

# Package ‘smarter’

February 26, 2025

**Title** A Collection of Modified R Functions to Make Basic Coding More Convenient

**Version** 1.0.1

**Date** 2025-02-25

**Description** A collection of recycled and modified R functions to aid in file manipulation, data exploration, wrangling, optimization, and object manipulation. Other functions aid in convenient data visualization, loop progression, software packaging, and installation.

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**Imports** devtools, RCurl, Rcpp, gplots, grDevices, stats, usethis, rmarkdown

**LinkingTo** Rcpp, RcppArmadillo

**License** GPL (>= 3)

**NeedsCompilation** yes

**Author** Paul Little [aut, cre]

**Maintainer** Paul Little <plittle321@gmail.com>

**Repository** CRAN

**Date/Publication** 2025-02-26 05:10:02 UTC

## Contents

bin_cont_var . . . . .	2
calc_JK . . . . .	3
chkInst_PACK . . . . .	3
chk_threads . . . . .	4
collapse_var . . . . .	4
logSumExp . . . . .	5
make_dummy . . . . .	5
make_menu . . . . .	6
name_change . . . . .	6
print_latex_table . . . . .	7

smart_boxplot . . . . .	8
smart_colors . . . . .	8
smart_compMATs . . . . .	9
smart_df . . . . .	10
smart_digits . . . . .	10
smart_dots . . . . .	11
smart_heatmap . . . . .	11
smart_hist . . . . .	12
smart_merge . . . . .	13
smart_mkdir . . . . .	13
smart_names . . . . .	14
smart_pack_versions . . . . .	14
smart_progress . . . . .	15
smart_reqNames . . . . .	15
smart_rmcols . . . . .	16
smart_RT . . . . .	16
smart_SN . . . . .	17
smart_solve . . . . .	17
smart_table . . . . .	18
smart_WT . . . . .	18

**Index** **19**

---

<code>bin_cont_var</code>	<i>bin_cont_var</i>
---------------------------	---------------------

---

**Description**

Transform numeric vector into discrete bins

**Usage**

`bin_cont_var(VAR, NUM_GROUPS, ROUND = 3, binNUM = FALSE)`

**Arguments**

VAR	A numeric vector of values to bin
NUM_GROUPS	A positive integer for the number of bins
ROUND	A nonnegative integer for displaying bin labels through binned intervals
binNUM	Boolean set to TRUE to map bins to numbers. Otherwise, bins are characterized by intervals

**Value**

A character or integer vector of collapsed/binned values

---

calc_JK	<i>calc_JK</i>
---------	----------------

---

**Description**

calc\_JK

**Usage**

calc\_JK(EST, L00\_EST, alpha = 0.05)

**Arguments**

EST	A numeric vector of parameter estimates
L00_EST	A numeric matrix of parameter estimates where columns correspond to each parameter and rows correspond to each leave one out estimate
alpha	A numeric value for constructing $(1 - \alpha) * 100\%$ confidence intervals

**Value**

A list of numeric jackknife summary mean and confidence intervals

---

chkInst_PACK	<i>chkInst_PACK</i>
--------------	---------------------

---

**Description**

Check package is installed

**Usage**

chkInst\_PACK(PACK)

**Arguments**

PACK	A character string for a package name
------	---------------------------------------

**Value**

Boolean for TRUE if package installed and FALSE if package is not installed or located

---

chk_threads	<i>chk_threads</i>
-------------	--------------------

---

**Description**

chk\_threads

**Usage**

chk\_threads(NN, ncores)

**Arguments**

NN	A positive integer for total iterations to loop over
ncores	A positive integer for number of threads

**Value**

An integer for number of threads.

---

collapse_var	<i>collapse_var</i>
--------------	---------------------

---

**Description**

Collapse a subset of values within a vector into a new value

**Usage**

collapse\_var(ORIG\_VAR, ORIG\_VALUES, NEW\_VALUE)

**Arguments**

ORIG_VAR	The input vector
ORIG_VALUES	A subset of values from the input vector to be collapsed
NEW_VALUE	The new value to replace ORIG_VALUES in ORIG_VAR

**Value**

A character or numeric vector

---

`logSumExp`*logSumExp*

---

**Description**

Calculates the  $\log(\text{sum}(\exp(x)))$  in Rcpp

**Usage**

```
logSumExp(x)
```

**Arguments**

`x` A numeric vector

**Value**

A numeric vector

---

`make_dummy`*make\_dummy*

---

**Description**

Construct a dummy-coded matrix for a single variable

**Usage**

```
make_dummy(x)
```

**Arguments**

`x` A numeric or character vector to convert to a dummy matrix

**Value**

A binary indicator matrix of ones and zeros

---

 make\_menu

*make\_menu*


---

### Description

Constructs an interactive menu for the user

### Usage

```
make_menu(PROMPT, OPTS, INDENT = "  ")
```

### Arguments

PROMPT	A character string prompt to the user
OPTS	A character vector where elements contain a number, then a closing parentheses, then the option value
INDENT	A character string for the amount of indentation from the left margin

### Value

Character string of user's response

---

 name\_change

*name\_change*


---

### Description

Substitute a column name of a matrix or data.frame with a new name

### Usage

```
name_change(DATA, ORIG_NAME, NEW_NAME)
```

### Arguments

DATA	A matrix or data.frame
ORIG_NAME	A single character column name to alter
NEW_NAME	A single character to replace ORIG_NAME

### Value

An updated data.frame with renamed fields

---

print\_latex\_table      *print\_latex\_table*

---

## Description

print\_latex\_table

## Usage

```
print_latex_table(
  DATA,
  repeat_VARS = NULL,
  my_align = NULL,
  add_table = FALSE,
  fontsize = NULL,
  caption = NULL,
  label = NULL,
  midrule1 = NULL,
  latex_comment = NULL,
  ...
)
```

## Arguments

DATA	A matrix or data.frame to present as a latex table
repeat_VARS	A string vector of colnames to avoid repeating values within a column
my_align	A string containing letters "l", "r", or "c" for left, right, and center alignment
add_table	Boolean set to TRUE to enclose tabular environment with table environment
fontsize	Defaults to NULL to not specify a fontsize. Otherwise, possible values are "tiny", "footnotesize", "small", "normalsize", "large", "Large", "LARGE", "huge", "Huge"
caption	A string to include a table caption
label	A string to represent a latex table label
midrule1	Default is set to NULL
latex_comment	Add a latex comment above the table for notes
...	arguments passed to cat

## Value

No return value

---

smart_boxplot	<i>smart_boxplot</i>
---------------	----------------------

---

**Description**

smart\_boxplot

**Usage**

smart\_boxplot(MAT, mar\_down = 8, srt = 45, ...)

**Arguments**

MAT	A numeric matrix of columns to plot as boxplots
mar_down	A positive numeric value to allow space below the x-axis for labels
srt	A numeric value to control the angle of x-axis labels
...	arguments passed to boxplot

**Value**

No return value.

---

smart_colors	<i>smart_colors</i>
--------------	---------------------

---

**Description**

smart\_colors

**Usage**

smart\_colors(nn, alpha = 1, overwrite = FALSE)

**Arguments**

nn	A positive integer greater than or equal to 2
alpha	A positive numeric value less than or equal to one
overwrite	Boolean If nn = 2, setting to FALSE will force colors to be white or black

**Value**

No return value.



---

smart_compMATs	<i>smart_compMATs</i>
----------------	-----------------------

---

**Description**

smart\_compMATs

**Usage**

```
smart_compMATs(
  MAT1,
  MAT2 = NULL,
  which_range = NULL,
  xlab,
  ylab,
  show_corr = TRUE,
  show_plot = FALSE,
  main = NULL,
  vec_col = NULL,
  ...
)
```

**Arguments**

MAT1	A numeric matrix
MAT2	A second numeric matrix of columns comparable to MAT1. Default is set to NULL resulting in histograms plotted for columns of MAT1
which_range	Default is set to NULL to calculate data ranges. Otherwise if set to "01", will enforce minimum 0 and maximum 1. If set to a numeric vector of two elements, will enforce the range.
xlab	A string for x-axis label
ylab	A string for y-axis label
show_corr	Boolean set to TRUE to print Pearson and Spearman correlations
show_plot	Boolean set to TRUE to plot comparison of two matrices
main	A string for the plot title
vec_col	A vector of colors to color scatter plot points
...	arguments passed to plot

**Value**

No return value.

---

smart_df	<i>smart_df</i>
----------	-----------------

---

**Description**

Construct `data.frame` with `data.frame()` but sets `stringsAsFactors` to `FALSE`.

**Usage**

```
smart_df(...)
```

**Arguments**

... arguments passed to `data.frame`

**Value**

A `data.frame`

---

smart_digits	<i>smart_digits</i>
--------------	---------------------

---

**Description**

Round numeric values to specific

**Usage**

```
smart_digits(x, digits = 2)
```

**Arguments**

`x` A numeric vector formatted to have consistently rounded values  
`digits` A positive integer to regulate the number of digits to round to

**Value**

Character version of rounded numeric value

---

smart_dots	<i>smart_dots</i>
------------	-------------------

---

**Description**

Prints a series of dots every few seconds

**Usage**

```
smart_dots(wait = 300, num_dots = 30)
```

**Arguments**

wait	A number of seconds to wait before printing "."
num_dots	The number of dots to print before printing a message

**Value**

No return value.

---

smart_heatmap	<i>smart_heatmap</i>
---------------	----------------------

---

**Description**

smart\_heatmap

**Usage**

```
smart_heatmap(  
  MAT = NULL,  
  DIST = FALSE,  
  main = "",  
  width = NULL,  
  height = NULL,  
  GRID = NULL,  
  clustRC = c(TRUE, TRUE),  
  nodePar_col = NULL,  
  nodePar_row = NULL,  
  mar = 2,  
  cex.main = 2,  
  rowData = NULL,  
  colData = NULL,  
  make_key = TRUE,  
  vec_cols = NULL  
)
```

**Arguments**

MAT	A numeric matrix of values
DIST	Boolean set to TRUE to treat MAT as distance matrix. Otherwise, function can perform row/column clustering
main	A string for the overall heatmap title
width	NULL
height	NULL
GRID	NULL
clustRC	NULL
nodePar_col	NULL
nodePar_row	NULL
mar	NULL
cex.main	NULL
rowData	NULL
colData	NULL
make_key	NULL
vec_cols	NULL

**Value**

No return value.

---

smart_hist	<i>smart_hist</i>
------------	-------------------

---

**Description**

smart\_hist

**Usage**

```
smart_hist(x, freq = FALSE, dens = TRUE, main = "", ...)
```

**Arguments**

x	A numeric vector
freq	Boolean set to FALSE to plot density on y-axis. Otherwise TRUE to plot frequencies
dens	Boolean set to TRUE to overlay kernel density
main	String for plot title
...	arguments passed to hist

**Value**

No return value.

---

smart_merge	<i>smart_merge</i>
-------------	--------------------

---

**Description**

Merges two data.frames assuming they have at least one shared column name

**Usage**

```
smart_merge(x, y, mess = FALSE, ...)
```

**Arguments**

x	A data.frame
y	A data.frame
mess	Default to FALSE. Otherwise a message is printed.
...	arguments passed to merge

**Value**

A merged data.frame

**Examples**

```
aa = smart_df(a = c(1,2,3),b = c("a","b","c"),c = c(4,5,6))
bb = smart_df(a = c(2,4,5),b = c("b","d","e"),d = c("alpha","beta","gamma"))
smart_merge(aa,bb,all.x = TRUE)
smart_merge(aa,bb,all.y = TRUE)
smart_merge(aa,bb,all = TRUE)
```

---

smart_mkdir	<i>smart_mkdir</i>
-------------	--------------------

---

**Description**

Create directory if it does not exist

**Usage**

```
smart_mkdir(input_dir)
```

**Arguments**

input_dir	A full path name for a directory to create
-----------	--

**Value**

No return value

---

smart_names	<i>smart_names</i>
-------------	--------------------

---

**Description**

Sets row/column names to matrix or data.frame

**Usage**

```
smart_names(MAT, ROW = NULL, COL = NULL)
```

**Arguments**

MAT	A matrix
ROW	A vector of length equal to nrow(MAT)
COL	A vector of length equal to ncol(MAT)

**Value**

Outputs a matrix or data.frame depending on input object class

---

smart_pack_versions	<i>smart_pack_versions</i>
---------------------	----------------------------

---

**Description**

Return all associated package versions

**Usage**

```
smart_pack_versions(pack, repo = "CRAN")
```

**Arguments**

pack	A string for the package name
repo	A string that takes values "CRAN", "aCRAN", and "cCRAN" for combining options "aCRAN" and "cCRAN". "cCRAN" refers to contributed packages. "aCRAN" refers to archived packages.

**Value**

A data.frame of available R packages

---

smart_progress	<i>smart_progress</i>
----------------	-----------------------

---

**Description**

Print progress of a for loop

**Usage**

```
smart_progress(ii, nn, string = ".", iter = 5, iter2 = 200, ...)
```

**Arguments**

ii	A positive integer to track a loop's progress
nn	A positive integer for the total number of loop iterations
string	A string to print
iter	A positive integer for how many multiple iterations to print "."
iter2	A positive integer to end a line of printed "." and track the loop's progress
...	arguments passed to cat

**Value**

No return value.

---

smart_reqNames	<i>smart_reqNames</i>
----------------	-----------------------

---

**Description**

Checks if required column names are contained in the matrix or data.frame.

**Usage**

```
smart_reqNames(DATA, REQ)
```

**Arguments**

DATA	A matrix or data.frame
REQ	A string vector of colnames required to be contained in DATA

**Value**

No return value.

---

smart_rmcols	<i>smart_rmcols</i>
--------------	---------------------

---

**Description**

Drops columns from a matrix or data.frame.

**Usage**

```
smart_rmcols(OBJ, rm_names)
```

**Arguments**

OBJ	A matrix or data.frame
rm_names	A string vector of colnames to remove

**Value**

A matrix or data.frame

---

smart_RT	<i>smart_RT</i>
----------	-----------------

---

**Description**

Calls `read.table()` but sets argument `stringsAsFactors = FALSE` to prevent treating character columns as factors.

**Usage**

```
smart_RT(...)
```

**Arguments**

...	arguments passed to <code>read.table</code>
-----	---

**Value**

Return is identical to `read.table()`



---

`smart_SN`*smart\_SN*

---

**Description**

Convert numeric values into scientific notation

**Usage**

```
smart_SN(x, digits = 2)
```

**Arguments**

`x`                    A numeric vector to convert to scientific notation  
`digits`                A positive integer for number of digits to include in notation

**Value**

A character vector

---

`smart_solve`*smart\_solve*

---

**Description**

`smart_solve`

**Usage**

```
smart_solve(mm)
```

**Arguments**

`mm`                    A square numeric matrix

**Value**

A square numeric matrix.

smart\_table

*smart\_table*

---

**Description**

Should elements passed into `table()` contain NA or NaN, we want to see them by default.

**Usage**

```
smart_table(...)
```

**Arguments**

... arguments passed to `table`

**Value**

Return a table

**Examples**

```
aa = c(1,1,2,2,2,3,NA)
table(aa)
smart_table(aa)
```

---

smart\_WT

*smart\_WF*

---

**Description**

Calls `write.table()` setting parameters `row.names` and `quote` to FALSE.

**Usage**

```
smart_WT(...)
```

**Arguments**

... arguments passed to `write.table`

**Value**

Return is identical to `write.table()`

# Index

`bin_cont_var`, 2

`calc_JK`, 3

`chk_threads`, 4

`chkInst_PACK`, 3

`collapse_var`, 4

`logSumExp`, 5

`make_dummy`, 5

`make_menu`, 6

`name_change`, 6

`print_latex_table`, 7

`smart_boxplot`, 8

`smart_colors`, 8

`smart_compMATs`, 9

`smart_df`, 10

`smart_digits`, 10

`smart_dots`, 11

`smart_heatmap`, 11

`smart_hist`, 12

`smart_merge`, 13

`smart_mkdir`, 13

`smart_names`, 14

`smart_pack_versions`, 14

`smart_progress`, 15

`smart_reqNames`, 15

`smart_rmcols`, 16

`smart_RT`, 16

`smart_SN`, 17

`smart_solve`, 17

`smart_table`, 18

`smart_WT`, 18