Package 'SkeletonComparativeEffectStudy'

August 24, 2018

Type Package

Title A Package Skeleton for Comparative Effectiveness Studies

Version 0.0.1

Author Martijn Schuemie [aut, cre]

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

Description
A skeleton package, to be used as a starting point when implementing comprative effect studies.

Depends DatabaseConnector

Imports SqlRender,
EmpiricalCalibration,
Cyclops,
FeatureExtraction,
CohortMethod,
ggplot2,
ff,
ffbase,
MethodEvaluation,
EvidenceSynthesis,
OhdsiSharing,
zip,
OhdsiRTools (>= 1.5.0)

Suggests knitr,
rmarkdown,
DT,
shiny

License Apache License 2.0

VignetteBuilder knitr

LazyData TRUE

RoxygenNote 6.1.0

R topics documented:

`createCohorts` *Create the exposure and outcome cohorts*

Description

Create the exposure and outcome cohorts

Usage

```
createCohorts(connectionDetails, cdmDatabaseSchema, cohortDatabaseSchema,
  cohortTable = "cohort", 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'.

`cohortDatabaseSchema`
: 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'.

`cohortTable`
: 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.

`createFiguresAndTables` *Generate diagnostics*

Description

Generate diagnostics

Usage

```
createFiguresAndTables(outputFolder, connectionDetails,
  cohortDatabaseSchema, cohortTable, oracleTempSchema = oracleTempSchema)
```

Arguments

`outputFolder`	Name of local folder where the results were generated; make sure to use forward slashes (/). Do not use a folder on a network drive since this greatly impacts performance.
`connectionDetails`	An object of type `connectionDetails` as created using the `createConnectionDetails` function in the DatabaseConnector package.
`cohortDatabaseSchema`	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'.
`cohortTable`	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.

Details

This function generates figures and tables.

`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.

---

execute *Execute the Study*

Description

Execute the Study

Usage

```
execute(connectionDetails, cdmDatabaseSchema,
  cohortDatabaseSchema = cdmDatabaseSchema, cohortTable = "cohort",
  oracleTempSchema = cohortDatabaseSchema, outputFolder,
  createCohorts = TRUE, synthesizePositiveControls = TRUE,
  runAnalyses = TRUE, runDiagnostics = TRUE, packageResults = TRUE,
  maxCores = 4, 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'.

cohortDatabaseSchema
: 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'.

cohortTable
: 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 cohortTable table with the exposure and outcome cohorts?
synthesizePositiveControls	Should positive controls be synthesized?
runAnalyses	Perform the cohort method analyses?
runDiagnostics	Compute study diagnostics?
packageResults	Should results be packaged for later sharing?
maxCores	How many parallel cores should be used? If more cores are made available this can speed up the analyses.
minCellCount	The minimum number of subjects contributing to a count before it can be included in packaged results.

Details

This function executes the SkeletonComparativeEffectStudy Study.

The createCohorts, synthesizePositiveControls, runAnalyses, and runDiagnostics arguments are intended to be used to run parts of the full study at a time, but none of the parts are considerd to be optional.

Examples

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

execute(connectionDetails,
        cdmDatabaseSchema = "cdm_data",
        cohortDatabaseSchema = "study_results",
        cohortTable = "cohort",
        oracleTempSchema = NULL,
        outputFolder = "c:/temp/study_results",
        maxCores = 4)

## End(Not run)
```

---

generateDiagnostics *Generate diagnostics*

Description

Generate diagnostics

Usage

```
generateDiagnostics(outputFolder)
```

Arguments

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

Details

This function generates analyses diagnostics. Requires the study to be executed first.

---

launchEvidenceExplorer *Launch the SqlRender Developer Shiny app*

Description

Launch the SqlRender Developer Shiny app

Usage

```
launchEvidenceExplorer(studyFolder, blind = TRUE,
  launch.browser = TRUE)
```

Arguments

studyFolder	The root folder containing the study results. The app expects each database to have a subfolder in this folder, containing the packaged results.
blind	Should the user be blinded to the main results?
launch.browser	Should the app be launched in your default browser, or in a Shiny window. Note: copying to clipboard will not work in a Shiny window.

Details

Launches a Shiny app that allows the user to explore the evidence

---

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.

---

prepareForEvidenceExplorer

Prepare results for the Evidence Explorer Shiny app.

Description

Prepare results for the Evidence Explorer Shiny app.

Usage

```
prepareForEvidenceExplorer(studyFolder)
```

Arguments

studyFolder	The root folder containing the study results. The app expects each database to have a subfolder in this folder, containing the packaged results.

SkeletonComparativeEffectStudy

SkeletonComparativeEffectStudy

Description

SkeletonComparativeEffectStudy

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

Description

Submit the study results to the study coordinating center

Usage

```
submitResults(outputFolder, key, secret)
```

Arguments

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

synthesizePositiveControls *Synthesize positive controls*

Description

Synthesize positive controls

Usage

```
synthesizePositiveControls(connectionDetails, cdmDatabaseSchema,
  cohortDatabaseSchema, cohortTable = "cohort", oracleTempSchema,
  outputFolder, maxCores = 1)
```

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'.
cohortDatabaseSchema	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'.
cohortTable	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 (/)
maxCores	How many parallel cores should be used? If more cores are made available this can speed up the analyses.

Details

This function will synthesize positve controls based on the negative controls. The simulated outcomes will be added to the cohort table.

Index