Package 'CaseCrossover'

April 20, 2017

Type Package

Title Case-Crossover

Version 0.0.1

Date 2017-04-19

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),
Cyclops (>= 1.2.2),
DatabaseConnector (>= 1.3.0),
survival,
FeatureExtraction (>= 1.0.1)

Imports RJDBC,
SqlRender (>= 1.1.1),
bit,
ff,
ffbase (>= 0.12.1),
Rcpp (>= 0.11.2),
OhdsiRTools (>= 1.1.1),
plyr,
CaseControl (>= 1.3.0)

Suggests testthat,
knitr,
rmarkdown,
EmpiricalCalibration

License Apache License 2.0

NeedsCompilation no

RoxygenNote 6.0.1

R topics documented:

`CaseCrossover` *CaseCrossover*

Description

CaseCrossover

`createMatchingCriteria` *Create matching criteria*

Description

Create matching criteria

Usage

```
createMatchingCriteria(controlsPerCase = 2, 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.

`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.

`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.

`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`.

---

saveCaseCrossoverData *Save the case-crossover data to folder*

Description

`saveCaseCrossoverData` saves an object of type caseCrossoverData to folder.

Usage

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

Arguments

folder	The name of the folder where the data will be written. The folder should not yet exist.
caseCrossover	An object of type caseCrossoverData as generated using getDbCaseCrossoverData.

Details

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

---

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.

Index