Package 'CaseCrossover'

November 7, 2018

Type Package

Title Case-Crossover

Version 1.1.0

Date 2018-011-07

Author Martijn Schuemie

Maintainer Martijn Schuemie <schuemie@ohdsi.org>

Description An R package for performing case-crossover and case-time-control analyses in an observational database in the OMOP Common Data Model.

VignetteBuilder knitr

Depends R (>= 3.2.2),
DatabaseConnector (>= 1.3.0),
survival

Imports Cyclops (>= 1.2.2),
ff,
ffbase (>= 0.12.1),
CaseControl (>= 1.3.2),
ParallelLogger

Suggests testthat,
knitr,
rmarkdown,
EmpiricalCalibration

License Apache License 2.0

NeedsCompilation no

RoxygenNote 6.1.0

R topics documented:

CaseCrossover   *CaseCrossover*

Description

   CaseCrossover

---

createCcrAnalysis   *Create a case-crossover analysis specification*

Description

   Create a case-crossover analysis specification

Usage

```
createCcrAnalysis(analysisId = 1, description = "",
  exposureType = NULL, outcomeType = NULL, nestingCohortType = NULL,
  getDbCaseCrossoverDataArgs, selectSubjectsToIncludeArgs,
  getExposureStatusArgs)
```

Arguments

analysisId	An integer that will be used later to refer to this specific set of analysis choices.
description	A short description of the analysis.
exposureType	If more than one exposure is provided for each exposureOutcomeNestingCohort, this field should be used to select the specific exposure to use in this analysis.
outcomeType	If more than one outcome is provided for each exposureOutcomeNestingCohort, this field should be used to select the specific outcome to use in this analysis.
nestingCohortType	
	If more than one nesting cohort is provided for each exposureOutcomeNestingCohort, this field should be used to select the specific nesting cohort to use in this analysis.

getDbCaseCrossoverDataArgs

> An object representing the arguments to be used when calling the [createGetDbCaseCrossoverDataA](#) function.

selectSubjectsToIncludeArgs

> An object representing the arguments to be used when calling the [selectSubjectsToInclude](#) function.

getExposureStatusArgs

> An object representing the arguments to be used when calling the [getExposureStatus](#) function.

Details

Create a set of analysis choices, to be used with the [runCcrAnalyses](#) function.

---

createExposureOutcomeNestingCohort
Create exposure-outcome-nesting-cohort combinations.

Description

Create exposure-outcome-nesting-cohort combinations.

Usage

```
createExposureOutcomeNestingCohort(exposureId, outcomeId,
  nestingCohortId = NULL)
```

Arguments

exposureId
: A concept ID indentifying the target drug in the exposure table. If multiple strategies for picking the exposure will be tested in the analysis, a named list of numbers can be provided instead. In the analysis, the name of the number to be used can be specified using the exposureType parameter in the [createCcrAnalysis](#) function.

outcomeId
: A concept ID indentifying the outcome in the outcome table. If multiple strategies for picking the outcome will be tested in the analysis, a named list of numbers can be provided instead. In the analysis, the name of the number to be used can be specified using the outcomeType parameter in the [createCcrAnalysis](#) function.

nestingCohortId
: A concept ID indentifying the nesting cohort in the nesting cohort table. If multiple strategies for picking the nesting cohort will be tested in the analysis, a named list of numbers can be provided instead. In the analysis, the name of the number to be used can be specified using the nestingCohortType parameter in the [createCcrAnalysis](#) function.

Details

Create a set of hypotheses of interest, to be used with the [runCcrAnalyses](#) function.

createGetDbCaseCrossoverDataArgs
                    Create a parameter object for the function getDbCaseCrossoverData

Description

Create a parameter object for the function getDbCaseCrossoverData

Usage

```
createGetDbCaseCrossoverDataArgs(useNestingCohort = FALSE,
  useObservationEndAsNestingEndDate = TRUE, getVisits = FALSE,
  studyStartDate = "", studyEndDate = "", getTimeControlData = FALSE)
```

Arguments

useNestingCohort

> Should the study be nested in a cohort (e.g. people witha specific indication)? If not, the study will be nestedin the general population.

useObservationEndAsNestingEndDate

> When using a nesting cohort, should the observationperiod end date be used instead of the cohort end date?

getVisits   Get data on visits? This is needed when performing atime- case-control study and matching on visit date isrequested later on.

studyStartDate A calendar date specifying the minimum date where data isused. Date format is 'yyyymmdd'.

studyEndDate   A calendar date specifying the maximum date where data isused. Date format is 'yyyymmdd'.

getTimeControlData

> Should data for time controls be fetched? (needed forcase-time-control analyses.

Details

Create an object defining the parameter values.

createGetExposureStatusArgs
                    Create a parameter object for the function getExposureStatus

Description

Create a parameter object for the function getExposureStatus

Usage

```
createGetExposureStatusArgs(firstExposureOnly = FALSE,
  riskWindowStart = -30, riskWindowEnd = 0,
  controlWindowOffsets = c(-60))
```

Arguments

firstExposureOnly

> Should only the first exposure per subject be included?

riskWindowStart

> The start of the risk window (in days) relative to the index date.This number should be non-positive.

riskWindowEnd The end of the risk window (in days) relative to the index date. Thisnumber should be non-positive.

controlWindowOffsets

> Offsets in days of the control windows relative to the case window.

Details

Create an object defining the parameter values.

---

createMatchingCriteria

Create matching criteria

Description

Create matching criteria

Usage

```
createMatchingCriteria(controlsPerCase = 1, matchOnAge = TRUE,
  ageCaliper = 2, matchOnGender = TRUE, matchOnProvider = FALSE,
  matchOnCareSite = FALSE, matchOnVisitDate = FALSE,
  visitDateCaliper = 30, matchOnTimeInCohort = FALSE,
  daysInCohortCaliper = 30)
```

Arguments

controlsPerCase

> Maximum number of controls to select per case.

matchOnAge Match on age?

ageCaliper Maximum difference (in years) in age when matching on age.

matchOnGender Match on gender?

matchOnProvider

> Match on provider (as specified in the person table)?

matchOnCareSite

> Match on care site (as specified in the person table)?

matchOnVisitDate

> Should the index date of the control be changed to the nearest visit date?

visitDateCaliper

> Maximum difference (in days) between the index date and the visit date when matching on visit date.

`matchOnTimeInCohort`

>   Match on time in nesting cohort? When not using nesting, this is interpreted as time observed prior to index.

`daysInCohortCaliper`

>   Maximum difference (in days) in time in cohort.

`createSelectSubjectsToIncludeArgs`
Create a parameter object for the function selectSubjectsToInclude

---

Description

Create a parameter object for the function selectSubjectsToInclude

Usage

```
createSelectSubjectsToIncludeArgs(firstOutcomeOnly = TRUE,
  washoutPeriod = 180, matchingCriteria = NULL, minAge = NULL,
  maxAge = NULL)
```

Arguments

`firstOutcomeOnly`

>   Use the first outcome per person?

`washoutPeriod`   Minimum required numbers of days of observation for inclusion as eithercase or control.

`matchingCriteria`

>   If provided, a case-time-control analysis will be performed and controlswill be matched based on these criteria.

`minAge`   Minimum age at which patient time will be included in the analysis. Notethat information prior to the min age is still used to determine exposurestatus after the minimum age (e.g. when a prescription was started justprior to reaching the minimum age). Also, outcomes occurring before theminimum age is reached will be considered as prior outcomes when usingfirst outcomes only. Age should be specified in years, but non-integervalues are allowed. If not specified, no age restriction will be applied.

`maxAge`   Maximum age at which patient time will be included in the analysis. Ageshould be specified in years, but non-integer values are allowed. If notspecified, no age restriction will be applied.

Details

Create an object defining the parameter values.

fitCaseCrossoverModel *Fit case-crossover model*

Description

Fit case-crossover model

Usage

```
fitCaseCrossoverModel(exposureStatus)
```

Arguments

exposureStatus   A data frame as generated using the getExposureStatus function.

Details

Fits a conditional logistic regression on the case-crossover data.

getAttritionTable *Get the attrition table for a set of subjects*

Description

Get the attrition table for a set of subjects

Usage

```
getAttritionTable(subjects)
```

Arguments

subjects   A data frame of subjects as generated by the function selectSubjectsToInclude.

Value

A data frame specifying the number of cases and events after various steps of filtering.

getDbCaseCrossoverData

Load case-crossover data from the database

Description

Load all data about the cases from the database.

Usage

```
getDbCaseCrossoverData(connectionDetails, cdmDatabaseSchema,
  oracleTempSchema = cdmDatabaseSchema,
  outcomeDatabaseSchema = cdmDatabaseSchema,
  outcomeTable = "condition_era", outcomeIds = c(),
  useNestingCohort = FALSE,
  nestingCohortDatabaseSchema = cdmDatabaseSchema,
  nestingCohortTable = "cohort", nestingCohortId = NULL,
  useObservationEndAsNestingEndDate = TRUE, getVisits = FALSE,
  exposureDatabaseSchema = cdmDatabaseSchema,
  exposureTable = "drug_era", exposureIds = c(), studyStartDate = "",
  studyEndDate = "", getTimeControlData = FALSE)
```

Arguments

connectionDetails

>An R object of type `ConnectionDetails` created using the function `createConnectionDetails` in the `DatabaseConnector` package.

cdmDatabaseSchema

>The name of the database schema that contains the OMOP CDM instance. Requires read permissions to this database. On SQL Server, this should specifiy both the database and the schema, so for example 'cdm_instance.dbo'.

oracleTempSchema

>A schema where temp tables can be created in Oracle.

outcomeDatabaseSchema

>The name of the database schema that is the location where the data used to define the outcome cohorts is available. If outcomeTable = CONDITION_ERA, outcomeDatabaseSchema is not used. Requires read permissions to this database.

outcomeTable

>The tablename that contains the outcome cohorts. If outcomeTable is not CONDITION_OCCURRENCE or CONDITION_ERA, then expectation is outcomeTable has format of COHORT table: COHORT_DEFINITION_ID, SUBJECT_ID, COHORT_START_DATE, COHORT_END_DATE.

outcomeIds

>A list of ids used to define outcomes. If outcomeTable = CONDITION_OCCURRENCE, the list is a set of ancestor CONCEPT_IDs, and all occurrences of all descendant concepts will be selected. If outcomeTable <> CONDITION_OCCURRENCE, the list contains records found in COHORT_DEFINITION_ID field.

useNestingCohort

>Should the study be nested in a cohort (e.g. people with a specific indication)? If not, the study will be nested in the general population.

nestingCohortDatabaseSchema

> The name of the database schema that is the location where the nesting cohort is defined.

nestingCohortTable

> Name of the table holding the nesting cohort. This table should have the same structure as the cohort table.

nestingCohortId

> A cohort definition ID identifying the records in the nestingCohortTable to use as nesting cohort.

useObservationEndAsNestingEndDate

> When using a nesting cohort, should the observation period end date be used instead of the cohort end date?

getVisits   Get data on visits? This is needed when performing a time- case-control study and matching on visit date is requested later on.

exposureDatabaseSchema

> The name of the database schema that is the location where the exposure data used to define the exposure cohorts is available. If exposureTable = DRUG_ERA, exposureDatabaseSchema is not used but assumed to be cdmSchema. Requires read permissions to this database.

exposureTable   The tablename that contains the exposure cohorts. If exposureTable <> DRUG_ERA, then expectation is exposureTable has format of COHORT table: cohort_concept_id, SUBJECT_ID, COHORT_START_DATE, COHORT_END_DATE.

exposureIds   A list of identifiers to define the exposures of interest. If exposureTable = DRUG_ERA, exposureIds should be CONCEPT_ID. If exposureTable <> DRUG_ERA, exposureIds is used to select the cohort_concept_id in the cohort-like table. If no exposureIds are provided, all drugs or cohorts in the exposureTable are included as exposures.

studyStartDate   A calendar date specifying the minimum date where data is used. Date format is 'yyyymmdd'.

studyEndDate   A calendar date specifying the maximum date where data is used. Date format is 'yyyymmdd'.

getTimeControlData

> Should data for time controls be fetched? (needed for case-time-control analyses.

Value

Returns an object of type `caseCrossoverData`, containing information on the cases, the nesting cohort, exposures, and optionally visits. Information about multiple outcomes can be captured at once for efficiency reasons. The generic `summary()` function has been implemented for this object.

---

getExposureStatus   *Get the exposure status for cases (and controls).*

Description

Get the exposure status for cases (and controls).

Usage

```
getExposureStatus(subjects, caseCrossoverData, exposureId,
  firstExposureOnly = FALSE, riskWindowStart = -30,
  riskWindowEnd = 0, controlWindowOffsets = c(-60))
```

Arguments

subjects	A data frame as generated using the selectSubjectsToInclude function.
caseCrossoverData	
	An object of type caseCrossoverData as generated using the getDbCaseCrossoverData function.
exposureId	The identifier of the exposure.
firstExposureOnly	
	Should only the first exposure per subject be included?
riskWindowStart	
	The start of the risk window (in days) relative to the index date. This number should be non-positive.
riskWindowEnd	The end of the risk window (in days) relative to the index date. This number should be non-positive.
controlWindowOffsets	
	Offsets in days of the control windows relative to the case window.

Details

This function determines the exposure status for a give, exposure ID in various windows relative to the index date.

Value

A data frame with these columns:

personId The person ID

indexDate The index date

isCase Is the person a case or a control?

stratumId The ID linking cases and controls in a matched set

isCaseWindow Is this a case window (as opposed to a control window)?

exposed Was the person exposed during the window?

---

loadCaseCrossoverData   *Load the case data from a folder*

Description

loadCaseCrossoverData loads an object of type caseCrossoverData from a folder in the file system.

Usage

```
loadCaseCrossoverData(folder, readOnly = TRUE)
```

Arguments

folder	The name of the folder containing the data.
readOnly	If true, the data is opened read only.

Details

The data will be written to a set of files in the folder specified by the user.

Value

An object of class `caseCrossoverData`.

---

loadCcrAnalysisList   *Load a list of ccrAnalysis from file*

Description

Load a list of objects of type `ccrAnalysis` from file. The file is in JSON format.

Usage

```
loadCcrAnalysisList(file)
```

Arguments

file	The name of the file

Value

A list of objects of type `ccrAnalysis`.

---

loadExposureOutcomeNestingCohortList
                    Load a list of exposureOutcomeNestingCohort from file

---

Description

Load a list of objects of type `exposureOutcomeNestingCohort` from file. The file is in JSON format.

Usage

```
loadExposureOutcomeNestingCohortList(file)
```

Arguments

file	The name of the file

Value

A list of objects of type `drugComparatorOutcome`.

---

runCcrAnalyses   *Run a list of analyses*

Description

Run a list of analyses

Usage

```
runCcrAnalyses(connectionDetails, cdmDatabaseSchema,
  oracleTempSchema = cdmDatabaseSchema,
  exposureDatabaseSchema = cdmDatabaseSchema,
  exposureTable = "drug_era",
  outcomeDatabaseSchema = cdmDatabaseSchema,
  outcomeTable = "condition_era",
  nestingCohortDatabaseSchema = cdmDatabaseSchema,
  nestingCohortTable = "condition_era", outputFolder = "./CcrOutput",
  ccrAnalysisList, exposureOutcomeNestingCohortList,
  getDbCaseCrossoverDataThreads = 1,
  selectSubjectsToIncludeThreads = 1, getExposureStatusThreads = 1,
  fitCaseCrossoverModelThreads = 1)
```

Arguments

connectionDetails

> An R object of type `ConnectionDetails` created using the function `createConnectionDetails` in the `DatabaseConnector` package.

cdmDatabaseSchema

> The name of the database schema that contains the OMOP CDM instance. Requires read permissions to this database. On SQL Server, this should specifiy both the database and the schema, so for example 'cdm_instance.dbo'.

oracleTempSchema

> A schema where temp tables can be created in Oracle.

exposureDatabaseSchema

> The name of the database schema that is the location where the exposure data used to define the exposure cohorts is available. If exposureTable = DRUG_ERA, exposureDatabaseSchema is not used but assumed to be cdmSchema. Requires read permissions to this database.

exposureTable   The tablename that contains the exposure cohorts. If exposureTable <> drug_era, then expectation is exposureTable has format of COHORT table: cohort_definition_id, subject_id, cohort_start_date, cohort_end_date.

outcomeDatabaseSchema

> The name of the database schema that is the location where the data used to define the outcome cohorts is available. If outcomeTable = CONDITION_ERA, outcomeDatabaseSchema is not used. Requires read permissions to this database.

outcomeTable   The tablename that contains the outcome cohorts. If outcomeTable is not CONDITION_OCCURRENCE or CONDITION_ERA, then expectation is outcomeTable has format of COHORT table: COHORT_DEFINITION_ID, SUBJECT_ID, COHORT_START_DATE, COHORT_END_DATE.

nestingCohortDatabaseSchema

      The name of the database schema that is the location where the nesting cohort
      is defined.

nestingCohortTable

      Name of the table holding the nesting cohort. This table should have the same
      structure as the cohort table.

outputFolder   Name of the folder where all the outputs will written to.

ccrAnalysisList

      A list of objects of type ccrAnalysis as created using the [createCcrAnalysis](#)
      function.

exposureOutcomeNestingCohortList

      A list of objects of type exposureOutcomeNestingCohort as created using the
      [createExposureOutcomeNestingCohort](#) function.

getDbCaseCrossoverDataThreads

      The number of parallel threads to use for building the caseControlData objects.

selectSubjectsToIncludeThreads

      The number of parallel threads to use for selecting subjects

getExposureStatusThreads

      The number of parallel threads to use for getting exposure status.

fitCaseCrossoverModelThreads

      The number of parallel threads to use for fitting the models.

Details

Run a list of analyses for the exposure-outcome-nesting cohorts of interest. This function will run
all specified analyses against all hypotheses of interest, meaning that the total number of outcome
models is 'length(ccrAnalysisList) * length(exposureOutcomeNestingCohortList)'. When you pro-
vide several analyses it will determine whether any of the analyses have anything in common, and
will take advantage of this fact. For example, if we specify several analyses that only differ in the
way the control windows are specified then this function will extract the data and select the subjects
only once, and re-use this in all the analysis.

---

saveCaseCrossoverData   *Save the case-crossover data to folder*

Description

saveCaseCrossoverData saves an object of type caseCrossoverData to folder.

Usage

```
saveCaseCrossoverData(caseCrossoverData, folder)
```

Arguments

caseCrossoverData

      An object of type caseCrossoverData as generated using [getDbCaseCrossoverData](#).

folder   The name of the folder where the data will be written. The folder should not yet
      exist.

Details

The data will be written to a set of files in the specified folder.

---

saveCcrAnalysisList *Save a list of ccrAnalysis to file*

Description

Write a list of objects of type ccrAnalysis to file. The file is in JSON format.

Usage

```
saveCcrAnalysisList(ccrAnalysisList, file)
```

Arguments

ccrAnalysisList

> The ccArnalysis list to be written to file

file   The name of the file where the results will be written

---

saveExposureOutcomeNestingCohortList
                          Save a list of drugComparatorOutcome to file

---

Description

Write a list of objects of type exposureOutcomeNestingCohort to file. The file is in JSON format.

Usage

```
saveExposureOutcomeNestingCohortList(exposureOutcomeNestingCohortList,
  file)
```

Arguments

exposureOutcomeNestingCohortList

> The exposureOutcomeNestingCohort list to be written to file

file   The name of the file where the results will be written

selectSubjectsToInclude

Select subjects to include

Description

Select subjects to include

Usage

```
selectSubjectsToInclude(caseCrossoverData, outcomeId,
  firstOutcomeOnly = TRUE, washoutPeriod = 180,
  matchingCriteria = NULL, minAge = NULL, maxAge = NULL)
```

Arguments

caseCrossoverData

An object of type caseCrossoverData as generated using the [getDbCaseCrossoverData](#) function.

outcomeId   The outcome ID of the cases for which we need to pick controls.

firstOutcomeOnly

Use the first outcome per person?

washoutPeriod   Minimum required numbers of days of observation for inclusion as either case or control.

matchingCriteria

If provided, a case-time-control analysis will be performed and controls will be matched based on these criteria.

minAge   Minimum age at which patient time will be included in the analysis. Note that information prior to the min age is still used to determine exposure status after the minimum age (e.g. when a prescription was started just prior to reaching the minimum age). Also, outcomes occurring before the minimum age is reached will be considered as prior outcomes when using first outcomes only. Age should be specified in years, but non-integer values are allowed. If not specified, no age restriction will be applied.

maxAge   Maximum age at which patient time will be included in the analysis. Age should be specified in years, but non-integer values are allowed. If not specified, no age restriction will be applied.

Details

Subject to include in the study are selected for a specific outcome, optionally filtering using a washout period, restricting to first occurrences of the outcome only, and restricting on age. If matching criteria are provided controls will be selected for each case. These controls will be used to adjust for time trends in exposure, turning the analysis into a case-time-control analysis (Suissa, 1995).

Value

A data frame with these columns:

personId The person ID

indexDate The index date

isCase Is the person a case or a control?

stratumId The ID linking cases and controls in a matched set

observationPeriodStartDate The observation period start date

References

Suissa S (1995) The case-time-control design. Epidemiology; 6:248-253.

---

summarizeCcrAnalyses   *Create a summary report of the analyses*

Description

Create a summary report of the analyses

Usage

```
summarizeCcrAnalyses(outcomeReference)
```

Arguments

outcomeReference

>   A data.frame as created by the runCcrAnalyses function.

Index