

# CIBM R utils

**A collection of utilities and functions in R for the adventurous.**

R. Vimieiro, C. Riveros

January 2014

This document serves as a short user manual for the utilities and functions included in this package, with some usage examples.

## 1 Introduction

This package grew out of disparate needs, and from the increased use of R that we make in the lab. It contains a series of utility functions to convert file formats, some interesting functions to compute distance matrices, ranking functions, an improved interface to some missing *Weka* engine functionality and some graph-based methods.

We collect in this manual a few usage examples, to bridge the gap of the deficient online documentation.

### 1.1 Installing `cibm.utils`

**Prerequisites** The following packages are required: `igraph`, `RWeka`, `rJava`, `devtools`, `plyr`.

**Install from repos:** There are a couple of ways to install `cibm.utils` in your current R environment. The best way is from the git repositories, updates work well this way. However, this requires you have correctly configured access to the CIBM git repos at fisher (out of the scope of this document).

```
> library("devtools")
> install_git("ssh://git@fisher/source/cibm-R-utils")
```

```
Preparing installation of cibm-R-utils using the Git-URL: ssh://git@fisher/source/cibm-R-utils
'/bin/git'clone --depth 1 --no-hardlinks ssh://git@fisher/source/cibm-R-utils /tmp/Rtmpr2L3F8c7
Installing package from /tmp/Rtmpr2L3F8c7/file2fd368856fc7
Installing cibm.utils
'/usr/lib64/R/bin/R' --vanilla CMD INSTALL '/tmp/Rtmpr2L3F8c7' \
  --library='/home/carlos/R/x86_64-redhat-linux-gnu-library/3.0' \
  --install-tests
```

```
* installing *source* package 'cibm.utils' ...
** R
** data
** inst
** preparing package for lazy loading
```

```

Creating a generic function for 'format' from package 'base' in package 'cibm.utils'
Creating a generic function for 'print' from package 'base' in package 'cibm.utils'
** help
*** installing help indices
    converting help for package 'cibm.utils'
      finding HTML links ... done
      CM-functions                html
      RNGraph                     html
      a2695                       html
      alzheimer                   html
      cibm-distances              html
      cibm-utils-package          html
      cibm.abk-class              html
      cibm.abk-methods            html
      cibm.data-class             html
      cibm.data-methods          html
      getClassifiers              html
      kNNCliques                  html
      predict_XVal                html
Rd warning: /tmp/Rtmpr2L3F8/file2fd368856fc7/man/predict_XVal.Rd:8: file link 'Weka_classifier'
in package 'RWeka' does not exist and so has been treated as a topic
      prob-matrix-comp           html
      read.abk                    html
      read.distance               html
      read.nbi                    html
      runWekaClassifiers          html
      write.abk                   html
      write.distance              html
      write.nbi                   html
** building package indices
** testing if installed package can be loaded
* DONE (cibm.utils)

```

**Install from source file:** Download the source tar file from the CIBM wiki and install it as follows:

```

> # For version 0.1, use:
> install.packages("cibm.utils_0.1.tar.gz",repos=NULL,type="source")

```

```

Installing package into '/home/renato/R/x86_64-pc-linux-gnu-library/3.0'
(as 'lib' is unspecified)
inferring 'repos = NULL' from the file name
* installing *source* package 'cibm.utils' ...
** R
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** testing if installed package can be loaded
* DONE (cibm.utils)

```

**Using cibm.utils:** It all begins with the usual stanza:

```

> library(cibm.utils)

```

```
Loading required package: igraph
Loading required package: RWeka
Loading required package: plyr
Loading required package: rJava
```

```
Attaching package: 'cibm.utils'
```

The following object is masked from `base`:

```
labels
```

In what follows, we will assume you have loaded the `cibm.utils` library.

## 2 Data I/O

### 2.1 NBI Format

The NBI format is a badly thought format, but several applications still use this format. It is a ‘pseudo XML’ format which wraps a matrix. `cibm.utils` comes with a small sample data file for this examples.

```
> # Load library
> library(cibm.utils)
> # defines filename
> filename <- system.file("extdata","alzheimer.nbi",package="cibm.utils")
> # reads
> alzheimer <- read.nbi(filename)
> # prints -- Notice that class is numeric
> dim(alzheimer@data)

[1] 120  83

> alzheimer[c(1:6,121),1:5]

      S1      S2      S3      S4      S5
ANG_1  4.5201044 4.4302240 4.2445931  2.90625036  3.10678080
BDNF_1  2.6926477 0.9458598 2.9161823 -0.26235322  0.08482655
BLC_1   0.7883548 0.2430130 1.0296384 -0.65142870 -0.53014371
BMP.4_1  1.1420449 0.4074050 1.9665506 -0.03288723  0.23938901
BMP.6_1 -0.3769505 0.4802725 2.3977701 -0.45563567 -0.56747729
CK_b8.1_1 0.3123880 0.0823939 0.9957698 -0.54465536 -0.49707862
NA      NA      NA      NA      NA      NA

> # Write data to NBI formatted file -- Notice by default the file is gzipped
> write.nbi(alzheimer,file="alzheimer-test.nbi",gzip=FALSE)
```

### 2.2 ABk Format

The ABk format is used by the  $(\alpha, \beta) - k$  Feature Set selection programs. It is an *ad-hoc* format, and has a series of options that will get read or written depending on their presence in the file. Below some examples of usage:

```

> # loads library
> library("cibm.utils")
> # defines filename
> filename <- system.file("extdata","2695.abk",package="cibm.utils")
> # reads
> a2695 <- read.abk(filename)
> # prints -- Notice that class is numeric
> dim(a2695@data)

[1] 100 22

> a2695[c(1:6,100),1:8]
      GSM90037 GSM90040 GSM90043 GSM90047 GSM90051 GSM90052 GSM90069 GSM90078
f644         0         1         0         1         0         0         0         0
f1260         1         1         0         1         0         0         1         0
f1402         0         0         0         1         0         0         0         0
f2267         0         1         0         1         0         0         0         0
f2866         0         1         0         1         0         0         0         1
f3041         0         1         0         1         0         0         0         0
f44491        0         1         0         0         0         0         0         0

> # Write data to NBI formatted file -- Notice by default the file is gzipped
> write.abk(a2695,file="a2695-test.abk",classes="first",out.equalweights=T,gzip=F)
> a2695@caseAttr["caseweights"]

      caseweights
1             1
2             1
3             1
4             1
5             1
6             1
7             1
8             1
9             1
10            1
11            1
12            1
13            1
14            1
15            1
16            1
17            1
18            1
19            1
20            1
21            1
22            1

> # Some made-up values
> a2695@featureAttr["featurecolors"] <- rep(c(1,2,3),length.out=100)
> a2695@caseAttr["casecolors"] <- c(1,2)
> a2695@beta <- c(0,1)
> write.abk(a2695,file="a2695-test2.abk",classes="first",out.equalweights=T,gzip=F)

```

**3 Distances**

**4 Classes**

**5 History**