Package 'CohortAlgebra'

July 14, 2022

Type Package

Title Cohort Algebra to create new cohort(s) from existing cohorts

Version 0.2.0

Date 2022-07-14

Maintainer Gowtham Rao <rao@ohdsi.org>

Description An R package that creates new cohort(s) from previously instantiated cohorts.

Depends DatabaseConnector (>= 5.0.0),
  R (>= 4.1.0)

Imports checkmate,
  dplyr,
  ParallelLogger,
  rlang

Suggests remotes,
  rmarkdown,
  knitr,
  testthat,
  withr

Remotes ohdsi/ParallelLogger

License Apache License

RoxygenNote 7.2.0

VignetteBuilder knitr

Roxygen list(markdown = TRUE)

Encoding UTF-8

Language en-US

URL https://ohdsi.github.io/CohortAlgebra/, https://github.com/OHDSI/CohortAlgebra

BugReports https://github.com/OHDSI/CohortAlgebra/issues

R topics documented:

Index   **9**

---

copyCohortsToTempTable

Get cohort ids in table

Description

Get cohort ids in table. This function is not exported.

Usage

```
copyCohortsToTempTable(
  connection = NULL,
  oldToNewCohortId,
  sourceCohortDatabaseSchema = NULL,
  sourceCohortTable,
  targetCohortTable = "#cohort_rows",
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema")
)
```

Arguments

connection   An object of type connection as created using the [connect](#) function in the
                     DatabaseConnector package. Can be left NULL if connectionDetails is pro-
                     vided, in which case a new connection will be opened at the start of the function,
                     and closed when the function finishes.

oldToNewCohortId
                     A data.frame object with two columns. oldCohortId and newCohortId. Both
                     should be integers. The oldCohortId are the cohorts that are the input cohorts
                     that need to be transformed. The newCohortId are the cohortIds of the corre-
                     sponding output after transformation. If the oldCohortId = newCohortId then
                     the data corresponding to oldCohortId will be replaced by the data from the
                     newCohortId.

sourceCohortDatabaseSchema
                     The database schema of the source cohort table.

sourceCohortTable
                     The name of the source cohort table.

targetCohortTable
                     A temp table to copy the cohorts from the source table.

tempEmulationSchema
                     Some database platforms like Oracle and Impala do not truly support temp ta-
                     bles. To emulate temp tables, provide a schema with write privileges where
                     temp tables can be created.

---

deleteCohortRecords   *Delete cohort records.*

Description

Delete all records from cohort table with the given cohort id. Edit privileges to the cohort table is
required.

Usage

```
deleteCohortRecords(
  connectionDetails = NULL,
  connection = NULL,
  cohortDatabaseSchema,
  cohortTable = "cohort",
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema"),
  cohortIds
)
```

Arguments

connectionDetails

An object of type connectionDetails as created using the createConnectionDetails
function in the DatabaseConnector package. Can be left NULL if connection
is provided.

connection   An object of type connection as created using the connect function in the
DatabaseConnector package. Can be left NULL if connectionDetails is pro-
vided, in which case a new connection will be opened at the start of the function,
and closed when the function finishes.

cohortDatabaseSchema

Schema name where your cohort tables reside. Note that for SQL Server, this
should include both the database and schema name, for example 'scratch.dbo'.

cohortTable   The name of the cohort table.

tempEmulationSchema

Some database platforms like Oracle and Impala do not truly support temp ta-
bles. To emulate temp tables, provide a schema with write privileges where
temp tables can be created.

cohortIds   A vector of one or more Cohort Ids.

---

eraFyCohorts   *Era-fy cohort(s)*

Description

Given a table with cohort_definition_id, subject_id, cohort_start_date, cohort_end_date execute era
logic. This will delete and replace the original rows with the cohort_definition_id(s). edit privileges
to the cohort table is required.

Usage

```
eraFyCohorts(
  connectionDetails = NULL,
  connection = NULL,
  cohortDatabaseSchema = NULL,
  cohortTable = "cohort",
  oldToNewCohortId,
  eraconstructorpad = 0,
  cdmDatabaseSchema = NULL,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema"),
  purgeConflicts = FALSE
)
```

Arguments

connectionDetails

An object of type connectionDetails as created using the [createConnectionDetails](#) function in the DatabaseConnector package. Can be left NULL if connection is provided.

connection   An object of type connection as created using the [connect](#) function in the DatabaseConnector package. Can be left NULL if connectionDetails is provided, in which case a new connection will be opened at the start of the function, and closed when the function finishes.

cohortDatabaseSchema

Schema name where your cohort tables reside. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.

cohortTable   The name of the cohort table.

oldToNewCohortId

A data.frame object with two columns. oldCohortId and newCohortId. Both should be integers. The oldCohortId are the cohorts that are the input cohorts that need to be transformed. The newCohortId are the cohortIds of the corresponding output after transformation. If the oldCohortId = newCohortId then the data corresponding to oldCohortId will be replaced by the data from the newCohortId.

eraconstructorpad

Optional value to pad cohort era construction logic. Default = 0. i.e. no padding.

cdmDatabaseSchema

Schema name where your patient-level data in OMOP CDM format resides. Note that for SQL Server, this should include both the database and schema name, for example 'cdm_data.dbo'. cdmDataschema is required when eraConstructorPad is > 0. eraConstructorPad is optional.

tempEmulationSchema

Some database platforms like Oracle and Impala do not truly support temp tables. To emulate temp tables, provide a schema with write privileges where temp tables can be created.

purgeConflicts   If there are conflicts in the target cohort table i.e. the target cohort table already has records with newCohortId, do you want to purge and replace them with transformed. By default - it will not be replaced, and an error message is thrown.

Value

NULL s

getCohortIdsInCohortTable

Get cohort ids in table

Description

Get cohort ids in table

Usage

```
getCohortIdsInCohortTable(
  connection = NULL,
  cohortDatabaseSchema = NULL,
  cohortTable,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema")
)
```

Arguments

connection	An object of type `connection` as created using the [connect](#) function in the DatabaseConnector package. Can be left NULL if connectionDetails is provided, in which case a new connection will be opened at the start of the function, and closed when the function finishes.
cohortDatabaseSchema	
	Schema name where your cohort tables reside. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortTable	The name of the cohort table.
tempEmulationSchema	
	Some database platforms like Oracle and Impala do not truly support temp tables. To emulate temp tables, provide a schema with write privileges where temp tables can be created.

intersectCohorts   *Intersect cohort(s)*

Description

Find the common cohort period for persons present in all the cohorts. Note: if subject is not found in any of the cohorts, then they will not be in the final cohort.

Usage

```
intersectCohorts(
  connectionDetails = NULL,
  connection = NULL,
  cohortDatabaseSchema = NULL,
  cohortTable = "cohort",
  cohortIds,
  newCohortId,
  purgeConflicts = FALSE,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema")
)
```

Arguments

connectionDetails
: An object of type connectionDetails as created using the [createConnectionDetails](#) function in the DatabaseConnector package. Can be left NULL if connection is provided.

connection
: An object of type connection as created using the [connect](#) function in the DatabaseConnector package. Can be left NULL if connectionDetails is provided, in which case a new connection will be opened at the start of the function, and closed when the function finishes.

cohortDatabaseSchema
: Schema name where your cohort tables reside. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.

cohortTable
: The name of the cohort table.

cohortIds
: A vector of one or more Cohort Ids.

newCohortId
: The cohort id of the result cohort.

purgeConflicts
: If there are conflicts in the target cohort table i.e. the target cohort table already has records with newCohortId, do you want to purge and replace them with transformed. By default - it will not be replaced, and an error message is thrown.

tempEmulationSchema
: Some database platforms like Oracle and Impala do not truly support temp tables. To emulate temp tables, provide a schema with write privileges where temp tables can be created.

Value

NULL

intersectCohorts(connection = conneciton, cohortDatabaseSchema = "study1234", cohortTable = "cohort", cohortIds = c(1, 2, 3), newCohortId = 9, purgeConflicts = TRUE)

---

minusCohorts   *Minus cohort(s)*

Description

Given two cohorts, substract (minus) the dates from the first cohort, the dates the subject also had on the second cohort.

Usage

```
minusCohorts(
  connectionDetails = NULL,
  connection = NULL,
  cohortDatabaseSchema = NULL,
  cohortTable = "cohort",
  firstCohortId,
  secondCohortId,
  newCohortId,
  purgeConflicts = FALSE,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema")
)
```

Arguments

connectionDetails

> An object of type connectionDetails as created using the [createConnectionDetails](#) function in the DatabaseConnector package. Can be left NULL if connection is provided.

connection

> An object of type connection as created using the [connect](#) function in the DatabaseConnector package. Can be left NULL if connectionDetails is provided, in which case a new connection will be opened at the start of the function, and closed when the function finishes.

cohortDatabaseSchema

> Schema name where your cohort tables reside. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.

cohortTable

> The name of the cohort table.

firstCohortId

> The cohort id of the cohort from which to substract.

secondCohortId

> The cohort id of the cohort that is used to substract.

newCohortId

> The cohort id of the result cohort.

purgeConflicts

> If there are conflicts in the target cohort table i.e. the target cohort table already has records with newCohortId, do you want to purge and replace them with transformed. By default - it will not be replaced, and an error message is thrown.

tempEmulationSchema

> Some database platforms like Oracle and Impala do not truly support temp tables. To emulate temp tables, provide a schema with write privileges where temp tables can be created.

Examples

```
minusCohorts(
 connection = conneciton,
 cohortDatabaseSchema = "study1234",
 cohortTable = "cohort",
 firstCohortId = 1,
 secondCohortId = 2,
 newCohortId = 9,
 purgeConflicts = TRUE
)
```

---

unionCohorts   *Union cohort(s)*

Description

Given a specified array of cohortIds in a cohort table, perform cohort union operator to create new cohorts.

Usage

```
unionCohorts(
  connectionDetails = NULL,
  connection = NULL,
  cohortDatabaseSchema = NULL,
  cohortTable = "cohort",
  oldToNewCohortId,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema"),
  purgeConflicts = FALSE
)
```

Arguments

connectionDetails

> An object of type connectionDetails as created using the [createConnectionDetails](#)
> function in the DatabaseConnector package. Can be left NULL if connection
> is provided.

connection   An object of type connection as created using the [connect](#) function in the
              DatabaseConnector package. Can be left NULL if connectionDetails is pro-
              vided, in which case a new connection will be opened at the start of the function,
              and closed when the function finishes.

cohortDatabaseSchema

> Schema name where your cohort tables reside. Note that for SQL Server, this
> should include both the database and schema name, for example 'scratch.dbo'.

cohortTable   The name of the cohort table.

oldToNewCohortId

> A data.frame object with two columns. oldCohortId and newCohortId. Both
> should be integers. The oldCohortId are the cohorts that are the input cohorts
> that need to be transformed. The newCohortId are the cohortIds of the corre-
> sponding output after transformation. If the oldCohortId = newCohortId then
> the data corresponding to oldCohortId will be replaced by the data from the
> newCohortId.

tempEmulationSchema

> Some database platforms like Oracle and Impala do not truly support temp ta-
> bles. To emulate temp tables, provide a schema with write privileges where
> temp tables can be created.

purgeConflicts If there are conflicts in the target cohort table i.e. the target cohort table already
                has records with newCohortId, do you want to purge and replace them with
                transformed. By default - it will not be replaced, and an error message is thrown.

Examples

```
unionCohorts(
 connection = conneciton,
 cohortDatabaseSchema = "study1234",
 cohortTable = "cohort",
 oldToNewCohortId = dplyr::tibble(
   oldCohortId = c(1, 2, 3),
   newCohortId = c(9, 9, 9)
 ),
 purgeConflicts = TRUE
)
```

Index