Package 'AlendronateVsRaloxifene'

May 15, 2017

Type Package

Title Alendronate versus Raloxifene and the risk of Hip Fracture

Version 0.3.0

Author Martijn Schuemie [aut, cre],
Yeesuk Kim [aut],
Patrick Ryan [aut],
Marc Suchard [aut]

Maintainer Martijn J. Schuemie <schuemie@ohdsi.org>

Description More about what it does (maybe more than one line).

Depends DatabaseConnector (>= 1.10.0)

Imports SqlRender (>= 1.3.0),
RJDBC,
FeatureExtraction (>= 1.2.0),
CohortMethod (>= 2.2.2),
EmpiricalCalibration (>= 1.2.0),
OhdsiSharing (>= 0.1.1),
Cyclops (>= 1.2.2),
rmarkdown,
ggplot2,
ff,
ffbase

Suggests OhdsiRTools (>= 1.3.0)

License Apache License 2.0

LazyData TRUE

RoxygenNote 6.0.1

R topics documented:

`AlendronateVsRaloxifene` *AlendronateVsRaloxifene*

Description

AlendronateVsRaloxifene

`assessFeasibility` *Execute OHDSI Alendronate Vs Raloxifene study feasibility assessment*

Description

Execute OHDSI Alendronate Vs Raloxifene study feasibility assessment

Usage

```
assessFeasibility(connectionDetails, cdmDatabaseSchema,
  workDatabaseSchema = cdmDatabaseSchema,
  studyCohortTable = "ohdsi_alendronate_raloxifene",
  oracleTempSchema = workDatabaseSchema, outputFolder)
```

Arguments

`connectionDetails`
: An object of type `connectionDetails` as created using the `createConnectionDetails` function in the DatabaseConnector package.

`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'.

`workDatabaseSchema`
: Schema name where intermediate data can be stored. You will need to have write priviliges in this schema. Note that for SQL Server, this should include both the database and schema name, for example 'cdm_data.dbo'.

`studyCohortTable`
: The name of the table that will be created in the work database schema. This table will hold the exposure and outcome cohorts used in this study.

`oracleTempSchema`
: Should be used in Oracle to specify a schema where the user has write priviliges for storing temporary tables.

`outputFolder` Name of local folder to place results; make sure to use forward slashes (/). Do not use a folder on a network drive since this greatly impacts performance.

Details

This function executes the OHDSI Alendronate Vs Raloxifene study feasibility assessment.

Examples

```
## Not run:
connectionDetails <- createConnectionDetails(dbms = "postgresql",
                                             user = "joe",
                                             password = "secret",
                                             server = "myserver")

assessFeasibility(connectionDetails,
                  cdmDatabaseSchema = "cdm_data",
                  workDatabaseSchema = "results",
                  studyCohortTable = "ohdsi_alendronate_raloxifene",
                  oracleTempSchema = NULL,
                  outputFolder = "c:/temp/feasibility_results")

## End(Not run)
```

---

createAnalysesDetails *Create the analyses details*

Description

Create the analyses details

Usage

```
createAnalysesDetails(connectionDetails, cdmDatabaseSchema, workFolder)
```

Arguments

connectionDetails	An object of type connectionDetails as created using the createConnectionDetails function in the DatabaseConnector package.
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'.
workFolder	Name of local folder to place results; make sure to use forward slashes (/)

Details

This function creates files specifying the analyses that will be performed.

createCohorts *Create the exposure and outcome cohorts*

Description

Create the exposure and outcome cohorts

Usage

```
createCohorts(connectionDetails, cdmDatabaseSchema, workDatabaseSchema,
  studyCohortTable = "ohdsi_alendronate_raloxifene", oracleTempSchema,
  outputFolder)
```

Arguments

connectionDetails	An object of type connectionDetails as created using the createConnectionDetails function in the DatabaseConnector package.
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'.
workDatabaseSchema	Schema name where intermediate data can be stored. You will need to have write priviliges in this schema. Note that for SQL Server, this should include both the database and schema name, for example 'cdm_data.dbo'.
studyCohortTable	The name of the table that will be created in the work database schema. This table will hold the exposure and outcome cohorts used in this study.
oracleTempSchema	Should be used in Oracle to specify a schema where the user has write priviliges for storing temporary tables.
outputFolder	Name of local folder to place results; make sure to use forward slashes (/)

Details

This function will create the exposure and outcome cohorts following the definitions included in this package.

createMetaData *Create metadata file*

Description

Create metadata file

Usage

```
createMetaData(connectionDetails, cdmDatabaseSchema, exportFolder)
```

Arguments

connectionDetails	An object of type connectionDetails as created using the createConnectionDetails function in the DatabaseConnector package.
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'.
exportFolder	The name of the folder where the metadata file should be created.

Details

Creates a file containing metadata about the source data (taken from the cdm_source table) and R package versions.

---

createTableAndFigures *Create tables and figures*

Description

Create tables and figures

Usage

```
createTableAndFigures(exportFolder)
```

Arguments

exportFolder	The path to the export folder containing the results.

Details

Creates tables and figures for viewing and interpreting the results. Requires that the execute function has completed first.

---

execute *Execute OHDSI Keppra and the Risk of Angioedema study*

Description

Execute OHDSI Keppra and the Risk of Angioedema study

Usage

```
execute(connectionDetails, cdmDatabaseSchema,
  workDatabaseSchema = cdmDatabaseSchema,
  studyCohortTable = "ohdsi_alendronate_raloxifene",
  oracleTempSchema = workDatabaseSchema, outputFolder, createCohorts = TRUE,
  runAnalyses = TRUE, packageResults = TRUE, maxCores = 4)
```

Arguments

connectionDetails	An object of type connectionDetails as created using the createConnectionDetails function in the DatabaseConnector package.
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'.
workDatabaseSchema	Schema name where intermediate data can be stored. You will need to have write priviliges in this schema. Note that for SQL Server, this should include both the database and schema name, for example 'cdm_data.dbo'.
studyCohortTable	The name of the table that will be created in the work database schema. This table will hold the exposure and outcome cohorts used in this study.
oracleTempSchema	Should be used in Oracle to specify a schema where the user has write priviliges for storing temporary tables.
outputFolder	Name of local folder to place results; make sure to use forward slashes (/). Do not use a folder on a network drive since this greatly impacts performance.
createCohorts	Create the studyCohortTable table with the exposure and outcome cohorts?
runAnalyses	Perform the cohort method analyses?
packageResults	Package the results for sharing?
maxCores	How many parallel cores should be used? If more cores are made available this can speed up the analyses.

Details

This function executes the OHDSI Keppra and the Risk of Angioedema study.

Examples

```
## Not run:
connectionDetails <- createConnectionDetails(dbms = "postgresql",
                                             user = "joe",
                                             password = "secret",
                                             server = "myserver")

execute(connectionDetails,
        cdmDatabaseSchema = "cdm_data",
        workDatabaseSchema = "results",
        studyCohortTable = "ohdsi_alendronate_raloxifene",
        oracleTempSchema = NULL,
        outputFolder = "c:/temp/study_results",
        maxCores = 4)

## End(Not run)
```

packageResults *Package the results for sharing with OHDSI researchers*

Description

Package the results for sharing with OHDSI researchers

Usage

```
packageResults(connectionDetails, cdmDatabaseSchema, outputFolder,
  minCellCount = 5)
```

Arguments

connectionDetails	An object of type connectionDetails as created using the createConnectionDetails function in the DatabaseConnector package.
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'.
outputFolder	Name of local folder to place results; make sure to use forward slashes (/)
minCellCount	The minimum number of subjects contributing to a count before it can be included in the results.

Details

This function packages the results.

submitResults *Submit the study results to the study coordinating center*

Description

Submit the study results to the study coordinating center

Usage

```
submitResults(exportFolder, key, secret)
```

Arguments

exportFolder	The path to the folder containing the StudyResults.zip file.
key	The key string as provided by the study coordinator
secret	The secret string as provided by the study coordinator

Details

This will upload the file StudyResults.zip to the study coordinating center using Amazon S3. This requires an active internet connection.

Value

TRUE if the upload was successful.

`writeReport` *Write a report summarizing all the results for a single database*

Description

Write a report summarizing all the results for a single database

Usage

```
writeReport(exportFolder, outputFile)
```

Arguments

`exportFolder`	The path to the export folder containing the results.
`outputFile`	The name of the .docx file that will be created.

Details

Requires that the `createTableAndFigures` has been executed first.

Index