

Package ‘RewardExecutionPackage’

January 11, 2022

Type Package

Title REWARD Execution Package

Version 0.1.0

Author James Gilbert

Maintainer Jamie Gilbert <jpg@azimov.co.uk>

Depends DatabaseConnector (≥ 5.0.0),
R (≥ 4.0.0)

Imports jsonlite,
ParallelLogger,
SqlRender,
yaml,
SelfControlledCohort (≥ 1.6.0),
keyring,
tools,
checkmate,
vroom,
dplyr

Suggests testthat,
withr,
Eunomia,
R.utils

Remotes OHDSI/SelfControlledCohort,
OHDSI/Eunomia

Description Study execution package for generating REWARD results.
Uses SelfControlledCohort package to generate effect estimates.
Handles uploading of results to reward system.

License Apache

Encoding UTF-8

LazyData true

RoxygenNote 7.1.2

R topics documented:

computeAtlasCohorts	2
computeSccResults	2

createCdmConfiguration	3
createCohorts	3
createCustomDrugEras	4
createOutcomeCohorts	4
execute	5
getUncomputedAtlasCohorts	5
importReferenceTables	6
loadCdmConfiguration	6
runScc	7
unzipAndVerify	7
validateCdmConfigFile	8

computeAtlasCohorts	<i>Compute Atlas cohorts</i>
---------------------	------------------------------

Description

Computes sql cohorts against the CDM

Usage

```
computeAtlasCohorts(connection, config, exposureCohorts = FALSE)
```

Arguments

connection	DatabaseConnector connection to cdm
config	cdmConfiguration object
exposureCohorts	Exposure or outcome based cohorts?

computeSccResults	<i>Get Zipped Scc Results</i>
-------------------	-------------------------------

Description

Get zip files for scc Partial reward execution with a subset of targets or outcomes. If both are null this will generate SCC results for all exposure and outcome pairs. This is only really useful if you're adding an cohort after the full result set has been generated.

Usage

```
computeSccResults(
  connection,
  config,
  analysisIds = NULL,
  outcomeCohortIds = NULL,
  targetCohortIds = NULL
)
```

Arguments

connection	DatabaseConnector connection
config	cdm config loaded with loadCdmConfig function
outcomeCohortIds	- vector of outcome cohort ids or NULL
targetCohortIds	- vector of exposure cohort ids or NULL
.generateCohortStats	- generate time on treatment and time to outcome stats or not

```
createCdmConfiguration
```

Create CDM configuration file

Description

Opens a file for editing that contains the default settings for a cdm

Usage

```
createCdmConfiguration(
  cdmConfigPath,
  keyring = NULL,
  overwrite = FALSE,
  testConnection = TRUE
)
```

Arguments

cdmConfigPath	path to cdm configuration file
keyring	keyring::keyring. For systems that support multiple keyrings, specify the name of the keyring to use here. If NULL, then the default keyring is used.
overwrite	Overwrite existing file (if it exists)
testConnection	Attempt to connect to database and write to schemas needed for writing?

```
createCohorts
```

Create exposure cohorts

Description

Create all the exposure cohorts on a CDM Note, will not recompute if they already exist

Usage

```
createCohorts(connection, config, deleteExisting = FALSE)
```

Arguments

connection	DatabaseConnector connection to cdm
config	cdm config
deleteExisting	delete any existing computed cohorts

createCustomDrugEras	<i>Create custom drug eras</i>
----------------------	--------------------------------

Description

create the custom drug eras, these are for drugs with nonstandard eras (e.g. where doeses aren't picked up by repeat perscriptions). Could be something like a vaccine where exposed time is non trivial. deprecated It is now best to use atlas cohort definitions, will be removed in future version

Usage

```
createCustomDrugEras(connection, config)
```

Arguments

connection	DatabaseConnector connection
config	CdmConfig object

createOutcomeCohorts	<i>Create outcome cohorts</i>
----------------------	-------------------------------

Description

Create outcome cohorts in the CDM - this function can take a very very long time Does not compute if results already exist in cohort tables

Usage

```
createOutcomeCohorts(connection, config, deleteExisting = FALSE)
```

Arguments

connection	DatabaseConnector connection to cdm
config	cdm configuration
deleteExisting	remove existing data

execute	<i>Execute package</i>
---------	------------------------

Description

Upload cohorts and references Execute SCC methods Store results (and optionally transfer them to main REWARD server location)

Usage

```
execute(cdmConfigPath, referenceZipFile, deleteExistingCohorts = FALSE)
```

Arguments

cdmConfigPath	path to cdm configuration file
referenceZipFile	Path to rewardb cohort references zip file
deleteExistingCohorts	If results for cohorts already exist, delete and start again?

getUncomputedAtlasCohorts	<i>get Uncomputed Atlas Cohorts</i>
---------------------------	-------------------------------------

Description

Get cohorts that haven't been computed and return their references from file on disk SQL and JSON references are not stored in the CDM database's scratch schema

Usage

```
getUncomputedAtlasCohorts(connection, config, exposureCohorts = FALSE)
```

Arguments

connection	connection
config	cdmConfig
exposureCohorts	get exposures or not

importReferenceTables	<i>Import reference tables</i>
-----------------------	--------------------------------

Description

Note that this always overwrites the existing reference tables stored in the database

Usage

```
importReferenceTables(connection, cdmConfig, zipFilePath)
```

Arguments

connection	DatabaseConnector connection
cdmConfig	cdmConfig object
zipFilePath	zip file path

loadCdmConfiguration	<i>load cdm config object</i>
----------------------	-------------------------------

Description

Loads config and prompt user for db password Password can be set in environment variable passwordEnvironmentVariable of yaml file

Usage

```
loadCdmConfiguration(cdmConfigPath, keyring = NULL)
```

Arguments

cdmConfigPath	path to cdm configuration file
keyring	keyring::keyring. For systems that support multiple keyrings, specify the name of the keyring to use here. If NULL, then the default keyring is used.

runScc	<i>Peform SCC from self controlled cohort package with rewardbs settings</i>
--------	--

Description

Peform SCC from self controlled cohort package with rewardbs settings

Usage

```
runScc(  
  config,  
  postProcessFunction,  
  postProcessArgs,  
  analysisSettings,  
  exposureIds = NULL,  
  outcomeIds = NULL,  
  cores = parallel::detectCores() - 1  
)
```

unzipAndVerify	<i>Unzip and verify results zip with meta-data json</i>
----------------	---

Description

Used to unzip and check all files in a zip folder with meta data file containing md5 hashes at time of creation Used by both results generation and reference files

Usage

```
unzipAndVerify(exportZipFilePath, unzipPath, overwrite)
```

Arguments

exportZipFilePath	zip file to inflate
unzipPath	path to create
overwrite	overwrite any existing files

`validateCdmConfigFile` *Validate a cdm configuration file*

Description

Opens a file for editing that contains the default settings for a cdm

Usage

```
validateCdmConfigFile(cdmConfigPath, testConnection = TRUE, keyring = NULL)
```

Arguments

<code>cdmConfigPath</code>	path to cdm configuration file
<code>testConnection</code>	Attempt to connect to database and write to schemas needed for writing?
<code>keyring</code>	keyring::keyring. For systems that support multiple keyrings, specify the name of the keyring to use here. If NULL, then the default keyring is used.