Package ‘tfse’

February 11, 2019

Version 0.5.0

Title Tools for Script Editing

Description A collection of useful tools for programming and writing-scripts. Several functions are simple wrappers around base R functions that extend their functionality while also providing some convenient properties—regular expression functions that automatically detect look-ahead and look-behind statements, a read-line function that suppresses incomplete-final-line warnings and automatically opens and closes connections, a version of substrings that starts from the end of strings, etc. Other functions are useful for checking whether packages are installed, omitting missing data, and showing in-use connections.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Language en-US

Depends R (>= 2.10)

Imports utils, magrittr, dapr, stats

Suggests testthat, covr

URL https://tfse.mikewk.com

BugReports https://github.com/mkearney/tfse/issues

NeedsCompilation no

Author Michael W. Kearney [aut, cre] (<https://orcid.org/0000-0002-0730-4694>)

Maintainer Michael W. Kearney <kearneymw@missouri.edu>

Repository CRAN

Date/Publication 2019-02-11 08:30:03 UTC
### R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>.Renviron</td>
<td>3</td>
</tr>
<tr>
<td>add_arg_if</td>
<td>3</td>
</tr>
<tr>
<td>apa_citation</td>
<td>4</td>
</tr>
<tr>
<td>boxcode</td>
<td>5</td>
</tr>
<tr>
<td>cc</td>
<td>5</td>
</tr>
<tr>
<td>copy_function</td>
<td>6</td>
</tr>
<tr>
<td>count_na</td>
<td>7</td>
</tr>
<tr>
<td>data_set</td>
<td>7</td>
</tr>
<tr>
<td>desc_get_var</td>
<td>8</td>
</tr>
<tr>
<td>file_edit</td>
<td>8</td>
</tr>
<tr>
<td>github_raw</td>
<td>9</td>
</tr>
<tr>
<td>gregexpr</td>
<td>9</td>
</tr>
<tr>
<td>home</td>
<td>10</td>
</tr>
<tr>
<td>match_arg</td>
<td>10</td>
</tr>
<tr>
<td>max_na</td>
<td>11</td>
</tr>
<tr>
<td>menuiline</td>
<td>11</td>
</tr>
<tr>
<td>min_var</td>
<td>12</td>
</tr>
<tr>
<td>na_omit</td>
<td>13</td>
</tr>
<tr>
<td>nin</td>
<td>14</td>
</tr>
<tr>
<td>n_uq</td>
<td>14</td>
</tr>
<tr>
<td>pasteCollapse</td>
<td>15</td>
</tr>
<tr>
<td>pbcopy</td>
<td>15</td>
</tr>
<tr>
<td>peel_lists</td>
<td>16</td>
</tr>
<tr>
<td>pmsg</td>
<td>16</td>
</tr>
<tr>
<td>print_as_col</td>
<td>17</td>
</tr>
<tr>
<td>print_complete</td>
<td>17</td>
</tr>
<tr>
<td>print_start</td>
<td>18</td>
</tr>
<tr>
<td>psub</td>
<td>18</td>
</tr>
<tr>
<td>readlines</td>
<td>19</td>
</tr>
<tr>
<td>readline</td>
<td>19</td>
</tr>
<tr>
<td>read_RDS</td>
<td>20</td>
</tr>
<tr>
<td>regmatches</td>
<td>20</td>
</tr>
<tr>
<td>rename_git_repo</td>
<td>21</td>
</tr>
<tr>
<td>rescale</td>
<td>21</td>
</tr>
<tr>
<td>rm_DDS_Store</td>
<td>22</td>
</tr>
<tr>
<td>save_RDS</td>
<td>23</td>
</tr>
<tr>
<td>search_files</td>
<td>23</td>
</tr>
<tr>
<td>search_function</td>
<td>24</td>
</tr>
<tr>
<td>search_files</td>
<td>24</td>
</tr>
<tr>
<td>search_these_files</td>
<td>25</td>
</tr>
<tr>
<td>set_class</td>
<td>25</td>
</tr>
<tr>
<td>set_names</td>
<td>25</td>
</tr>
<tr>
<td>set_renv</td>
<td>26</td>
</tr>
<tr>
<td>shhh</td>
<td>26</td>
</tr>
<tr>
<td>show_connections</td>
<td>27</td>
</tr>
<tr>
<td>substrev</td>
<td>27</td>
</tr>
<tr>
<td>tfse</td>
<td>27</td>
</tr>
</tbody>
</table>
.Renviron

this_in_that .......................................................... 28
trim_ws ............................................................... 28
unset_row_names .................................................. 29
un_zip ................................................................. 29
write_function ....................................................... 30
%P% ................................................................. 30

Index

<table>
<thead>
<tr>
<th>.Renviron</th>
<th>.Renviron file</th>
</tr>
</thead>
</table>

Description

Gets path to .Renviron file

Usage

.Renviron()

Details

Checks local directory first and then checks home directory

Value

Returns path to .Renviron file

add_arg_if  Add defaults to argument list

Description

Adds parameters to argument list if list does not already include those parameters

Usage

add_arg_if(args, ..., override = FALSE)

Arguments

args  Argument list
...
override Logical indicating whether to override existing values in args with the values provided as a named argument here.
Value

Argument list with updated values.

Examples

```r
## arg list
args <- list(x = 5, y = TRUE, z = FALSE)

## add arg defaults
add_arg_if(args, w = TRUE, z = TRUE)

## add arg defaults, overriding any previous values
add_arg_if(args, x = 10, z = TRUE, override = TRUE)
```

Description

Returns an APA-formatting citation of an R package

Usage

```r
apa_citation(pkg)
```

Arguments

- `pkg` Name of package (quoted string)

Value

A character vector of the APA citation. If on Mac or PC, then also it’s stored to the system’s clipboard.

Examples

```r
## cite this package
apa_citation("tfse")
```
Description
 Load clipboard with code chunk square. Paste to insert square into R script file at cursor location.

Usage
  boxcode(...)  
  codebox(...)  
  box_code(...)  
  code_box(...)

Arguments
  ... Name of section/block

Value
  Text for code box saved into clipboard. Paste to use at cursor.

cc 

Description
  Split strings by comma into character vector(s)

Usage
  cc(x, simplify)

Arguments
  x Vector of comma separated character strings
  simplify Logical indicating whether to return a character vector (the default) when the length of x is one. This argument does nothing if the length of x is greater than 1.

Value
  If length of x is 1 then a character vector otherwise a list of character vectors.
### Examples

```
## comma separated alphabet
abcs <- paste(letters, collapse = ",")

## split single string
cc(abcs)

## return as list
cc(abcs, simplify = FALSE)

## select columns
mtcars[, cc("cyl, mpg, wt, gear")]

## character vector with multiple strings
x <- c("v1, v2, v3", "y1, y2, y5")

## convert strings into list of [split] character vectors
cc(x)
```

---

### Description
Copy the function code to clipboard (ready to paste)

### Usage

```r
copy_function(....)
```

### Arguments

```
... Functions to write to clipboard. It’s best to name these.
```

### Value

```
Writes to clipboard (ready to paste)
```
**count_na**

*Count missing values*

**Description**

Returns counts of missing values

**Usage**

`count_na(x)`

**Arguments**

`x`  
Input data.

**Value**

Counts of missing observations

---

**data_set**

*Create data set*

**Description**

Simple way to make data frames

**Usage**

`data_set(...)`

**Arguments**

`...`  
Data to be converted to data frame

**Value**

A tibble data frame
### desc_get_var

**Description**
Get values from package DESCRIPTION

**Usage**

```r
desc_get_var(pkg, field = NULL)
desc_get_url(pkg)
desc_gh_repo(pkg)
```

**Arguments**

- `pkg` Character string, name of package
- `field` Character string, name of field/key

**Value**
Data frame

### file_edit

**Description**
Open file in current text editor

**Usage**

```r
file_edit(file)
```

**Arguments**

- `file` Name of file(s) to open.

**Details**
Should open using system default or current text editor.

**Value**
Opens file and returns invisible file name.
### github_raw

**Generate link to raw Github file**

**Description**

Converts Github path/repo/file information into a link to the raw version of the file.

**Usage**

```
github_raw(file, repo = NULL)
```

**Arguments**

- `file` Name of desired file; if `repo` is NULL, then this value should also provide repo information, i.e., owner and name of the repository—e.g., "owner/repo/file.ext". Alternatively, user may supply the URL to the file as it appears in a web browser, e.g., "https://github.com/mkearney/driver-snow/blob/master/theme/driver-snow.rstheme"

- `repo` Repository name information formatted as username/repo. If this information is provided in the value supplied to `file` then leave this as NULL (the default)

**Value**

Returns the URL path to the raw version of the file.

---

### gregexpr_

**smart gregexpr wrapper**

**Description**

smart gregexpr wrapper

**Usage**

```
gregexpr_(x, pat, NNN)
```

**Arguments**

- `x` Input text
- `pat` Reg ex pattern
- `NNN` Other args passed to base (g)regexpr

**Value**

Pattern match positions
### home

**Home directory**

**Description**

Gets user’s home directory

**Usage**

```r
description()  
```

**Details**

Looks for "HOME" environment variable and/or normalizes the tilde path

**Value**

Returns system/user’s default home directory

---

### match_arg

**match arg to choices**

**Description**

Wrapper around **match.arg** that defaults to ignoring case and trimming white space

**Usage**

```r
match_arg(arg, choices, multiple = FALSE, ignore_case = TRUE, trim_ws = TRUE)
```

**Arguments**

- **arg**: a character vector (of length one unless several.ok is TRUE) or NULL
- **choices**: a character vector of candidate values
- **multiple**: logical specifying if arg should be allowed to have more than one element. Defaults to FALSE
- **ignore_case**: logical indicating whether to ignore capitalization. Defaults to TRUE
- **trim_ws**: logical indicating whether to trim surrounding white space. Defaults to TRUE

**Value**

Value(s) matched via partial matching.
**max_na**

*Filter based on proportion of missing data*

**Description**

Returns columns that have less than or equal to a specified amount of missingness.

**Usage**

```
max_na(x, max = 0.05)
```

**Arguments**

- `x` Input data frame or matrix.
- `max` Maximum proportion of missingnesses allowed. Columns with higher proportions of missingness compared to this value will be dropped. Columns will only be returned if they have 1 - max proportion non-missing. This value must be between 0-1. It defaults to .05.

**Value**

Data frame or matrix with columns with less than or equal to the max allowed proportion of missingness.

**menuline**

*menuline*

**Description**

Creates interactive multiple choice question.

**Usage**

```
menuline(q, a)
```

**Arguments**

- `q` Question to be asked in interactive session.
- `a` Answer choices.

**Value**

Selection provided by user in interactive session.
### min_var

Select columns with minimum amount of variance

**Description**

Filters numeric columns by requiring a minimum amount of variance

**Usage**

```r
min_var(x, min = 1)
```

**Arguments**

- `x`: Input data, which should be either a data frame or matrix.
- `min`: Minimum amount of variance to require per column.

**Details**

This function omits missing values.

**Value**

Returns data frame (or matrix, depending on input class) with all non-numeric columns and only those numeric columns that meet the minimum amount of variance.

**Examples**

```r
# set seed (for replication purposes)
set.seed(206195)

# create data set
d <- data_set(
    w = rnorm(100, 0, 0.0),
    v = rnorm(100, 0, 0.5),
    x = rnorm(100, 0, 1.0),
    y = rnorm(100, 0, 2.0),
    z = rnorm(100, 0, 3.0)
)

# minimum var of 1.0 (default)
min_var(d)

# min variance of 0.1
min_var(d, 0.1)

# min var of 2.0
min_var(d, 2.0)
```
### Description

Returns data object with NA values (if atomic vector), all NA elements (if list), or all NA rows (if data frame or matrix) omitted.

### Usage

```r
na_omit(x)
```

### Arguments

- **x**: Data object

### Value

Data with NA values (if atomic vector), all NA elements (if list), or all NA rows (if data frame or matrix) omitted.

### Examples

```r
## generate data
df <- data.frame(
  a = I(list(c(1, 2), c(NA_integer_, NA_integer_), c(1, 2))),
  b = c("a", NA_character_, "c"),
  c = c(1.1243, NA_real_, -1.234134)
)

## data frame
na_omit(df)

## matrix
na_omit(as.matrix(df))

## list
na_omit(apply(df, 1, c))

## atomic vector
na_omit(df$b)
```
nin

Return lhs values not in rhs values

Description
Return lhs values not in rhs values

Usage

nin(lhs, rhs, value = TRUE)
yin(lhs, rhs, value = TRUE)

Arguments

lhs Values to check whether they are/not contained in the other
rhs Values to use as the reference
value Logical indicating whether to return the value or a logical vector

Examples

## a, b, zz in alphabet letters
yin(c("a", "b", "zz"), letters)

## a, b, zz NOT in alphabet letters
nin(c("a", "b", "zz"), letters)

n_uq

Number of unique elements

Description
Estimates number of unique elements in a data object

Usage

n_uq(x)

Arguments

x Input data

Value
Integer number of unique elements
### paste_collapse
**Paste collapse**

**Description**
Paste with sep and collapse set to empty.

**Usage**
paste_collapse(...)

**Arguments**
...

One or more character strings to paste together with paste0 and collapse equal to ""

**Value**
A single string collapsed and separated with empty spaces

### pbcopy
**pbcopy**

**Description**
Adds input to clipboard for pasting

**Usage**
pbcopy(x)

**Arguments**
x

Input passed to cat function.

**Value**
Prints x to clipboard.

**Examples**

```r
## Not run:
## alphabet as string
pbcopy(paste(letters, collapse = ""))
## paste e.g., C-v

## End(Not run)
```
peel_lists  

Description
peel lists

Usage
peel_lists(x)

Arguments
x  
Input data

Value
Peeled object

pmsg  

Paste collapse input and print as message

Description
Paste collapse input and print as message

Usage
pmsg(..., print = TRUE)

Arguments
...  
Strings to be paste0 collapse = ""

print  
Logical indicating whether to print the message (default) or return an unevaluated expression

Value
Either invisibly returns text of message or unevaluated expression
**print_as_col**  
*Print data frame as a single column*

**Description**
Prints first row of data frame

**Usage**
```
print_as_col(x)
```

**Arguments**
- `x`  Input data frame

**Value**
Prints first row and variable names

---

**print_complete**  
*Print message about completing a task*

**Description**
Prints a check-mark bulleted message presumably about task completion

**Usage**
```
print_complete(...)```

**Arguments**
- `...`  Strings collapsed (with no additional space added) into black message prefixed with a heavy-check emoji (color and emoji print may appear differently depending on your system/UI configuration)

**Value**
A printed message
**print_start**  
*Print message about starting a task*

**Description**
Prints a next-arrow bulleted message presumably about task completion

**Usage**

```plaintext
print_start(...)  
```

**Arguments**

- **...**  
  Strings collapsed (with no additional space added) into gray message prefixed with a next-arrow emoji (color and emoji print may appear differently depending on your system/UI configuration)

**Value**

A printed message

---

**psub**  
*Paste sub*

**Description**
Glue-like sub pasting of strings

**Usage**

```plaintext
psub(x, ...)  
```

**Arguments**

- **x**  
  Input string
- **...**  
  Named strings with names being the values to replace and the strings being the desired new value.
**readlines**

### Description
Read lines of file

### Usage

```r
c.readlines(x, ...)```

### Arguments

- `x`: Input
- `...`: Other args passed to `readLines`.

### Details
Simple wrapper around `readLines` that automates opening and closing of connection file.

### Value
Output

---

**readline**

### Description
Worry free way to read lines from interactive sessions.

### Usage

```r
c.readline(...)```

### Arguments

- `...`: Character string or vector to be used as prompt during interactive R session. Ultimately, this function only sends a single string to the user, but it will accept a vector if you’re picky about not creating strings of a certain width.

### Value
Input entered during interactive session without extra quotes.
**read_RDS**  
*Read RDS*

**Description**
Read serialized R data file

**Usage**
```r
read_RDS(path)
```

**Arguments**
- `path`: Name of .rds file

**Value**
A data object

**See Also**
Other readsave: `save_RDS`

---

**regmatches_**  
*extract reg expr matches*

**Description**
A wrapper around the base function combo of gregexpr and regmatches

**Usage**
```r
regmatches_(x, pat, drop = FALSE, ...)

regmatches_first(x, pat, drop = FALSE, ...)
```

**Arguments**
- `x`: Text data.
- `pat`: Reg ex pattern
- `drop`: Logical indicating whether to drop empty matches. Defaults to FALSE.
- `...`: Other args (like ignore.case) passed to gregexpr

**Value**
Matching expression from text.
rename_git_repo

rename_git_repo Rename git repo code

Description
Command line (bash) syntax for renaming a git repo

Usage
rename_git_repo(new_url = NULL)

Arguments
new_url Optional, URL pointing to correct (renamed) repo. If NULL (default), "new_url" is printed in brackets.

Value
text of git command to rename git repo

rescale Standard: Rescale values to a standard normal scale

Description
Standard: Rescale values to a standard normal scale
Normal: Rescale values to a standard (0-1) scale
Log: Rescale values to a natural log scale
Point-scale: Rescale values to a new point scale

Usage
rescale_standard(x, na_omit = TRUE)
rescale_normal(x, na_omit = TRUE)
rescale_log(x, na_omit = TRUE)
rescale_pointscale(x, lower, upper, lower0 = NULL, upper0 = NULL, na_omit = TRUE)
Arguments

- **x**: Input vector
- **na_omit**: Logical indicating whether to drop missing (NA) values. Default is TRUE.
- **lower**: Min value of new scale. Only applicable for pointscales.
- **upper**: Max value of new scale. Only applicable for pointscales.
- **lowerP**: Min value of old scale. If NULL, defaults to min of input. Only applicable for pointscales.
- **upperP**: Max value of old scale. If NULL, defaults to max of input. Only applicable for pointscales.

Value

- Rescaled vector

Examples

```r
## randomly sample 10 values ranging from -10 to 100
x <- sample(-10:100, 10)

## rescale to 0-1 scale
rescale_standard(x)

## rescale to normal distribution (z-scores)
rescale_normal(x)

## rescale to logged distribution (natural log)
rescale_log(x)

## rescale to new point scale
rescale_pointscale(x, 1, 7, lowerP = -10, upperP = 100)
```

---

**rm_DS_Store**

*Remove pesky .DS_Store files*

Description

- Recursively removes all .DS_Store files in working directory.

Usage

- `rm_DS_Store()`
**save_RDS**

**Save RDS**

**Description**

Save serialized R data file

**Usage**

`save_RDS(x, path, compress = FALSE)`

**Arguments**

- **x**: Data object to be saved
- **path**: Name of the file to be saved
- **compress**: Logical indicating whether to compress data—it will take up less space but it will be slower. Defaults to FALSE.

**Value**

Invisible data object

**See Also**

Other readsave: `read_RDS`

**search_files**

**search_files**

**Description**

Returns matching files and line numbers of given string pattern.

**Usage**

`search_files(x, path = ".", recursive = TRUE, all.files = FALSE)`
search_these_files

Arguments

- **x**  
  Pattern.
- **path**  
  Path on which to restrict search. Defaults to current working directory.
- **recursive**  
  logical
- **all.files**  
  default false excludes dot files

Value

Output from terminal - file name, line number, and preview of matching text

---

search_function

**search functions**

Description

Like search_files but for functions

Usage

`search_function(pat, fun)`

Arguments

- **pat**  
  Pattern to match
- **fun**  
  Function to inspect

---

search_these_files

**Search these files**

Description

Look for text in a group of files

Usage

`search_these_files(x, f)`

Arguments

- **x**  
  Regex pat
- **f**  
  Vector of files

Value

Prints matches
**set_class**

**Description**
Set class with a parenthetical function.
Add class with a parenthetical function.

**Usage**

```
set_class(x, value)
add_class(x, value)
```

**Arguments**

- `x` Object to assign new class to.
- `value` Class value to assign to `x`

**Value**

Object `x` as class value.

---

**set_names**

**Description**
Add names with a parenthetical function.

**Usage**

```
set_names(x, nms)
```

**Arguments**

- `x` Data object
- `nms` Names to assign object.
**set_renv**  
*Set R environment variable*

**Description**

Sets R environment variable and adds it to user’s home `.Renviron` file.

**Usage**

```r
set_renv(...)  
```

**Arguments**

...  
Named environment variables/values

**Value**

Appends environment variable entry to `~/.Renviron` file

---

**shhh**  
*execute expression quietly*

**Description**

execute expression quietly

**Usage**

```r
shhh(expr)  
```

**Arguments**

expr  
Expression to be evaluated without additional printing.

**Value**

Output from evaluated expression.
show_connections

Description
Displays active connections as a tidy tibble

Usage
show_connections()

Value
Prints and invisibly returns data frame.

substr

Description
Returns portion of string starting from end of string (otherwise just like substr)

Usage
substr(x, start, stop = 0)

Arguments

x Character vector
start Number of characters to include relative to the last character position.
stop Specify the number of characters from the final character to set as the last character position.

Value
Sub string with last i characters.

tfse
tfse: Various Useful Functions

Description
Collection of useful functions.

Author(s)
Michael W. Kearney
this_in_that  Where is this in that?

Description

Looks up (matches) the position of this in that (table)

Usage

this_in_that(this, that, value = NULL)

Arguments

this  Values to look up in that
that  Value positions matched to this
value  Optional, values to be returned rather than the default, which returns positions (integers)

trim_ws  trim_ws

Description

Returns character vector without extra spaces and trimmed of white space.

Usage

trim_ws(x)

Arguments

x  Character vector

Value

Character vector without extra spaces
**unset_row_names**

**Description**

Unset row names with a parenthetical function.

**Usage**

`unset_row_names(x)`

**Arguments**

- `x` Data object

---

**un_zip**

*Unzip files into directory*

**Description**

Unzip archive into similarly named directory

**Usage**

`un_zip(path)`

**Arguments**

- `path` Name of zipfile. Must end in ".zip"

**Value**

Creates a directory in the same folder with the same name (minus the zip part)
write_function

Write function to file

Description

Write the function code to a file and open the file

Usage

write_function(...)  

Arguments

...  
Functions to write to file. It’s best to name these.

Value

Writes to temporary file and opens that file.

%P%

Paste grapes

Description

Paste0 strings together with grapes (inverse/inside out function call)

Usage

lhs %P% rhs

Arguments

lhs  
Left hand side presumably character string

rhs  
Right hand side presumably another character string

Value

A pasted together (with no space) string(s)
Index

+Topic package
  tfse, 27
+Topic tfse-package
  tfse, 27
  .Renviron, 3
  %P%, 30

add_arg_if, 3
add_class(set_class), 25
apa_citation, 4

box_code(boxcode), 5
boxcode, 5
  cc, 5
code_box(boxcode), 5
codebox(boxcode), 5
copy_function, 6
count_na, 7
data_set, 7
desc_get_url(desc_get_var), 8
desc_get_var, 8
desc_gh_repo(desc_get_var), 8

file_edit, 8
github_raw, 9
gregexpr_, 9

home, 10
match_arg, 10
match_arg, 10
max_na, 11
menulist, 11
min_var, 12

n_uq, 14
na_omit, 13
nin, 14

pasteCollapse, 15
pbcopy, 15
peel_lists, 16
pmsg, 16
print_as_col, 17
print_complete, 17
print_start, 18
psub, 18

read_RDS, 20, 23
readline_, 19
readlines, 19
regexpr_(gregexpr_), 9
regmatches_, 20
regmatches_first(regmatches_), 20
rename_git_repo, 21
rescale, 21
rescale_log(rescale), 21
rescale_normal(rescale), 21
rescale_pointscale(rescale), 21
rescale_standard(rescale), 21
rm_.DS_Store, 22

save_RDS, 20, 23
search_files, 23
search_function, 24
search_these_files, 24
set_class, 25
set_names, 25
set_renv, 26
shhh, 26
show_connections, 27
substrrev, 27

tfse, 27
tfse-package(tfse), 27
this_in_that, 28
trim_ws, 28
un_zip, 29
unset_row_names, 29
write_function, 30
yin (nin), 14