

1. Versions of R on the ACCRE Cluster
2. Checking Installed Packages
3. Installing New Packages
4. Example Scripts
5. Contributing New Examples
6. FAQ

R is a widely used statistical analysis environment and programming language. Many versions of R are available to use on the cluster. Users typically first develop code interactively on their laptop/desktop, and then run batch processing jobs on the ACCRE cluster through the SLURM job scheduler.

1 Versions of R on the ACCRE Cluster

When a user initially logs into the cluster, the system version (R that comes pre-installed on the operating system) is 3.1.0 and is located in `/usr/local/bin`:

```
[bob@vmps65 ~]$ which R
/usr/bin/R
[bob@vmps65 ~]$ R --version
R version 3.1.0 (2014-04-10) -- "Spring Dance"
Copyright (C) 2014 The R Foundation for Statistical Computing
Platform: x86_64-redhat-linux-gnu (64-bit)
```

R is free software and comes with ABSOLUTELY NO WARRANTY.
 You are welcome to redistribute it under the terms of the
 GNU General Public License versions 2 or 3.
 For more information about these matters see

<http://www.gnu.org/licenses/>.

This version of R may be sufficient for simple tasks, but it does not include many of the essential numerical libraries cluster users need. ACCRE administrators have hand-compiled multiple other versions of R that are linked against highly optimized linear algebras like ATLAS and Intel's MKL, and therefore will in general yield better performance (faster execution time). We encourage users to use the most recent version (3.2.0) installed. This version was built with Intel compilers and linked against Intel's MKL library, and should therefore yield better performance on ACCRE's Intel processors.

```
[bob@vmps65 ~]$ pkginfo | grep R_3.2.0
R_3.2.0  R (3.2.0) statistical programming language (Intel 14 / x86_64)
```

Multiple other versions of R are available on the cluster but users should use version 3.2.0 unless they have a good reason for not doing so. We can also not guarantee that we will continue supporting older versions of R on the cluster.

To list find all R versions available on the cluster, pipe the result of *pkginfo* into *grep* :

```
[bob@vm65 ~]$ pkginfo | grep "\s*R"
      R R (2.14.0) statistical programming language (GCC 4.6 / x86_64)
R_2.15.0 R (2.15.0) statistical programming language (GCC 4.6 / x86_64)
R_3.0.0 R (3.0.0) statistical programming language (GCC 4.6 / x86_64)
R_3.0.2 R (3.0.2) statistical programming language (GCC 4.6 / x86_64)
R_3.1.1 R (3.1.1) statistical programming language (GCC 4.6 / x86_64)
R_3.1.1_intel R (3.1.1) statistical programming language (Intel 14 / x86_64)
R_3.2.0 R (3.2.0) statistical programming language (Intel 14 / x86_64)
R_3.2.0_gcc R (3.2.0) statistical programming language (GCC 4.6 / x86_64)
R_3.2.2_gcc R (3.2.2) statistical programming language (GCC 4.6 / x86_64)
      R_gpu R (2.14.0) GPU version - statistical programming language (GCC 4.6 / x86_64)
R_x86-64_gcc R (2.14.0) statistical programming language (GCC 4.6 / x86_64)
R_x86-64_intel R (2.14.0) statistical programming language (Intel 12 / x86_64)
```

To load a particular version of R, use *setpkgs* :

```
[bob@vm65 ~]$ setpkgs -a R_3.2.0
[bob@vm65 ~]$ R --version
R version 3.2.0 (2015-04-16) -- "Full of Ingredients"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-unknown-linux-gnu (64-bit)
```

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under the terms of the
GNU General Public License versions 2 or 3.
For more information about these matters see

<http://www.gnu.org/licenses/>.

```
[bob@vm65 ~]$ which R
/usr/local/R/3.2.0/x86_64/intel14/nonet/bin/R
```

2 Checking Installed Packages

Each different version of R has its own packages installed into it, so as you are switching between different versions it is prudent to check the packages that are available. One simple way to do this is by typing *installed.packages()* from the R command prompt. For example:

```
[frenchwr@vm65 ~]$ R

R version 3.2.0 (2015-04-16) -- "Full of Ingredients"
Copyright (C) 2015 The R Foundation for Statistical Computing
```

Platform: x86_64-unknown-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

```
> installed.packages()
      Package
base      "base"
boot      "boot"
class     "class"
cluster   "cluster"
codetools "codetools"
compiler  "compiler"
datasets  "datasets"
foreign   "foreign"
graphics  "graphics"
grDevices "grDevices"
grid       "grid"
KernSmooth "KernSmooth"
lattice    "lattice"
MASS       "MASS"
Matrix     "Matrix"
methods    "methods"
mgcv       "mgcv"
nlme       "nlme"
nnet       "nnet"
parallel  "parallel"
rpart      "rpart"
spatial    "spatial"
splines    "splines"
stats      "stats"
stats4     "stats4"
survival   "survival"
tcltk      "tcltk"
tools      "tools"
```

```
utils      "utils"
```

Note that the above output has been truncated for brevity. If you were to run this command you would see additional information about installed packages: the path to the package, version, dependencies, license information, and a few other details. To load a package into a R session simply type `library("package_name")` . For example, to load the parallel package one would need to type:

```
library("parallel")
```

3 Installing New Packages

If you find that a particular package you need is missing from the R version you use, you have a few options. If you believe the package will be widely used by other cluster users, you can open a helpdesk ticket and request that we install the package for cluster-wise access. The other option is to install the package yourself into your home directory. There are multiple ways to install R packages. Below is an example of how you would go about installing a package from the R command prompt. To begin, create a directory in your home directory to install these packages into. In this example, the packages will be installed into a directory at `~/R/rlib`:

```
[bob@vmmps65 ~]$ mkdir -p ~/R/rlib
```

Now load R and start up an R session from the terminal. In this example we will install the Zelig package.

```
[bob@vmmps65 ~]$ setpkgs -a R_3.2.0
[bob@vmmps65 ~]$ R
```

```
R version 3.2.0 (2015-04-16) -- "Full of Ingredients"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-unknown-linux-gnu (64-bit)
```

```
R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.
```

```
  Natural language support but running in an English locale
```

```
R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.
```

```
Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
```

Type 'q()' to quit R.

```
> .libPaths("~/R/rlib")
> install.packages("Zelig")
Installing package into '/gpfs22/home/bob/R/rlib'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
CRAN mirror

1: 0-Cloud
3: Argentina (La Plata)
5: Australia (Melbourne)
7: Belgium
9: Brazil (PR)
11: Brazil (SP 1)
13: Canada (BC)
15: Canada (ON)
17: Canada (QC 2)
19: China (Beijing 1)
21: China (Beijing 3)
23: China (Hefei)
25: China (Xiamen)
27: Czech Republic
29: Ecuador
31: Estonia
33: France (Lyon 2)
35: France (Paris 2)
37: Germany (Berlin)
39: Germany (Frankfurt)
41: Greece
43: Iceland
45: Indonesia (Jakarta)
47: Ireland
49: Italy (Padua)
51: Japan (Tokyo)
53: Korea (Seoul 1)
55: Korea (Ulsan)
57: Mexico (Mexico City)
59: Netherlands (Amsterdam)
61: New Zealand
63: Philippines
65: Portugal
67: Russia (Moscow 2)
69: Slovakia
71: Spain (A Coruña)
73: Sweden

2: Algeria
4: Australia (Canberra)
6: Austria
8: Brazil (BA)
10: Brazil (RJ)
12: Brazil (SP 2)
14: Canada (NS)
16: Canada (QC 1)
18: Chile
20: China (Beijing 2)
22: China (Beijing 4)
24: China (Lanzhou)
26: Colombia (Cali)
28: Denmark
30: El Salvador
32: France (Lyon 1)
34: France (Montpellier)
36: France (Strasbourg)
38: Germany (Goettingen)
40: Germany (Münster)
42: Hungary
44: India
46: Iran
48: Italy (Milano)
50: Italy (Palermo)
52: Japan (Yamagata)
54: Korea (Seoul 2)
56: Lebanon
58: Mexico (Texcoco)
60: Netherlands (Utrecht)
62: Norway
64: Poland
66: Russia (Moscow 1)
68: Singapore
70: South Africa (Johannesburg)
72: Spain (Madrid)
74: Switzerland
```

75: Taiwan (Chungli)	76: Taiwan (Taipei)
77: Thailand	78: Turkey
79: UK (Bristol)	80: UK (Cambridge)
81: UK (Hampshire)	82: UK (London)
83: UK (London)	84: UK (St Andrews)
85: USA (CA 1)	86: USA (CA 2)
87: USA (IA)	88: USA (IN)
89: USA (KS)	90: USA (MD)
91: USA (MI 1)	92: USA (MI 2)
93: USA (MO)	94: USA (OH 1)
95: USA (OH 2)	96: USA (OR)
97: USA (PA 1)	98: USA (PA 2)
99: USA (TN)	100: USA (TX 1)
101: USA (WA 1)	102: USA (WA 2)
103: Venezuela	104: Vietnam

Here we are prompted for the repository we would like to download the package from. Let's choose the Tennessee repository (option 99):

Selection: 99

also installing the dependencies 'zoo', 'sandwich'

trying URL 'http://mirrors.nics.utk.edu/cran/src/contrib/zoo_1.7-12.tar.gz'

Content type 'application/x-gzip' length 839181 bytes (819 KB)

=====

downloaded 819 KB

trying URL 'http://mirrors.nics.utk.edu/cran/src/contrib/sandwich_2.3-3.tar.gz'

Content type 'application/x-gzip' length 466503 bytes (455 KB)

=====

downloaded 455 KB

trying URL 'http://mirrors.nics.utk.edu/cran/src/contrib/Zelig_4.2-1.tar.gz'

Content type 'application/x-gzip' length 3262531 bytes (3.1 MB)

=====

downloaded 3.1 MB

* installing *source* package 'zoo' ...

** package 'zoo' successfully unpacked and MD5 sums checked

** libs

icc -std=gnu99 -I/usr/local/R/3.2.0/x86_64/intel14/nonet/lib64/R/include

-DNDEBUG -I../inst/include -I/usr/local/include -fpic -O3 -msse3

-funroll-loops -funsigned-char -c coredata.c -o coredata.o

icc -std=gnu99 -I/usr/local/R/3.2.0/x86_64/intel14/nonet/lib64/R/include

-DNDEBUG -I../inst/include -I/usr/local/include -fpic -O3 -msse3

-funroll-loops -funsigned-char -c init.c -o init.o

icc -std=gnu99 -I/usr/local/R/3.2.0/x86_64/intel14/nonet/lib64/R/include

```

-DNDEBUG -I../inst/include -I/usr/local/include -fpic -O3 -msse3
-funroll-loops -funsigned-char -c lag.c -o lag.o
icc -std=gnu99 -shared -L/usr/local/lib64 -o zoo.so coredata.o init.o lag.o
installing to /gpfs22/home/frenchwr/R/rlib/zoo/libs
** R
** demo
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded
* DONE (zoo)
* installing *source* package 'sandwich' ...
** package 'sandwich' successfully unpacked and MD5 sums checked
** R
** data
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded
* DONE (sandwich)
* installing *source* package 'Zelig' ...
** package 'Zelig' successfully unpacked and MD5 sums checked
** R
** data
** demo
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded
* DONE (Zelig)

```

The downloaded source packages are in
 '/tmp/RtmpmGSgLS/downloaded_packages'

Notice that Zelig had a few dependencies (zoo and sandwich) that were also installed along the way. It appears that the installation was successful, so let's exit the R session to check if the packages are now in our home directory:

```

> quit()
Save workspace image? [y/n/c]: n
[Bob@vm65 ~]$ ls ~/R/rlib/
sandwich  Zelig  zoo

There they are! Finally, let's re-start R to make sure we can load the package
we've installed:

[Bob@vm65 ~]$ R

R version 3.2.0 (2015-04-16) -- "Full of Ingredients"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-unknown-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> .libPaths("~/R/rlib")
> library("Zelig")
Loading required package: boot
Loading required package: MASS
Loading required package: sandwich
ZELIG (Versions 4.2-1, built: 2013-09-12)

+-----+
| Please refer to http://gking.harvard.edu/zelig for full |
| documentation or help.zelig() for help with commands and |
| models support by Zelig. |
| |
| Zelig project citations: |
| Kosuke Imai, Gary King, and Olivia Lau. (2009). |
| ``Zelig: Everyone's Statistical Software,' |
| http://gking.harvard.edu/zelig |
| and |
| Kosuke Imai, Gary King, and Olivia Lau. (2008). |
| ``Toward A Common Framework for Statistical Analysis |

```



```
| and Development, '' Journal of Computational and |
| Graphical Statistics, Vol. 17, No. 4 (December) |
| pp. 892-913. |
| |
| To cite individual Zelig models, please use the citation |
| format printed with each model run and in the documentation. |
+-----+
```

Attaching package: ‘Zelig’

The following object is masked from ‘package:utils’:

`cite`

Zelig appears to load properly, confirming we have successfully installed the package. Note that we first needed to type:

```
> .libPaths("~/R/rlib")
```

in order to point R to the directory where our packages are installed. This command was also need before installing the packages. Alternatively, you may drop this line in your .RProfile if you always want R to see these libraries. Note that these packages were installed for a specific version of R, so it’s unlikely that they will work for a different version. If you need information on installing a package from source code or from Bioconductor, refer to our FAQ page.

4 Example Scripts

Running a R script within a SLURM job is generally straightforward. Unless you are attempting to run one of R’s multi-processing packages, you will want to request a single task, load the appropriate version of R from your SLURM script, and then run your script using the *Rscript* command. The *-no-save* flag passed to *Rscript* prevents R from saving the workspace, which in this example would be relatively large. The following example runs R 3.2.0 on a simple R script that demonstrates the utility of writing vectorized R code:

```
[bob@vmmps65 run1]$ ls
R.slurm  vectorize.R

[bob@vmmps65 run1]$ cat R.slurm
#!/bin/bash
#SBATCH --nodes=1
#SBATCH --ntasks=1
#SBATCH --time=00:10:00
#SBATCH --mem=500M
#SBATCH --output=R_job_slurm.out
```

```
setpkgs -a R_3.2.0
```

```
Rscript --no-save vectorize.R
```

```
[bob@vm65 run1]$ cat vectorize.R
```

```
n = 10^7
```

```
# populate with random nos
```

```
v=runif(n)
```

```
system.time({vv<-v*v; m<-mean(vv)}); m
```

```
system.time({for(i in 1:length(v)) { vv[i]<-v[i]*v[i] }; m<-mean(vv)}); m
```

Note this example was taken from a Stackoverflow thread . We next submit the job with *sbatch* :

```
[bob@vm65 run1]$ sbatch R.slurm
```

```
Submitted batch job 2271536
```

After waiting a few minutes:

```
[bob@vm65 run1]$ ls
```

```
R_job_slurm.out R.slurm vectorize.R
```

```
[bob@vm65 run1]$ cat R_job_slurm.out
```

```
user system elapsed
```

```
0.047 0.014 0.062
```

```
[1] 0.3333861
```

```
user system elapsed
```

```
20.158 0.058 20.253
```

```
[1] 0.3333861
```

The elapsed column indicates that the vectorized version of the code executed in 0.062 seconds while the non-vectorized section executed in 20.253 seconds. Both versions produced identical results (0.3333861). Moral of the story: used vectorized code in R scripts whenever possible!

5 Contributing New Examples

In order to foster collaboration and develop local R expertise at Vanderbilt, we encourage users to submit examples of their own to ACCRE's R Github repository . Instructions for doing this can be found on this page .

6 FAQ

When trying to install a package, I get ‘Warning: unable to access index for repository <https://some.repository.name>’.

From an R interactive session, try:

```
> install.packages('package_name', dependencies=TRUE, repos='http://cran.rstudio.com/')
```

When trying to install a package, I get ‘Warning message: package ‘somepackage’ is not available (for R version 3.0.0) ‘.

Try using the latest version of R.
