Package 'CohortAlgebra'

February 23, 2025

Type Package

Title Use of Interval Algebra to Create New Cohort(s) from Existing
Cohorts

Version 0.3.1

Date 2025-02-22

Maintainer Gowtham Rao <rao@ohdsi.org>

Description This software tool is designed to generate new cohorts utilizing data from
previously instantiated cohorts. It employs interval algebra operators such as UNION,
INTERSECT, and MINUS to manipulate the data within the instantiated cohorts and
create new cohorts.

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

Imports checkmate, dplyr, lifecycle, rlang, SqlRender

Suggests knitr, rmarkdown, testthat, withr

License Apache License

RoxygenNote 7.3.2

VignetteBuilder knitr

Encoding UTF-8

Language en-US

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

https://CRAN.R-project.org/package=CohortAlgebra

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

NeedsCompilation no

Author Gowtham Rao [aut, cre],
Observational Health Data Science and Informatics [cph]

Repository CRAN

Date/Publication 2025-02-23 15:10:01 UTC

Contents

---

appendCohortTables   *Append cohort data from multiple cohort tables(s)*

Description

Append cohort data from multiple cohort tables.

[Stable]

Usage

```
appendCohortTables(
  connectionDetails = NULL,
  connection = NULL,
  sourceTables,
  targetCohortDatabaseSchema = NULL,
  targetCohortTable,
  isTempTable = 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 pro-
                  vided, in which case a new connection will be opened at the start of the function,
                  and closed when the function finishes.

sourceTables   A data.frame object with the columns sourceCohortDatabaseSchema, source-
                  CohortTableName.

targetCohortDatabaseSchema

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

targetCohortTable

> The name of the target cohort table.

isTempTable   Is the output a temp table. If yes, a new temp table is created. This will required an active connection. Any old temp table is dropped and replaced.

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

Nothing is returned

---

copyCohorts   *Copy cohorts from one table to another*

Description

Copy cohorts from one table to another table. If the new cohort table has any cohort id that matches the cohort id being copied, an error will be displayed.

[Stable]

Usage

```
copyCohorts(
  connectionDetails = NULL,
  connection = NULL,
  oldToNewCohortId,
  sourceCohortDatabaseSchema = NULL,
  targetCohortDatabaseSchema = sourceCohortDatabaseSchema,
  sourceCohortTable,
  targetCohortTable,
  isTempTable = FALSE,
  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.

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.

sourceCohortDatabaseSchema

The database schema of the source cohort table.

targetCohortDatabaseSchema

The database schema of the source cohort table.

sourceCohortTable

The name of the source cohort table.

targetCohortTable

The name of the target cohort table.

isTempTable	Is the output a temp table. If yes, a new temp table is created. This will required an active connection. Any old temp table is dropped and replaced.

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

Nothing is returned

---

deleteCohort	*Delete cohort*

Description

Delete all records for a given set of cohorts from the cohort table. Edit privileges to the cohort table is required.

[**Stable**]

Usage

```
deleteCohort(
  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 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.

cohortIds   A vector of one or more Cohort Ids.

Value

Nothing is returned

---

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.

[Stable]

Usage

```
eraFyCohorts(
  connectionDetails = NULL,
  connection = NULL,
  sourceCohortDatabaseSchema = NULL,
  sourceCohortTable = "cohort",
  targetCohortDatabaseSchema = NULL,
  targetCohortTable,
  oldCohortIds,
  newCohortId,
  eraconstructorpad = 0,
  cdmDatabaseSchema = NULL,
  purgeConflicts = FALSE,
  isTempTable = 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.

sourceCohortDatabaseSchema

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

sourceCohortTable

          The name of the source cohort table.

targetCohortDatabaseSchema

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

targetCohortTable

          The name of the target cohort table.

oldCohortIds   An array of 1 or more integer id representing the cohort id of the cohort on which the function will be applied.

newCohortId   The cohort id of the output cohort.

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'.

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.
isTempTable	Is the output a temp table. If yes, a new temp table is created. This will required an active connection. Any old temp table is dropped and replaced.
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

Nothing is returned

---

getCohortIdsInCohortTable

Get cohort ids in table

Description

Get cohort ids in table

[Stable]

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.

Value

An array of integers called cohort id.

---

  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.

[Stable]

Usage

```
intersectCohorts(
  connectionDetails = NULL,
  connection = NULL,
  sourceCohortDatabaseSchema = NULL,
  sourceCohortTable,
  targetCohortDatabaseSchema = NULL,
  targetCohortTable,
  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 pro-
             vided, in which case a new connection will be opened at the start of the function,
             and closed when the function finishes.

sourceCohortDatabaseSchema

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

sourceCohortTable

             The name of the source cohort table.

targetCohortDatabaseSchema

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

targetCohortTable

        The name of the target cohort table.

cohortIds   A vector of one or more Cohort Ids.

newCohortId   The cohort id of the output 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

Nothing is returned

---

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.

[Stable]

Usage

```
minusCohorts(
  connectionDetails = NULL,
  connection = NULL,
  sourceCohortDatabaseSchema = NULL,
  sourceCohortTable = "cohort",
  targetCohortDatabaseSchema = sourceCohortDatabaseSchema,
  targetCohortTable = sourceCohortTable,
  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.

sourceCohortDatabaseSchema

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

sourceCohortTable

  The name of the source cohort table.

targetCohortDatabaseSchema

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

targetCohortTable

  The name of the target cohort table.

firstCohortId The cohort id of the cohort from which to subtract.

secondCohortId The cohort id of the cohort that is used to subtract.

newCohortId The cohort id of the output 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

Nothing is returned

---

reindexCohortsByDays *Reindex cohort(s) by relative days*

Description

reindexCohort changes the cohort_start_date and/or cohort_end_date of one or more source cohorts based on a set of reindexing rules. The output is a one or more valid target cohorts.

[Experimental]

Usage

```
reindexCohortsByDays(
  connectionDetails = NULL,
  connection = NULL,
  sourceCohortDatabaseSchema = NULL,
  sourceCohortTable = "cohort",
```

```
    sourceCohortIds,
    targetCohortDatabaseSchema = NULL,
    targetCohortTable,
    offsetStartAnchor = "cohort_start_date",
    offsetEndAnchor = "cohort_end_date",
    reindexRules,
    cdmDatabaseSchema = NULL,
    purgeConflicts = FALSE,
    isTempTable = FALSE,
    bulkLoad = Sys.getenv("DATABASE_CONNECTOR_BULK_UPLOAD"),
    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 pro-
> vided, in which case a new connection will be opened at the start of the function,
> and closed when the function finishes.

sourceCohortDatabaseSchema

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

sourceCohortTable

> The name of the source cohort table.

sourceCohortIds

> An array of one or more cohortIds in the source cohort table.

targetCohortDatabaseSchema

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

targetCohortTable

> The name of the target cohort table.

offsetStartAnchor

> Determines the anchor point for the start of the reindexing. It can be either
> cohort_start_date or cohort_end_date of sourceCohort.

offsetEndAnchor

> Determines the anchor point for the end of the reindexing. It can be either
> cohort_start_date or cohort_end_date of targetCohort.

reindexRules A data frame specifying the reindexing rules. It should contain the following
> columns: 'offsetId' a unique key for identifying the newly generated cohorts.
> Each offsetId corresponds to a specific reindex rule and will be used to create
> new cohort id in targetCohort. 'offsetStartValue' is an integer value indicating
> the number of days to 'offsetStartAnchor'. A positive values will extend, while
> negative values will shorten the start date from the 'offsetStartAnchor'. offse-
> tEndValue' An integer value indicating the number of days to offset the end date.

Positive values will extend, while negative values will shorten the end date from the 'offsetEndAnchor'.

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'.

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.

isTempTable   Is the output a temp table. If yes, a new temp table is created. This will required an active connection. Any old temp table is dropped and replaced.

bulkLoad   See 'insertTable' function in 'DatabaseConnector'.

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

If output is temp table, then the name of the temp table is returned.

---

removeOverlappingSubjects

Remove subjects in cohort that overlap with another cohort

Description

Remove subjects in cohort that overlap with another cohort. Given a Cohort A, check if the records of subjects in cohort A overlaps with records for the same subject in cohort B. If there is overlap then remove all records of that subject from Cohort A. Overlap is defined as b.cohort_end_date >= a.cohort_start_date AND b.cohort_start_date <= a.cohort_end_date. The overlap logic maybe offset by using a startDayOffSet (applied on cohort A's cohort_start_date) and endDayOffSet (applied on Cohort A's cohort_end_date). If while applying offset, the window becomes such that (a.cohort_start_date + startDayOffSet) > (a.cohort_end_date + endDayOffset) that record is ignored and thus deleted.

[Experimental]

Usage

```
removeOverlappingSubjects(
  connectionDetails = NULL,
  connection = NULL,
  cohortDatabaseSchema,
  cohortId,
  newCohortId,
  cohortsWithSubjectsToRemove,
```

```
    offsetCohortStartDate = -99999,
    offsetCohortEndDate = 99999,
    restrictSecondCohortStartBeforeFirstCohortStart = FALSE,
    restrictSecondCohortStartAfterFirstCohortStart = FALSE,
    cohortTable = "cohort",
    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'.

cohortId
The cohort id of the cohort whose subjects will be removed.

newCohortId
The cohort id of the output cohort.

cohortsWithSubjectsToRemove

An array of one or more cohorts with subjects to remove from given cohorts.

offsetCohortStartDate

(Default = 0) If you want to offset cohort start date, please provide a integer number.

offsetCohortEndDate

(Default = 0) If you want to offset cohort start date, please provide a integer number.

restrictSecondCohortStartBeforeFirstCohortStart

(Default = FALSE) If TRUE, then the secondCohort's cohort_start_date should be < firstCohort's cohort_start_date.

restrictSecondCohortStartAfterFirstCohortStart

(Default = FALSE) If TRUE, then the secondCohort's cohort_start_date should be > firstCohort's cohort_start_date.

cohortTable
The name of the cohort table.

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

Nothing is returned

---

unionCohorts   *Union cohort(s)*

Description

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

[Stable]

Usage

```
unionCohorts(
  connectionDetails = NULL,
  connection = NULL,
  sourceCohortDatabaseSchema = NULL,
  sourceCohortTable,
  targetCohortDatabaseSchema = NULL,
  targetCohortTable,
  oldToNewCohortId,
  isTempTable = FALSE,
  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.

sourceCohortDatabaseSchema

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

sourceCohortTable

The name of the source cohort table.

targetCohortDatabaseSchema

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

targetCohortTable

> The name of the target 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.

isTempTable Is the output a temp table. If yes, a new temp table is created. This will required an active connection. Any old temp table is dropped and replaced.

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

Nothing is returned

Index