Vista and later can prevent the execution of code from data areas via `Data Execution Prevention' (from a tab in System Properties -> Advanced -> Performance), and sysadmins can turn this on for all programs. R runs correctly with DEP enabled.

2.25 Quotes don't come out right on the console/terminal/pager.

R may make use of directional quotes that are not always rendered correctly by Windows: these are used by default only by Rgui in suitable locales (not Chinese/Japanese/Korean).

Whether these are used in R output (from functions sQuote and dQuote) is controlled by getOption("useFancyQuotes") whose default is FALSE except for the Rgui console. There are two potential problems with rendering directional quotes. The first is with running Rterm: in European locales the `Windows Command Prompt' is by default set up to use MS-DOS and not Windows default encodings: this can be changed via chcp, with chcp 1252 being appropriate for Western European (including English) locales. The other is that the default raster fonts only include directional single quotes and not directional double quotes (which will probably be rendered as a filled rectangle).

Directional quotes will also be used in text help which is normally displayed in R's internal pager: these may not be rendered correctly in an external pager. They are also used in HTML help, where most browsers use fonts which render them correctly.

The font used can affect whether quotes are rendered correctly. The default font in the Rgui console and internal pager is Courier New, which has directional quotes on all the systems we tried. Lucida Console has elegant glyphs for directional quotes (but seems rather light unless ClearType is in use): Consolas is another font which we often select when ClearType is in use. Non-TrueType fonts such as Courier and FixedSys lack directional double quotes on all the systems we tried.

There is a related problem with using Sweave output in Rgui, for LaTeX needs to be told about the encoding of directional quotes by including in the LaTeX preamble e.g. (for a Western European locale)

```
\usepackage[cp1252]{inputenc}
```

or their use suppressed by options(useFancyQuotes=FALSE).

2.26 There is no tilde on my keyboard!

Where tilde does not appear on the main keyboard, it can normally be accessed by pressing AltGr (the right Alt key) plus some other key. This is] in Canadian (multilingual), German and Scandinavian layouts, 1 in Eastern Europe, [in Portuguese, 4 or 5 in Spanish, / in Francophone Belgian, and so on. For some keyboards the uses of AltGr are at http://office.microsoft.com/en-us /word/HP052590631033.aspx: you can explore those for your keyboard via the `On-Screen Keyboard' (under Accessories, Accessibility on XP, Accessories, Ease of access on Windows 7).

On all Windows versions you should be able to get tilde by holding the down the left Alt key and typing 0126 on the numeric keypad (if you have one), then releasing the Alt key.

2.27 Can I use R on 64-bit Windows?

The 32-bit build of R for Windows will run on both 32-bit and 64-bit2 versions of Windows. 64-bit versions of Windows run 32-bit executables under the WOW (Windows on Windows) subsystem: they run in almost exactly the same way as on a 32-bit version of Windows, except that the address limit for the R process is 4GB (rather than 2GB or perhaps 3GB).

When R is installed on 64-bit Windows there is the option of installing 32- and/or 64-bit builds: the default is to install both. The 64-bit build has 'x64' suffixed to its name on the Start menu and desktop shortcuts. If you are using the 32-bit build, replace `x64' by `i386' in the examples in this FAQ.

2.28 Should I run 32-bit or 64-bit R?

Obviously, only relevant if you are using 64-bit Windows.

For most users we would recommend using the `native' build, that is the 32-bit version on 32-bit Windows and the 64-bit version of 64-bit Windows.

The advantage of a native 64-bit application is that it gets a 64-bit address space and hence can address far more than 4GB (how much depends on the version of Windows, but in principle 8TB). This allows a single process to take advantage of more than 4GB of RAM (if available) and for R's memory manager to more easily handle large objects (in particular those of 1GB or more). The disadvantages are that all the pointers are 8 rather than 4 bytes and so small objects are larger and more data has to be moved around, and that less external software is available for 64-bit versions of the OS. The 64-bit compilers are able to take advantage of extra features of all x86-64 chips (more registers, SSE2/3 instructions, ...) and so the code may run faster despite using larger pointers. The 64-bit build is nowadays usually slightly faster than the 32-bit build on a recent CPU (Intel Core 2 or later or AMD equivalent).

For advanced users the choice may be dictated by whether the contributed packages needed are available in 64-bit builds (although CRAN only offers 32/64-bit builds). The considerations can be more complex: for example 32/64-bit RODBC need 32/64-bit ODBC drivers respectively, and where both exist they may not be able to be installed together. An extreme example is the Microsoft Access/Excel ODBC drivers: if you have installed 64-bit Microsoft Office you can only install the 64-bit drivers and so need to use 64-bit RODBC and hence R. (And similarly for 32-bit Microsoft Office.)

2.29 Can both 32- and 64-bit R be installed on the same machine?

Obviously, only relevant if the machine is running a 64-bit version of Windows - simply select both when using the installer. You can also go back and add 64-bit components to a 32-bit install, or vice versa.

For many Registry items, 32- and 64-bit programs have different views of the Registry, but clashes can occur. The most obvious problem is the file association for .RData files, which will use the last installation for which this option is selected, and if that was for an installation of both, will use 64-bit R. To change the association the safest way is to edit the Registry entry `HKEY_CLASSES_ROOT₩RWorkspace₩shell₩open₩command' and replace `x64' by `i386' or vice versa.

2.30 Rcmd is not found in my PATH!

This has often been reported after an upgrade.

The R installer does not put Rcmd.exe (nor any other R executable) on your PATH. What seems to have happened is that people did this for themselves in the past, upgraded R (which by default will install to a different location) and un-installed the old version of R. If you do that (or install R for the first time), you need to edit the PATH.

The element you want to add to the path is something like

```
c:\Program Files\R\R-2.14.0\bin\x64
```

for 64-bit Rcmd.exe, replacing x64 by i386 for 32-bit.

How you set the path depends on your OS version. On Windows XP/Server 2003 you can use `System'

in the control panel or the properties of `My Computer' (under the `Advanced' tab). Under Vista/7 /Server 2008, go to `User Accounts' in the Control Panel, and select your account and then `Change my environment variables'. (System policies can prevent end users making changes.)

An alternative is to set the PATH in the shell you are running (Rcmd.exe is a command-line program). For those using the standard Windows `Command Prompt' Duncan Murdoch suggested:

The simple way to do it just for the command prompt is to write a little batch file setpath.bat containing

```
set PATH=newstuff;%PATH%
```

and then run cmd with

```
CMD /K setpath.bat
```

원본 주소 "http://r-project.kr/w/index.php?title=R_FAQ_Windows&oldid=1906"

- 이 문서는 2012년 5월 4일 (금) 01:33에 마지막으로 바뀌었습니다.
- 이 문서는 660번 읽혔습니다.