

Package ‘SmoothR’

June 22, 2024

Title A Simple R Project Manager

Version 0.0.0.9000

Description SmoothR is an R package specifically designed to streamline the setup and management of Slurm job environments. It creates a project-specific directory, housing its own SLURM configuration in a separate directory, thus facilitating the easy configuration of environment variables, session management, and integration with the Slurm job submission system. Essentially, SmoothR acts as an organizer for R projects within a remote compute cluster.

License GPL-3

Imports dplyr,
lubridate,
sessioninfo,
lintr

Suggests testthat

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

R topics documented:

addFat	2
checkDir	2
checkpoint	3
CopyRunSmoothRScript	3
InitNow	4
initSmoothR	4
los	4
QuitNow	5
ReadFromConfig	5
runExpression	6
runScript	6
safeExecute	7
saveNow	7
setAccount	8
setConfig	8
setEmail	9
setModules	9

updateConfig	10
updateJobTime	10
updateNumThreads	11
updateOutputDir	11
updatePartition	11
updateSuffix	12
Index	13

addFat	<i>Add or Update FAT Configuration</i>
--------	--

Description

Adds or updates the FAT setting in the configuration file. It validates and accepts only 'T', 'F', 't', 'f' as values. If no value is provided, the current configuration is printed.

Usage

```
addFat(fat_value = NULL)
```

Arguments

fat_value Optional character indicating the FAT setting ('T', 'F', 't', 'f'); default NULL.

checkDir	<i>Check and Adjust Working Directory Based on Configuration</i>
----------	--

Description

Verifies the presence of required temp files in the current directory and ensures the output directory matches the expected configuration. Adjusts the working directory if necessary, based on the configuration.

Usage

```
checkDir()
```

Value

None; function is used for its side effects (changing directories or stopping execution).

checkpoint	<i>Print Custom Checkpoint Message in SLURM Output</i>
------------	--

Description

Outputs a checkpoint message with a user-defined comment. Useful for tracking progress in non-interactive SLURM tasks.

Usage

```
checkpoint(comment)
```

Arguments

comment	A character string for the checkpoint comment.
---------	--

Value

Prints to the console; returns invisible NULL.

Examples

```
checkpoint("Data loaded successfully")
checkpoint("Model training started")
```

CopyRunSmoothRScript	<i>Copy runSmoothR.sh Script to a Specified Directory</i>
----------------------	---

Description

This function copies the 'runSmoothR.sh' script from the 'shell_helpers' folder of the package to a specified folder.

Usage

```
CopyRunSmoothRScript(dest_folder)
```

Arguments

dest_folder	The destination folder where the script will be copied.
-------------	---

InitNow

Initialize Session After Slurm Execution

Description

Loads a specified R session file. Aborts if run interactively or if the session file does not exist. Continues without loading if the file is missing.

Usage

```
InitNow(session_file_name = NULL)
```

Arguments

session_file_name

The name of the session file to load, defaults to the first command line argument with ".RData" appended.

Examples

```
InitNow()
```

initSmoothR

Activate runSmoothR.sh by Copying and Adding to \$PATH

Description

This function copies 'runSmoothR.sh' to a specified directory and adds the directory to the \$PATH variable by updating the appropriate shell configuration file, only if the path is not already present.

Usage

```
initSmoothR(force = F)
```

los

List Objects with Sizes (Shortened Name: los)

Description

This function 'los' lists all objects in the specified environment, sorted by their memory sizes in a human-readable format. It allows for quick identification of large objects that may be using significant memory.

Usage

```
los(envir = .GlobalEnv, order = "decreasing")
```

Arguments

<code>envir</code>	Environment to search for objects, defaults to global environment.
<code>order</code>	Descending or ascending order for size sorting. Default is 'decreasing'.

Value

Prints a table of object names and their formatted sizes. Does not return any value.

Examples

```
los()
los(envir = .GlobalEnv, order = "decreasing")
```

QuitNow	<i>Quit R Session when not Interactive</i>
---------	--

Description

Terminates the current R session non-interactively with a success status. Useful in scripts and batch processing. Does not save the session before exiting. Precede with `SaveNow()` to save data first.

Usage

```
QuitNow()
```

Examples

```
QuitNow()
```

ReadFromConfig	<i>Read Configuration Values from a File</i>
----------------	--

Description

Retrieves a specific configuration value by key from a file formatted as 'export KEY='VALUE''. Throws an error if the key is not found or if the file does not exist.

Usage

```
ReadFromConfig(key_to_check)
```

Arguments

<code>key_to_check</code>	The configuration key to retrieve.
<code>config_file</code>	Path to the configuration file, defaults to '~/.temp_shell_exports'.

Value

The value associated with the provided key.

runExpression	<i>Run an R expression and submit it to Slurm for execution.</i>
---------------	--

Description

This function runs an R expression and submits it to Slurm for execution.

Usage

```
runExpression(r_expression, save = FALSE)
```

Arguments

r_expression The R expression to run.

Value

NA

runScript	<i>Run a R script and submit it to Slurm for execution.</i>
-----------	---

Description

This function runs a R script and submits it to Slurm for execution.

Usage

```
runScript(script_name, wait = FALSE, lint = FALSE)
```

Arguments

script_name The name of the R script to run.
lint Whether to lint the script instead of running it.
output_file The name of the output file.

Value

NA

safeExecute

*Safely Evaluate Expressions with Optional Logging***Description**

Evaluates an expression while handling errors and warnings to prevent unexpected script termination. If enabled, captures and logs messages, warnings, and errors to a file, facilitating smooth operation in production or critical scripts requiring uninterrupted execution.

Usage

```
safeExecute(
  expr,
  logging = FALSE,
  log_file = "R_console_log_file.log",
  envir = parent.frame()
)
```

Arguments

expr	Expression to evaluate, which can range from a single command to a block of code.
logging	Boolean; enables logging of all runtime messages, warnings, and errors when set to TRUE.
log_file	String; specifies the file path for logging, effective only if 'logging' is TRUE.
envir	Evaluation environment, defaulting to the parent frame.

Value

Returns the result of the evaluated expression, or NULL in the event of an error or warning.

Examples

```
safeExecute({
  x <- rnorm(100)
  if (mean(x) > 0) "Positive" else "Negative"
}, logging = TRUE, log_file = "run_log.txt")
```

saveNow

*Save the Current R Session***Description**

This function saves the current R session into a .RData file, enabling the session to be resumed in future Slurm job executions. It also logs session details into a timestamped text file, aiding in reproducibility and debugging. The session file name and timestamp can be specified; if not, defaults are used.

Usage

saveNow(session_file_name = NULL)

Arguments

session_file_name string (optional); the filename for saving the session. If not provided, the first command line argument will be used with "Data" appended.

timestamp string (optional); the timestamp to append to the session info file. Defaults to the current datetime if not provided.

Examples

saveNow() # Save using default settings

setAccount	<i>Set COMPUTE_ACCOUNT Environment Variable</i>
------------	---

Description

Sets the COMPUTE_ACCOUNT environment variable to the specified account name and displays it.

Usage

setAccount(accountName)

Arguments

accountName Account name to set as COMPUTE_ACCOUNT.

Examples

setAccount("snic123456")

setConfig	<i>Set Environment Configuration Interactively</i>
-----------	--

Description

Interactively prompts for various configuration settings and writes them to a shell script in the user's home directory. The script, formatted for environment variable export, is intended for shell sourcing. Includes conversion of job time from minutes to D-H:M:S format.

Usage

setConfig(force = F)

Arguments

None Parameters are input interactively.

Value

No return value; outputs file path and contents to console.

Examples

```
setConfig() # Run this in an interactive R session
```

setEmail	<i>Set USER_E_MAIL Environment Variable</i>
----------	---

Description

Sets the USER_E_MAIL environment variable to the specified email and displays it.

Usage

```
setEmail(email)
```

Arguments

email	Email address to set as USER_E_MAIL.
-------	--------------------------------------

Examples

```
setEmail("user@example.com")
```

setModules	<i>Update or Create Module Configuration</i>
------------	--

Description

Updates or creates a module configuration file in a specified directory, overwriting the existing file if allowed. The function is designed for interactive R sessions only and prints the file path and content to the console.

Usage

```
setModules(modules = c("R/4.1.1", "R_packages/4.1.1"), force = TRUE)
```

Arguments

modules	A character vector of module names to be loaded (default is c("R/4.1.1", "R_packages/4.1.1")).
force	Logical; if TRUE, forcefully overwrite the existing module file if it exists (default is TRUE).

Value

No return value; outputs file path and contents to console.

Examples

```
setModules() # Default usage
setModules(modules = c("Python/3.8.5", "Java/1.8"), force = FALSE)
```

updateConfig	<i>Update Configuration Setting</i>
--------------	-------------------------------------

Description

This function updates a specific configuration key in the '.temp_shell_exports' file located in the user's home directory. If the key does not exist, it adds the key with the specified value.

Usage

```
updateConfig(key, value)
```

Arguments

- key The configuration key to update.
- value The new value for the key.

Value

Invisible NULL; updates are made in-place.

Examples

```
updateConfig("NUM_THREADS", "8")
```

updateJobTime	<i>Update Job Time</i>
---------------	------------------------

Description

Update Job Time

Usage

```
updateJobTime(JOB_TIME_MINUTES)
```

Arguments

- JOB_TIME_MINUTES
New job time to set in minutes.

updateNumThreads	<i>Update the Number of Threads</i>
------------------	-------------------------------------

Description

Update the Number of Threads

Usage

```
updateNumThreads(NUM_THREADS)
```

Arguments

NUM_THREADS	New number of threads to set.
-------------	-------------------------------

updateOutputDir	<i>Update Output Directory</i>
-----------------	--------------------------------

Description

Update Output Directory

Usage

```
updateOutputDir(OUTPUT_DIR)
```

Arguments

OUTPUT_DIR	New output directory to set.
------------	------------------------------

updatePartition	<i>Update Job Partition</i>
-----------------	-----------------------------

Description

Update Job Partition

Usage

```
updatePartition(PARTITION)
```

Arguments

PARTITION	New partition to set.
-----------	-----------------------

updateSuffix	<i>Update File Suffix</i>
--------------	---------------------------

Description

Update File Suffix

Usage

updateSuffix(SUFFIX)

Arguments

SUFFIX	New suffix for the files to set.
--------	----------------------------------

Index

addFat, [2](#)

checkDir, [2](#)
checkpoint, [3](#)
CopyRunSmoothRScript, [3](#)

InitNow, [4](#)
initSmoothR, [4](#)

los, [4](#)

QuitNow, [5](#)

ReadFromConfig, [5](#)
runExpression, [6](#)
runScript, [6](#)

safeExecute, [7](#)
saveNow, [7](#)
setAccount, [8](#)
setConfig, [8](#)
setEmail, [9](#)
setModules, [9](#)

updateConfig, [10](#)
updateJobTime, [10](#)
updateNumThreads, [11](#)
updateOutputDir, [11](#)
updatePartition, [11](#)
updateSuffix, [12](#)