Package 'SkeletonComparativeEffectStudy'

June 2, 2021

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 comparative effect studies.

Depends R(>= 3.6.0),
DatabaseConnector (>= 4.0.2)

Imports SqlRender (>= 1.7.0),
EmpiricalCalibration (>= 2.1.0),
Cyclops (>= 3.1.1),
FeatureExtraction (>= 3.1.1),
CohortMethod (>= 4.1.0),
ggplot2,
Andromeda (>= 0.4.1),
dplyr,
tibble,
readr,
rlang,
MethodEvaluation (>= 2.1.0),
OhdsiSharing (>= 0.2.2),
ParallelLogger (>= 2.0.1),
survival,
plyr,
jsonlite

Suggests knitr,
rmarkdown,
DT,
shiny,
EvidenceSynthesis,
meta,
tidyr

Remotes ohdsi/FeatureExtraction,
ohdsi/CohortMethod,
ohdsi/MethodEvaluation,
ohdsi/OhdsiSharing

License Apache License 2.0

VignetteBuilder knitr

LazyData TRUE

RoxygenNote 7.1.1

R topics documented:

`execute` *Execute the Study*

Description

Execute the Study

Usage

```
execute(
  connectionDetails,
  cdmDatabaseSchema,
  cohortDatabaseSchema = cdmDatabaseSchema,
  cohortTable = "cohort",
  oracleTempSchema = NULL,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema"),
  verifyDependencies = TRUE,
  outputFolder,
  databaseId = "Unknown",
  databaseName = "Unknown",
  databaseDescription = "Unknown",
  createCohorts = TRUE,
  synthesizePositiveControls = TRUE,
  runAnalyses = 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`
: DEPRECATED: use 'tempEmulationSchema' instead.

`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.

`verifyDependencies`
: Check whether correct package versions are installed?

`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.

`databaseId`
: A short string for identifying the database (e.g. 'Synpuf').

`databaseName`
: The full name of the database (e.g. 'Medicare Claims Synthetic Public Use Files (SynPUFs)').

`databaseDescription`
: A short description (several sentences) of the database.

`createCohorts`
: Create the cohortTable table with the exposure and outcome cohorts?

`synthesizePositiveControls`
: Should positive controls be synthesized?

`runAnalyses`
: Perform the cohort method analyses?

`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 considered 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)
```

`exportResults` *Export all results to tables*

Description

Outputs all results to a folder called 'export', and zips them.

Usage

```
exportResults(
  outputFolder,
  databaseId,
  databaseName,
  databaseDescription,
  minCellCount = 5,
  maxCores
)
```

Arguments

`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.
`databaseId`	A short string for identifying the database (e.g. 'Synpuf').
`databaseName`	The full name of the database.
`databaseDescription`	A short description (several sentences) of the database.
`minCellCount`	The minimum cell count for fields contains person counts or fractions.
`maxCores`	How many parallel cores should be used? If more cores are made available this can speed up the analyses.

launchEvidenceExplorer *Launch the SqlRender Developer Shiny app*

Description

Launch the SqlRender Developer Shiny app

Usage

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

Arguments

dataFolder	A folder where the data files for the Evidence Explorer app will be stored. Use the prepareForEvidenceExplorer to populate this folder.
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

prepareForEvidenceExplorer *Prepare results for the Evidence Explorer Shiny app.*

Description

Prepare results for the Evidence Explorer Shiny app.

Usage

```
prepareForEvidenceExplorer(resultsZipFile, dataFolder)
```

Arguments

resultsZipFile	Path to a zip file containing results from a study executed by this package.
dataFolder	A folder where the data files for the Evidence Explorer app will be stored.

Examples

```
## Not run:
# Add results from three databases to the Shiny app data folder:
prepareForEvidenceExplorer("ResultsMDCD.zip", "/shinyData")
prepareForEvidenceExplorer("ResultsMDCR.zip", "/shinyData")
prepareForEvidenceExplorer("ResultsCCAE.zip", "/shinyData")

# Launch the Shiny app:
launchEvidenceExplorer("/shinyData")

## End(Not run)
```

---

synthesizeResults *Conducts a meta-analysis across PLE result sets*

Description

Conducts a meta-analysis across PLE result sets

Usage

```
synthesizeResults(
  allDbsFolder,
  maExportFolder = allDbsFolder,
  maxCores = 1,
  method = "BayesianNonNormal",
  resultsZipPattern = "^Results_.*\\.zip",
  addTraditional = TRUE
)
```

Arguments

`allDbsFolder`	Folder on the local file system containing the individual zip files across databases (i.e., sites)
`maExportFolder`	A local folder where the meta-analysis results will be written. If not specified, results will be written to same directory with all other results.
`maxCores`	Maximum number of CPU cores to be used when computing the meta-analyses.
`method`	The meta-analysis method to use. Possible values are "BayesianNonNormal" (Schumie et al.) or "DL" (DerSimonian-Laird).
`addTraditional`	Boolean indicating if traditional meta-analysis (i.e., "DL") results should be added to result (if `method` is "BayesianNonNormal").

Details

Conducts a meta-analysis across result sets generated from a population level effect (PLE) study package. Meta-analysis methodology is based on DerSimonian and Laird (1986) or Schuemie et al. (2021).

Value

Does not return a value, but creates a new zip file in the `maExportFolder` for the meta-analyses.

References

DerSimonian R, Laird N. Meta-analysis in clinical trials. Control Clin Trials. 1986 Sep;7(3):177-88. doi: 10.1016/0197-2456(86)90046-2

Schuemie M, Chen Y, Madigan D, Suchard M, Combining Cox Regressions Across a Heterogeneous Distributed Research Network Facing Small and Zero Counts. arXiv: 2101.01551, 2021

`uploadResults` *Upload results to OHDSI server*

Description

Upload results to OHDSI server

Usage

```
uploadResults(outputFolder, privateKeyFileName, userName)
```

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.
`privateKeyFileName`	A character string denoting the path to the RSA private key provided by the study coordinator.
`userName`	A character string containing the user name provided by the study coordinator.

Details

This function uploads the 'Results_<databaseId>.zip' to the OHDSI SFTP server. Before sending, you can inspect the zip file, wich contains (zipped) CSV files. You can send the zip file from a different computer than the one on which is was created.

Index