

# Package ‘RbcBook1’

July 21, 2025

**Version** 1.76.0

**Title** Support for Springer monograph on Bioconductor

**Author** Vince Carey <stvjc@channing.harvard.edu> and Wolfgang Huber <huber@ebi.ac.uk>

**Maintainer** Vince Carey <stvjc@channing.harvard.edu>

**Description** tools for building book

**Depends** R (>= 2.10), Biobase, graph, rpart

**License** Artistic-2.0

**URL** <http://www.biostat.harvard.edu/~carey>

**biocViews** Software

**git\_url** <https://git.bioconductor.org/packages/RbcBook1>

**git\_branch** RELEASE\_3\_21

**git\_last\_commit** 5c9602a

**git\_last\_commit\_date** 2025-04-15

**Repository** Bioconductor 3.21

**Date/Publication** 2025-07-20

## Contents

|                               |   |
|-------------------------------|---|
| bcr.cor . . . . .             | 2 |
| bcStangle . . . . .           | 2 |
| bcSweave . . . . .            | 3 |
| checkingBookSources . . . . . | 4 |
| imageSize . . . . .           | 5 |
| perf . . . . .                | 6 |
| require.RbcBook1 . . . . .    | 6 |
| rpart2gNEL . . . . .          | 7 |

|              |          |
|--------------|----------|
| <b>Index</b> | <b>9</b> |
|--------------|----------|

---

bcr.cor

*Illustrative datasets for distance measures*


---

**Description**

Illustrative datasets for distance measures

**Usage**

```
data(ALL.dist) # all the components live in here
```

**Details**

These datasets are used in conjunction with the bioDist package for the distance chapter of the monograph

**Value**

these are `dist` objects

**Author(s)**

Vince Carey <stvjc@channing.harvard.edu>

**Examples**

```
data(ALL.dist)
objects()
dim(as.matrix(bcr.cor))
```

---

bcStangle

*A function to extract the code chunks from the book for the online supplement*


---

**Description**

A function to extract the code chunks from the book for the online supplement.

**Usage**

```
bcStangle(files = .RbcBook1Files(), outfile= "bioCSpringer.R")
tangleToSingleFiles()
```

**Arguments**

|         |                                                                            |
|---------|----------------------------------------------------------------------------|
| files   | character vector with filenames of book chapters. Must have extension .Rnw |
| outfile | character of length 1 with the name of the output file.                    |

**Details**

It should suffice to change to the book's top level directory (e.g. `.../Springer1/trunk/Book1`) and run `bcStangle()`.

This function is specialized to its one purpose given in the title. It is not envisaged that it could be useful beyond, or even replace Stangle.

`tangleToSingleFiles` creates a directory named `Rfiles` and tangles each chapter separately into this directory. You must have the current working directory set to the top level directory of the book (`Springer/Book1`).

**Value**

Function is called for its side effects.

**Author(s)**

Wolfgang Huber <huber@ebi.ac.uk>

**See Also**

`.RbcBook1Files`, `Stangle`

**Examples**

```
## bcStangle()
```

---

bcSweave

*Wrapper for Sweave*

---

**Description**

Wrapper for Sweave - allows to set options, or perform additional preprocessing or QC steps.

**Usage**

```
bcSweave(f)
```

**Arguments**

`f` character, filename. Must have extension `.Rnw`

**Details**

No details.

**Value**

Function is called for its side effects.

**Author(s)**

Wolfgang Huber <huber@ebi.ac.uk>

**Examples**

```
## bcSweave("Intro.Rnw")
```

---

checkingBookSources     *Formatting and standardization checks on book chapters*

---

**Description**

Formatting and standardization checks on book chapters

**Usage**

```
checkVerbatim(files = .RbcBook1Files(ext=".tex"),
               maxc = 70,
               which = "both",
               verbose = TRUE)
```

```
checkRnw(files = .RbcBook1Files(),
          verbose = TRUE, stopOnError=FALSE)
```

```
checkPackage(files = .RbcBook1Files(ext=".Rnw"),
              verbose = TRUE)
```

```
.RbcBook1Files(ext=".Rnw")
```

**Arguments**

|             |                                                                                                                                                             |
|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| files       | character vector with names (and path) of chapter source files                                                                                              |
| ext         | character of length 1. File name extension. .Rnw (the default) and .tex should be the most important cases.                                                 |
| maxc        | integer of length 1. Maximal number of characters in a verbatim line. All lines exceeding this limit will be reported in the return value of this function. |
| which       | character of length 1. 'Sinput' will look at Sinput environments, 'Soutput' at Soutput environments, 'both' at both.                                        |
| verbose     | Logical.                                                                                                                                                    |
| stopOnError | Logical. If FALSE, first error found will lead to stop. If TRUE, try to continue checking.                                                                  |

**Details**

checkVerbatimLines sees whether all verbatim lines have length at most maxc.

checkPackage finds all the occurrences of \Rpackage{...} in the text, checks whether the package is known, and returns a named list with all occurrences of the packages.

**Value**

For checkVerbatimLines and checkSetup, a data frame with one row for each offending line and various columns describing it.

For .RbcBook1Files, a character vector.

**Author(s)**

Wolfgang Huber <huber@ebi.ac.uk>

**Examples**

```
f = tempfile()
zap = function(n) paste(formatC(1:n, width=2), collapse=" ")
writeLines(c("*begin{Sinput}", zap(20), zap(30), "*end{Sinput}"), con=f)
checkVerbatim(files=f)
```

---

imageSize

*File info for all image files in the book*

---

**Description**

File info for all image files in the book

**Usage**

```
imageSize(dirs= c("Preproc", "Analysis", "Metadata", "Graphs", "CaseStudies"), ext=c("pdf", "png"))
```

**Arguments**

dirs                      character vector with directories.  
ext                        character vector with file name extensions.

**Details**

No details.

**Value**

Data frame.

**Author(s)**

Wolfgang Huber <huber@ebi.ac.uk>

**Examples**

```
# a = imageSize()
# print(a[1:10,c(1,4)])
```

---

perf

*Results of tedious computations for computational inference*


---

**Description**

Results of tedious computations for computational inference

**Usage**

```
data(Survperformance)
```

**Details**

These datasets are used in conjunction with the computational inference chapter of the monograph

**Value**

these are `data.frame` objects

**Author(s)**

Vince Carey <stvjc@channing.harvard.edu>

**Examples**

```
data(Survperformance)
objects()
dim(performance)
```

---

require.RbcBook1

*Load a lengthy list of packages that are used in the Book*


---

**Description**

Load a lengthy list of packages that are used in the Book

**Usage**

```
require.RbcBook1()
.RbcBook1.pkgs()
```

**Details**

`require.RbcBook1` can be used to see at once whether all packages required for the book can be loaded.

To do for `.RbcBook1.pkgs`: version numbers?

**Value**

`.RbcBook1.pkgs` returns a character vector with the names of all packages used in the book.

**Author(s)**

Wolfgang Huber <huber@ebi.ac.uk>

**Examples**

```
## Not run:
library(reposTools)
install.packages2(.RbcBook1.pkgs(), develOK=TRUE)
require.RbcBook1()

## End(Not run)
```

---

rpart2gNEL

*rpart to graphNEL Converter*


---

**Description**

Creates an object of class `graphNEL` from one of class `rpart`.

**Usage**

```
rpart2gNEL(tr, remap=function(x) x, nsep="\n")
```

**Arguments**

|                    |                                                                          |
|--------------------|--------------------------------------------------------------------------|
| <code>tr</code>    | instance of <code>rpart</code> class from <a href="#">rpart</a> package. |
| <code>remap</code> | function that maps node names in tree to node names in graph.            |
| <code>nsep</code>  | token that separates node name and vote tally in the rendering.          |

**Details**

Not explicitly used in the book, but the basis for a certain graph illustrating application of `rpart`. The newer `coin`/`party` packages have more interesting rendering approaches based on `graphviz`.

`grabSplitV` is a utility function for acquiring the variable names, `remapAff` will remap affymetrix probe names to gene symbols for use in the `remap` parameter.

**Value**

a [graphNEL-class](#) instance with nodes constructed to convey information on the data tree fit

**Author(s)**

Vince Carey <stvjc@channing.harvard.edu>

**See Also**

[rpart](#), [graphNEL-class](#)

**Examples**

```
library(rpart)
example(rpart)
ff = rpart2gNEL(fit2)
ff
nodes(ff)
```



# Index

- \* **data**
  - bcr.cor, [2](#)
  - perf, [6](#)
- \* **misc**
  - bcStangle, [2](#)
  - bcSweave, [3](#)
  - checkingBookSources, [4](#)
  - imageSize, [5](#)
  - require.RbcBook1, [6](#)
- \* **models**
  - rpart2gNEL, [7](#)
  - .RbcBook1Files (checkingBookSources), [4](#)
- ALLperformance (perf), [6](#)
- bcr.cor, [2](#)
- bcr.euc (bcr.cor), [2](#)
- bcr.kldist (bcr.cor), [2](#)
- bcr.man (bcr.cor), [2](#)
- bcr.mi (bcr.cor), [2](#)
- bcr.spear (bcr.cor), [2](#)
- bcr.tau (bcr.cor), [2](#)
- bcStangle, [2](#)
- bcSweave, [3](#)
- checkingBookSources, [4](#)
- checkPackage (checkingBookSources), [4](#)
- checkRnw (checkingBookSources), [4](#)
- checkVerbatim (checkingBookSources), [4](#)
- dist, [2](#)
- grabSplitV (rpart2gNEL), [7](#)
- imageSize, [5](#)
- kidpackperformance (perf), [6](#)
- perf, [6](#)
- performance (perf), [6](#)
- remapAff (rpart2gNEL), [7](#)
- require.RbcBook1, [6](#)
- rpart, [7](#), [8](#)
- rpart2gNEL, [7](#)
- Survperformance (perf), [6](#)
- tangleToSingleFiles (bcStangle), [2](#)