

Package ‘StudyDiagnostics’

January 8, 2020

Type Package

Title Diagnostics for OHDSI studies

Version 0.0.1

Date 2019-12-17

Description Diagnostics for studies that use the OMOP Common Data Model and the OHDSI tools.

Depends DatabaseConnector

Imports SqlRender,
ggplot2,
ROhdsiWebApi,
ParallelLogger,
readr,
tibble,
dplyr,
RJSONIO,
FeatureExtraction,
ff

Suggests testthat,
knitr,
rmarkdown

License Apache License

VignetteBuilder knitr

URL <https://ohdsi.github.io/StudyDiagnostics>, <https://github.com/OHDSI/StudyDiagnostics>

BugReports <https://github.com/OHDSI/StudyDiagnostics/issues>

RoxygenNote 6.1.1

Encoding UTF-8

R topics documented:

breakDownIndexEvents	2
compareCohortCharacteristics	3
computeCohortOverlap	4
createCohortTable	4
createConceptCountsTable	5
findCohortIncludedSourceConcepts	6

findCohortOrphanConcepts	7
findOrphanConcepts	9
getCohortCharacteristics	10
getIncidenceProportion	11
getInclusionStatistics	12
getInclusionStatisticsFromFiles	13
instantiateCohort	13
plotIncidenceProportion	15
plotIncidenceProportionByYear	16
runStudyDiagnostics	16
StudyDiagnostics	17

Index 18

breakDownIndexEvents	<i>Break down index events</i>
----------------------	--------------------------------

Description

For the concepts included in the index event definition, count how often they are encountered at the cohort index date.

Usage

```
breakDownIndexEvents(connectionDetails = NULL, connection = NULL,
  cdmDatabaseSchema, oracleTempSchema = NULL,
  cohortDatabaseSchema = cdmDatabaseSchema, cohortTable = "cohort",
  baseUrl = NULL, webApiCohortId = NULL, cohortJson = NULL,
  cohortSql = NULL, cohortId = cohortId)
```

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.
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'.
oracleTempSchema	Should be used in Oracle to specify a schema where the user has write privileges for storing temporary tables.
cohortDatabaseSchema	Schema name where your cohort table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortTable	Name of the cohort table.

baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI". Needn't be provided if cohortJson and cohortSql are provided.
webApiCohortId	The ID of the cohort in the WebAPI instance. Needn't be provided if cohortJson and cohortSql are provided.
cohortJson	A character string containing the JSON of a cohort definition. Needn't be provided if baseUrl and cohortId are provided.
cohortSql	The OHDSI SQL representation of the same cohort definition. Needn't be provided if baseUrl and cohortId are provided.
cohortId	The cohort definition ID used to reference the cohort in the cohort table.

Value

A data frame with concepts, and per concept the count of how often the concept was encountered at the index date.

compareCohortCharacteristics

Compare cohort characteristics

Description

Compare the characteristics of two cohorts, computing the standardized difference of the mean.

Usage

```
compareCohortCharacteristics(characteristics1, characteristics2)
```

Arguments

characteristics1

Characteristics of the first cohort, as created using the [getCohortCharacteristics](#) function.

characteristics2

Characteristics of the second cohort, as created using the [getCohortCharacteristics](#) function.

Value

A data frame comparing the characteristics of the two cohorts.

computeCohortOverlap	<i>Compute overlap between two cohorts</i>
----------------------	--

Description

Computes the overlap between a target and a comparator cohort.

Usage

```
computeCohortOverlap(connectionDetails = NULL, connection = NULL,
  cohortDatabaseSchema = cdmDatabaseSchema, cohortTable = "cohort",
  targetCohortId, comparatorCohortId)
```

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 table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortTable	Name of the cohort table.
targetCohortId	The cohort definition ID used to reference the target cohort in the cohort table.
comparatorCohortId	The cohort definition ID used to reference the comparator cohort in the cohort table.

Value

A data frame with overlap statistics.

createCohortTable	<i>Create cohort table(s)</i>
-------------------	-------------------------------

Description

This function creates an empty cohort table. Optionally, additional empty tables are created to store statistics on the various inclusion criteria.

Usage

```
createCohortTable(connectionDetails = NULL, connection = NULL,
  cohortDatabaseSchema, cohortTable = "cohort",
  createInclusionStatsTables = FALSE,
  resultsDatabaseSchema = cohortDatabaseSchema,
  cohortInclusionTable = paste0(cohortTable, "_inclusion"),
  cohortInclusionResultTable = paste0(cohortTable, "_inclusion_result"),
  cohortInclusionStatsTable = paste0(cohortTable, "_inclusion_stats"),
  cohortSummaryStatsTable = paste0(cohortTable, "_summary_stats"))
```

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 table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortTable	Name of the cohort table.
createInclusionStatsTables	Create the four additional tables for storing inclusion rule statistics?
resultsDatabaseSchema	Schema name where the statistics tables reside. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortInclusionTable	Name of the inclusion table, one of the tables for storing inclusion rule statistics.
cohortInclusionResultTable	Name of the inclusion result table, one of the tables for storing inclusion rule statistics.
cohortInclusionStatsTable	Name of the inclusion stats table, one of the tables for storing inclusion rule statistics.
cohortSummaryStatsTable	Name of the summary stats table, one of the tables for storing inclusion rule statistics.

createConceptCountsTable

Create concept counts table

Description

Create a table with counts of how often each concept ID occurs in the CDM.

Usage

```
createConceptCountsTable(connectionDetails = NULL, connection = NULL,
  cdmDatabaseSchema, conceptCountsDatabaseSchema = cdmDatabaseSchema,
  conceptCountsTable = "concept_counts")
```

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.
- 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'`.
- conceptCountsDatabaseSchema**
Schema name where your concept counts table resides. Note that for SQL Server, this should include both the database and schema name, for example `'scratch.dbo'`.
- conceptCountsTable**
Name of the concept counts table. This table can be created using the [createConceptCountsTable](#).

`findCohortIncludedSourceConcepts`

Check source codes used in a cohort definition

Description

This function first extracts all concept sets used in a cohort definition. Then, for each concept set the concept found in the CDM database the contributing source codes are identified.

Usage

```
findCohortIncludedSourceConcepts(connectionDetails = NULL,
  connection = NULL, cdmDatabaseSchema, oracleTempSchema = NULL,
  baseUrl = NULL, webApiCohortId = NULL, cohortJson = NULL,
  cohortSql = NULL, byMonth = FALSE, useSourceValues = 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.
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'.
oracleTempSchema	Should be used in Oracle to specify a schema where the user has write privileges for storing temporary tables.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI". Needn't be provided if cohortJson and cohortSql are provided.
webApiCohortId	The ID of the cohort in the WebAPI instance. Needn't be provided if cohortJson and cohortSql are provided.
cohortJson	A character string containing the JSON of a cohort definition. Needn't be provided if baseUrl and cohortId are provided.
cohortSql	The OHDSI SQL representation of the same cohort definition. Needn't be provided if baseUrl and cohortId are provided.
byMonth	Compute counts by month? If FALSE, only overall counts are computed.
useSourceValues	Use the source_value fields to find the codes used in the data? If not, this analysis will rely entirely on the source_concept_id fields instead. Note that, depending on the source data and ETL, it might be possible for the source_value fields to contain patient-identifiable information by accident.

Value

A data frame with source codes, with counts per domain how often the code was encountered in the CDM.

findCohortOrphanConcepts

Find orphan concepts for all concept sets in a cohort

Description

Searches for concepts that should belong to the concept sets in a cohort definition but don't, for example because of missing source-to-standard concept maps, or erroneous hierarchical relationships.

Usage

```
findCohortOrphanConcepts(connectionDetails = NULL, connection = NULL,
  cdmDatabaseSchema, oracleTempSchema = NULL, baseUrl = NULL,
  webApiCohortId = NULL, cohortJson = NULL,
  conceptCountsDatabaseSchema = cdmDatabaseSchema,
  conceptCountsTable = "concept_counts")
```

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.
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'.
oracleTempSchema	Should be used in Oracle to specify a schema where the user has write privileges for storing temporary tables.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI". Needn't be provided if cohortJson is provided.
webApiCohortId	The ID of the cohort in the WebAPI instance. Needn't be provided if cohortJson is provided.
cohortJson	A characteric string containing the JSON of a cohort definition. Needn't be provided if baseUrl and webApiCohortId are provided.
conceptCountsDatabaseSchema	Schema name where your concept counts table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
conceptCountsTable	Name of the concept counts table. This table can be created using the createConceptCountsTable .

Details

Logically, this function performs the following steps for each concept set expression in the cohort definition:

- Given the concept set expression, find all included concepts.
- Find all names of the input concepts, including synonyms, and the names of source concepts that map to them.
- Search for concepts (standard and source) that contain any of those names as substring.
- Filter those concepts to those that are not in the original set of concepts (i.e. orphans).
- Restrict the set of orphan concepts to those that appear in the CDM database and across network concept prevalence (as either source concept or standard concept).

Value

A data frame with orphan concepts, with counts how often the code was encountered in the CDM.

findOrphanConcepts	<i>Find (source) concepts that do not roll up to their ancestor(s)</i>
--------------------	--

Description

Searches for concepts that should belong to the set of concepts but don't, for example because of missing source-to-standard concept maps, or erroneous hierarchical relationships.

Usage

```
findOrphanConcepts(connectionDetails = NULL, connection = NULL,
  cdmDatabaseSchema, oracleTempSchema = NULL, conceptIds,
  conceptCountsDatabaseSchema = cdmDatabaseSchema,
  conceptCountsTable = "concept_counts")
```

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.
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'.
oracleTempSchema	Should be used in Oracle to specify a schema where the user has write privileges for storing temporary tables.
conceptIds	A vector of concept IDs for which we want to find orphans.
conceptCountsDatabaseSchema	Schema name where your concept counts table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
conceptCountsTable	Name of the concept counts table. This table can be created using the createConceptCountsTable .

Details

Logically, this function performs the following steps for the input set of concept IDs:

- Find all names of the input concepts, including synonyms, and the names of source concepts that map to them.
- Search for concepts (standard and source) that contain any of those names as substring.
- Filter those concepts to those that are not in the original set of concepts (i.e. orphans).
- Restrict the set of orphan concepts to those that appear in the CDM database and across network concept prevalence (as either source concept or standard concept).

Value

A data frame with orphan concepts, with counts how often the code was encountered in the CDM.

```
getCohortCharacteristics
```

Create characterization of a cohort

Description

Computes features using all drugs, conditions, procedures, etc. observed on or prior to the cohort index date.

Usage

```
getCohortCharacteristics(connectionDetails = NULL, connection = NULL,
  cdmDatabaseSchema, oracleTempSchema = NULL,
  cohortDatabaseSchema = cdmDatabaseSchema, cohortTable = "cohort",
  cohortId,
  covariateSettings = FeatureExtraction::createDefaultCovariateSettings())
```

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.

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'.

oracleTempSchema

Should be used in Oracle to specify a schema where the user has write privileges for storing temporary tables.

cohortDatabaseSchema

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

cohortTable

Name of the cohort table.

cohortId

The cohort definition ID used to reference the cohort in the cohort table.

covariateSettings

Either an object of type covariateSettings as created using one of the create-Covariate functions in the FeatureExtraction package, or a list of such objects.

Value

A data frame with cohort characteristics.

getIncidenceProportion

Compute incidence proportion for a cohort

Description

Returns yearly incidence proportion time series data stratified by age and gender

Usage

```
getIncidenceProportion(connectionDetails = NULL, connection = NULL,
  cohortDatabaseSchema, cohortTable, cdmDatabaseSchema,
  firstOccurrenceOnly = TRUE, minObservationTime = 365, cohortId)
```

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 table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'. |
| cohortTable | Name of the cohort table. |
| 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'. |
| firstOccurrenceOnly | Use only the first occurrence of the cohort per person? |
| minObservationTime | The minimum amount of observation time required before the occurrence of a cohort entry. This is also used to eliminate immortal time from the denominator. |
| cohortId | The cohort definition ID used to reference the cohort in the cohort table. |

Details

Returns a data frame of cohort count, background count, and incidence proportion per 1000 persons of cohort entry with the following stratifications: 1) no stratification, 2) gender stratification, 3) age (10-year) stratification, 4) calendar year and age (10-year) stratification, 5) calendar year and gender stratification, 6) calendar year, age (10-year), and gender stratification with option to save dataframes.

Value

A data frame

getInclusionStatistics

Get statistics on cohort inclusion criteria

Description

Get statistics on cohort inclusion criteria

Usage

```
getInclusionStatistics(connectionDetails = NULL, connection = NULL,
  resultsDatabaseSchema, cohortId, simplify = TRUE,
  cohortTable = "cohort", cohortInclusionTable = paste0(cohortTable,
    "_inclusion"), cohortInclusionResultTable = paste0(cohortTable,
    "_inclusion_result"), cohortInclusionStatsTable = paste0(cohortTable,
    "_inclusion_stats"), cohortSummaryStatsTable = paste0(cohortTable,
    "_summary_stats"))
```

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.
resultsDatabaseSchema	Schema name where the statistics tables reside. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortId	The cohort definition ID used to reference the cohort in the cohort table.
simplify	Simply output the attrition table?
cohortTable	Name of the cohort table. Used only to conveniently derive names of the four rule statistics tables.
cohortInclusionTable	Name of the inclusion table, one of the tables for storing inclusion rule statistics.
cohortInclusionResultTable	Name of the inclusion result table, one of the tables for storing inclusion rule statistics.
cohortInclusionStatsTable	Name of the inclusion stats table, one of the tables for storing inclusion rule statistics.
cohortSummaryStatsTable	Name of the summary stats table, one of the tables for storing inclusion rule statistics.

Value

If 'simplify = TRUE', this function returns a single data frame. Else a list of data frames is returned.

getInclusionStatisticsFromFiles	<i>Get inclusion criteria statistics from files</i>
---------------------------------	---

Description

Gets inclusion criteria statistics from files, as stored when using the `ROhdsiWebApi::insertCohortDefinitionSetInPa` function with `generateStats = TRUE`.

Usage

```
getInclusionStatisticsFromFiles(cohortId, folder,
  cohortInclusionFile = file.path(folder, "cohortInclusion.csv"),
  cohortInclusionResultFile = file.path(folder, "cohortIncResult.csv"),
  cohortInclusionStatsFile = file.path(folder, "cohortIncStats.csv"),
  cohortSummaryStatsFile = file.path(folder, "cohortSummaryStats.csv"),
  simplify = TRUE)
```

Arguments

- `cohortId` The cohort definition ID used to reference the cohort in the cohort table.
- `folder` The path to the folder where the inclusion statistics are stored.
- `cohortInclusionFile`
 Name of the inclusion table, one of the tables for storing inclusion rule statistics.
- `cohortInclusionResultFile`
 Name of the inclusion result table, one of the tables for storing inclusion rule statistics.
- `cohortInclusionStatsFile`
 Name of the inclusion stats table, one of the tables for storing inclusion rule statistics.
- `cohortSummaryStatsFile`
 Name of the summary stats table, one of the tables for storing inclusion rule statistics.
- `simplify` Simply output the attrition table?

Value

If ‘`simplify = TRUE`’, this function returns a single data frame. Else a list of data frames is returned.

<code>instantiateCohort</code>	<i>Instantiate a cohort</i>
--------------------------------	-----------------------------

Description

This function instantiates the cohort in the cohort table. Optionally, the inclusion rule statistics are computed and stored in the inclusion rule statistics tables described in [createCohortTable](#)).

Usage

```
instantiateCohort(connectionDetails = NULL, connection = NULL,
  cdmDatabaseSchema, oracleTempSchema = NULL,
  cohortDatabaseSchema = cdmDatabaseSchema, cohortTable = "cohort",
  baseUrl = NULL, webApiCohortId = NULL, cohortJson = NULL,
  cohortSql = NULL, cohortId = webApiCohortId,
  generateInclusionStats = FALSE,
  resultsDatabaseSchema = cohortDatabaseSchema,
  cohortInclusionTable = paste0(cohortTable, "_inclusion"),
  cohortInclusionResultTable = paste0(cohortTable, "_inclusion_result"),
  cohortInclusionStatsTable = paste0(cohortTable, "_inclusion_stats"),
  cohortSummaryStatsTable = paste0(cohortTable, "_summary_stats"))
```

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.
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'.
oracleTempSchema	Should be used in Oracle to specify a schema where the user has write privileges for storing temporary tables.
cohortDatabaseSchema	Schema name where your cohort table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortTable	Name of the cohort table.
baseUrl	The base URL for the WebApi instance, for example: "http://server.org:80/WebAPI". Needn't be provided if cohortJson and cohortSql are provided.
webApiCohortId	The ID of the cohort in the WebAPI instance. Needn't be provided if cohortJson and cohortSql are provided.
cohortJson	A character string containing the JSON of a cohort definition. Needn't be provided if baseUrl and cohortId are provided.
cohortSql	The OHDSI SQL representation of the same cohort definition. Needn't be provided if baseUrl and cohortId are provided.
cohortId	The cohort definition ID used to reference the cohort in the cohort table.
generateInclusionStats	Compute and store inclusion rule statistics?
resultsDatabaseSchema	Schema name where the statistics tables reside. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortInclusionTable	Name of the inclusion table, one of the tables for storing inclusion rule statistics.

cohortInclusionResultTable	Name of the inclusion result table, one of the tables for storing inclusion rule statistics.
cohortInclusionStatsTable	Name of the inclusion stats table, one of the tables for storing inclusion rule statistics.
cohortSummaryStatsTable	Name of the summary stats table, one of the tables for storing inclusion rule statistics.

plotIncidenceProportion

Plot incidence proportion by year, age, and gender

Description

Characterizes the incidence proportion of a phenotype as a time series visualization

Usage

```
plotIncidenceProportion(incidenceProportion,
  restrictToFullAgeData = FALSE, fileName = NULL)
```

Arguments

incidenceProportion	Incidence proportion time series data for plotting generated using getIncidenceProportion function.
restrictToFullAgeData	Restrict to panels having data on all ages?
fileName	Optional: name of the file where the plot should be saved, for example 'plot.png'. See the function ggsave in the ggplot2 package for supported file formats.

Details

Generates time series plots of the incidence proportion per 1000 persons of phenotype entry by year, age, and gender.

Value

A ggplot object. Use the [ggsave](#) function to save to file in a different format.

```
plotIncidenceProportionByYear
```

Plot incidence proportion by year

Description

Characterizes the incidence proportion of a phenotype as a time series visualization

Usage

```
plotIncidenceProportionByYear(incidenceProportion, fileName = NULL)
```

Arguments

`incidenceProportion`

Incidence proportion time series data for plotting generated using [getIncidenceProportion](#) function.

`fileName`

Optional: name of the file where the plot should be saved, for example 'plot.png'. See the function `ggsave` in the `ggplot2` package for supported file formats.

Details

Generates time series plots of the incidence proportion per 1000 persons of phenotype entry by year.

Value

A ggplot object. Use the [ggsave](#) function to save to file in a different format.

```
runStudyDiagnostics      Run study diagnostics
```

Description

Runs the study diagnostics on all (or a subset of) the cohorts instantiated using the `ROhdsiWebApi::insertCohortDefinition` function.

Usage

```
runStudyDiagnostics(packageName,
  cohortToCreateFile = "settings/CohortsToCreate.csv",
  connectionDetails = NULL, connection = NULL, cohortDatabaseSchema,
  cohortTable = "cohort", cohortIds = NULL,
  inclusionStatisticsFolder = NULL, exportFolder,
  runInclusionStatistics = TRUE, runIncludedSourceConcepts = TRUE,
  runOrphanConcepts = TRUE, runBreakdownIndexEvents = TRUE)
```


Arguments

packageName	The name of the package containing the cohort definitions
cohortToCreateFile	The location of the cohortToCreate file writing the package.
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 table resides. Note that for SQL Server, this should include both the database and schema name, for example 'scratch.dbo'.
cohortTable	Name of the cohort table.
cohortIds	Optionally, provide a subset of cohort IDs to restrict the diagnostics to.
inclusionStatisticsFolder	The folder where the inclusion rule statistics are stored. Can be left NULL if runInclusionStatistics = FALSE.
exportFolder	The folder where the output will be exported to. If this folder does not exist it will be created.
runInclusionStatistics	Generate and export statistic on the cohort incusion rules?
runIncludedSourceConcepts	Generate and export the source concepts included in the cohorts?
runOrphanConcepts	Generate and export potential orphan concepts?
runBreakdownIndexEvents	Generate and export the breakdown of index events?

StudyDiagnostics

*StudyDiagnostics***Description**

StudyDiagnostics

Index

breakDownIndexEvents, [2](#)

compareCohortCharacteristics, [3](#)

computeCohortOverlap, [4](#)

connect, [2](#), [4–12](#), [14](#), [17](#)

createCohortTable, [4](#), [13](#)

createConceptCountsTable, [5](#), [6](#), [8](#), [9](#)

createConnectionDetails, [2](#), [4–6](#), [8–12](#), [14](#),
[17](#)

findCohortIncludedSourceConcepts, [6](#)

findCohortOrphanConcepts, [7](#)

findOrphanConcepts, [9](#)

getCohortCharacteristics, [3](#), [10](#)

getIncidenceProportion, [11](#), [15](#), [16](#)

getInclusionStatistics, [12](#)

getInclusionStatisticsFromFiles, [13](#)

ggsave, [15](#), [16](#)

instantiateCohort, [13](#)

plotIncidenceProportion, [15](#)

plotIncidenceProportionByYear, [16](#)

runStudyDiagnostics, [16](#)

StudyDiagnostics, [17](#)

StudyDiagnostics-package
 (StudyDiagnostics), [17](#)