Package ‘matlabr’

August 13, 2018

Type Package
Title An Interface for MATLAB using System Calls
Version 1.5.2
Date 2018-08-13
Maintainer John Muschelli <muschellij2@gmail.com>
Description Provides users to call MATLAB from using the ``system’’ command. Allows users to submit lines of code or MATLAB m files. This is in comparison to 'R.matlab', which creates a MATLAB server.

Imports stringr
License GPL-2
Encoding UTF-8
SystemRequirements MATLAB

BugReports https://github.com/muschellij2/matlabr/issues
RoxygenNote 6.1.0
Suggests covr
NeedsCompilation no
Author John Muschelli [aut, cre]
Repository CRAN
Date/Publication 2018-08-13 16:30:05 UTC

R topics documented:

add_path .................................................. 2
get_matlab ................................................ 2
have_matlab .............................................. 3
rm_at_to_matlab_mat ................................... 4
run_matlab_code ......................................... 4
run_matlab_script ....................................... 5
rvec_to_matlab .......................................... 6
rvec_to_matlabcell ..................................... 6
rvec_to_matlabclist ................................... 7
add_path

Create PATHs to add to MATLAB PATHs

Usage

add_path(path)

gen_path(path)

add_gen_path(path)

Arguments

path: path to add

Value

A character vector

Examples

add_path("-/")
gen_path("-/")
gen_path("-/")

get_matlab

Find matlab path

Description

This tries to find matlab’s path using a system which command, and then, if not found, looks at getOption("matlab.path"). If not path is found, it fails.

Usage

get_matlab(try_defaults = TRUE, desktop = FALSE, splash = FALSE,
   display = FALSE, wait = TRUE, single_thread = FALSE)
Arguments

- try_defaults: (logical) If `matlab` is not found from `Sys.which`, and `matlab.path` not found, then try some default PATHs for Linux and OS X.
- desktop: Should desktop be active for MATLAB?
- splash: Should splash be active for MATLAB?
- display: Should display be active for MATLAB?
- wait: Should R wait for the command to finish. Both passed to `system` and adds the `-wait` flag.
- single_thread: Should the flag `-singleCompThread` be executed to limit MATLAB to a single computational thread?

Value

Character of command for MATLAB

Examples

```r
if (have_matlab()) {
  get_matlab()
}
```

---

have_matlab  Logical check if MATLAB is accessible

Description

Uses `get_matlab` to check if MATLAB’s path accessible

Usage

```r
have_matlab()
```

Value

Logical TRUE is MATLAB is accessible, FALSE if not

Examples

```r
have_matlab()
```
**rmat_to_matlab_mat**  \hspace{1em} *Convert R matrix to matlab matrix*

**Description**

This function takes in an R matrix then turns it into a matrix in matlab.

**Usage**

```r
rmat_to_matlab_mat(x, matname = NULL, transpose = FALSE)
```

**Arguments**

- `x`: matrix of values
- `matname`: Object in matlab to be assigned
- `transpose`: Transpose the matrix

**Value**

Character scalar of matlab code

---

**run_matlab_code**  \hspace{1em} *Runs matlab code*

**Description**

This function takes in matlab code, where the last line must end with a `;` and returns the exit status.

**Usage**

```r
run_matlab_code(code, endlines = TRUE, verbose = TRUE, 
    add_clear_all = FALSE, paths_to_add = NULL, ...)
```

**Arguments**

- `code`: Character vector of code.
- `endlines`: Logical of whether the semicolon (`;`) should be pasted to each element of the vector.
- `verbose`: Print out filename to run
- `add_clear_all`: Add `clear all;` to the beginning of code
- `paths_to_add`: Character vector of PATHs to add to the script using `add_path`
- `...`: Options passed to `run_matlab_script`
run_matlab_script

Value

Exit status of matlab code

Examples

```c
if (have_matlab()){
    run_matlab_code(c("disp('The version of the matlab is:'), "disp(version)"),
        paths_to_add = "-/")
}
```

```c
CC not run:
if (have_matlab()){
    run_matlab_code("disp(version)"
    run_matlab_code("disp(version)", paths_to_add = "-/")
    run_matlab_code(c("x = 5", "disp(['The value of x is ', num2str(x)])")
}
```

## End(Not run)

---

run_matlab_script  Run matlab script

Description

This function runs a matlab script, and returns exit statuses

Usage

```c
run_matlab_script(fname, verbose = TRUE, desktop = FALSE,
    splash = FALSE, display = FALSE, wait = TRUE,
    single_thread = FALSE, ...)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>fname</td>
<td>Filename of matlab script (.m file)</td>
</tr>
<tr>
<td>verbose</td>
<td>print diagnostic messages</td>
</tr>
<tr>
<td>desktop</td>
<td>Should desktop be active for MATLAB?</td>
</tr>
<tr>
<td>splash</td>
<td>Should splash be active for MATLAB?</td>
</tr>
<tr>
<td>display</td>
<td>Should display be active for MATLAB?</td>
</tr>
<tr>
<td>wait</td>
<td>Should R wait for the command to finish. Both passed to <code>system</code> and adds the <code>-wait</code> flag.</td>
</tr>
<tr>
<td>single_thread</td>
<td>Should the flag <code>-singleCompThread</code> be executed to limit MATLAB to a single computational thread?</td>
</tr>
<tr>
<td>...</td>
<td>Options passed to <code>system</code></td>
</tr>
</tbody>
</table>

Value

Exit status of matlab code
rvec_to_matlab \hspace{1cm} \textit{Convert R vector to matlab cell mat}

\textbf{Description}

This function takes in an R numeric and returns a status

\textbf{Usage}

\texttt{rvec\_to\_matlab(x, row = FALSE, sep = NULL, matname = NULL)}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{x} \hspace{1cm} \text{Numeric vector of values}
  \item \texttt{row} \hspace{1cm} \text{Create row vector instead of column vector}
  \item \texttt{sep} \hspace{1cm} \text{separator to use to separate cells. Will override row argument}
  \item \texttt{matname} \hspace{1cm} \text{Object in matlab to be assigned}
\end{itemize}

\textbf{Value}

Character scalar of matlab code

rvec\_to\_matlabcell \hspace{1cm} \textit{Convert R vector to matlab cell}

\textbf{Description}

This function takes in an R vector then turns it into a cell

\textbf{Usage}

\texttt{rvec\_to\_matlabcell(x, sep = ";", matname = NULL, transpose = FALSE)}

\textbf{Arguments}

\begin{itemize}
  \item \texttt{x} \hspace{1cm} \text{Character vector of values}
  \item \texttt{sep} \hspace{1cm} \text{separator to use to separate values. Defaults to ";" argument}
  \item \texttt{matname} \hspace{1cm} \text{Object in matlab to be assigned}
  \item \texttt{transpose} \hspace{1cm} \text{Transpose the cell}
\end{itemize}

\textbf{Value}

Character scalar of matlab code
rvec_to_matlabclist

Convert R vector to matlab cell mat

Description
This function takes in an R vector then turns it into a cell list

Usage
rvec_to_matlabclist(x, matname = NULL)

Arguments
x Character vector of values
matname Object in matlab to be assigned

Value
Character scalar of matlab code
Index

add_gen_path (add_path), 2
add_path, 2, 4

gen_path (add_path), 2
get_matlab, 2, 3

have_matlab, 3

rmat_to_matlab_mat, 4
run_matlab_code, 4
run_matlab_script, 4, 5
rvec_to_matlab, 6
rvec_to_matlabcell, 6
rvec_to_matlabclist, 7

system, 3, 5