Package 'MethodEvaluation'

March 20, 2015

Type Package

Title Package for evaluation of estimation methods

Version 0.0.1

Date 2015-03-16

Author Martijn J. Schuemie [aut, cre],

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

Description This package contains resources for the evaluation of the performance of methods that aim to estimate
the magnitude (relative risk) of the effect of a drug on an outcome. These resources include reference sets for evaluating
methods on real data, as well as functions for inserting simulated effects in real data based on negative control drug-outcome pairs.
Further included are functions for the computation of the minimum detectable relative risks and functions for computing performance
statistics such as predictive accuracy, error and bias.

License Apache License 2.0

Depends R (>= 3.1.0),
ffbase,
DatabaseConnector (>= 1.1.2),
CohortMethod (>= 1.0.3)

Imports RJDBC,
SqlRender (>= 1.1.0),
pROC

Suggests testthat,

R topics documented:

`computeAuc` *Compute the area under the ROC curve*

Description

Compute the area under the ROC curve

Usage

```
computeAuc(methodResults, referenceSet, confidenceIntervals = TRUE)
```

`computeBias` *Compute the bias (mean error)*

Description

Compute the bias (mean error)

Usage

```
computeBias()
```

`computeMdrr` *Compute minimal detectable relative risk (MDRR)*

Description

`computeMdrr` computes the minimal detectable relative risk (MDRR) for drug-outcome pairs.

Usage

```
computeMdrr(connectionDetails, cdmDatabaseSchema,
  oracleTempSchema = cdmDatabaseSchema, exposureOutcomePairs,
  exposureDatabaseSchema = cdmDatabaseSchema, exposureTable = "drug_era",
  outcomeDatabaseSchema = cdmDatabaseSchema, outcomeTable = "condition_era",
  outcomeConditionTypeConceptIds = c())
```

Arguments

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

`cdmDatabaseSchema`
: Name of database schema that contains OMOP CDM and vocabulary.

`oracleTempSchema`
: For Oracle only: the name of the database schema where you want all temporary tables to be managed. Requires create/insert permissions to this database.

`exposureOutcomePairs`
: A data frame with at least two columns:
  - "exposureConceptId" containing the drug_concept_ID or cohort_definition_id of the exposure variable
  - "outcomeConceptId" containing the condition_concept_ID or cohort_definition_id of the outcome variable

`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 by 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 exposureTable = CONDITION_ERA, exposureDatabaseSchema is not used by assumed to be cdmSchema. Requires read permissions to this database.

`outcomeTable`
: The tablename that contains the outcome cohorts. If outcomeTable <> CONDITION_OCCURRENCE, then expectation is outcomeTable has format of COHORT table: COHORT_DEFINITION_ID, SUBJECT_ID, COHORT_START_DATE, COHORT_END_DATE.

`outcomeConditionTypeConceptIds`
: A list of TYPE_CONCEPT_ID values that will restrict condition occurrences. Only applicable if outcomeTable = CONDITION_OCCURRENCE.

Details

Computes the MDRR using simple power-calculations using person-level statistics stratified by age and gender.

Value

A data frame containing the MDRRs for the given exposure-outcome pairs.

Examples

```
## Not run:
  connectionDetails <- createConnectionDetails(dbms="sql server", server="RNDUSRDHIT07.jnj.com")
  exposureOutcomePairs = data.frame(exposureConceptId = c(767410,1314924,907879), outcomeConceptId = c(44438
  mdrrs <- computeMdrr(connectionDetails, "cdm_truven_mdcr" exposureOutcomePairs, outcomeTable = "condition_
```

```
## End(Not run)
```

`computeMse` *Compute the mean squared error*

Description

Compute the mean squared error

Usage

```
computeMse()
```

`createOutcomeCohorts` *Create outcomes of interest*

Description

Create outcomes of interest

Usage

```
createOutcomeCohorts(connectionDetails, cdmDatabaseSchema,
  createNewCohortTable = FALSE, cohortDatabaseSchema = cdmDatabaseSchema,
  cohortTable = "cohort", referenceSet = "omopReferenceSet")
```

Arguments

`connectionDetails`	An R object of type `ConnectionDetails` created using the function `createConnectionDetails` in the `DatabaseConnector` package.
`cdmDatabaseSchema`	A database schema containing health care data in the OMOP Commond Data Model. Note that for SQL Server, botth the database and schema should be specified, e.g. 'cdm_schema.dbo'
`createNewCohortTable`	Should a new cohort table be created, or should the outcomes be inserted in a existing table?
`cohortDatabaseSchema`	The database schema where the target table is located. Note that for SQL Server, botth the database and schema should be specified, e.g. 'cdm_schema.dbo'
`cohortTable`	The name of the table where the outcomes will be stored.

Details

This function will create the outcomes of interest referenced in the various reference sets. The outcomes of interest are derives using information like diagnoses, procedures, and drug prescriptions. The outcomes are stored in a table on the database server.

`euadrReferenceSet` *The EU-ADR reference set*

Description

A reference set of 43 drug-outcome pairs where we believe the drug causes the outcome (positive controls) and 50 drug-outcome pairs where we believe the drug does not cause the outcome (negative controls). The controls involve 10 health outcomes of interest. Note that originally, there was an additional positive control (Nimesulide and acute liver injury), but but Nimesulide is not in RxNorm, and is not available in many countries).

Usage

```
data(euadrReferenceSet)
```

Format

A data frame with 399 rows and 10 variables:

exposureConceptId Concept ID identifying the exposure

exposureConceptName Name of the exposure

outcomeConceptId Concept ID identifying the outcome

outcomeConceptName Name of the outcome

groundTruth 0 = negative control, 1 = positive control

indicationConceptId Concept Id identifying the (primary) indication of the drug. To be used when one wants to nest the analysis within the indication

indicationConceptName Name of the indication

comparatorDrugConceptId Concept ID identifying a comparator drug that can be used as a counterfactual

comparatorDrugConceptName Name of the comparator drug

comparatorType How the comparator was selected

References

Coloma PM, Avillach P, Salvo F, Schuemie MJ, Ferrajolo C, Pariente A, Fourrier-RC)glat A, Molokhia M, Patadia V, van der Lei J, Sturkenboom M, TrifirC2 G. A reference standard for evaluation of methods for drug safety signal detection using electronic healthcare record databases. Drug Safety 36(1):13-23, 2013

filterOnMdrr *Filter data based on MDRR*

Description

Filters a dataset to those exposure-outcome pairs with sufficient power.

Usage

```
filterOnMdrr(data, mdrr, threshold = 1.25)
```

Arguments

data	A data frame with at least two columns: • "exposureConceptId" containing the drug_concept_ID or cohort_definition_id of the exposure variable • "outcomeConceptId" containing the condition_concept_ID or cohort_definition_id of the outcome variable
mdrr	A data frame as generated by the computeMdrr function.
threshold	The required minimum detectable relative risk.

Value

A subset of the data object.

injectSignals *Inject signals in database*

Description

Inject signals in database

Usage

```
injectSignals(connectionDetails, cdmDatabaseSchema,
  oracleTempSchema = cdmDatabaseSchema,
  exposureDatabaseSchema = cdmDatabaseSchema, exposureTable = "drug_era",
  outcomeDatabaseSchema = cdmDatabaseSchema, outcomeTable = "condition_era",
  outcomeConditionTypeConceptIds = c(),
  outputDatabaseSchema = cdmDatabaseSchema,
  outputTable = "generated_outcomes", exposureOutcomePairs,
  modelType = "poisson", buildOutcomeModel = TRUE,
  firstExposureOnly = FALSE, washoutWindow = 183, riskWindowStart = 0,
  riskWindowEnd = 0, addExposureDaysToEnd = TRUE,
  firstOutcomeOnly = FALSE, effectSizes = c(1, 1.25, 1.5, 2, 4, 8))
```

Arguments

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

`cdmDatabaseSchema`
: Name of database schema that contains OMOP CDM and vocabulary.

`oracleTempSchema`
: For Oracle only: the name of the database schema where you want all temporary tables to be managed. Requires create/insert permissions to this database.

`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 by 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 exposureTable = CONDITION_ERA, exposureDatabaseSchema is not used by assumed to be cdmSchema. Requires read permissions to this database.

`outcomeTable`
: The tablename that contains the outcome cohorts. If outcomeTable <> CONDITION_OCCURRENCE, then expectation is outcomeTable has format of COHORT table: COHORT_DEFINITION_ID, SUBJECT_ID, COHORT_START_DATE, COHORT_END_DATE.

`outcomeConditionTypeConceptIds`
: A list of TYPE_CONCEPT_ID values that will restrict condition occurrences. Only applicable if outcomeTable = CONDITION_OCCURRENCE.

`exposureOutcomePairs`
: A data frame with at least two columns:

  - "exposureConceptId" containing the drug_concept_ID or cohort_definition_id of the exposure variable
  - "outcomeConceptId" containing the condition_concept_ID or cohort_definition_id of the outcome variable

Details

This function will insert additional outcomes for a given set of drug-outcome pairs. It is assumed that these drug-outcome pairs represent negative controls, so the true relative risk before inserting any outcomes should be 1. There are several models for inserting the outcomes during the specified risk window of the drug.

`omopReferenceSet` *The OMOP reference set*

Description

A reference set of 165 drug-outcome pairs where we believe the drug causes the outcome (positive controls) and 234 drug-outcome pairs where we believe the drug does not cause the outcome (negative controls). The controls involve 4 health outcomes of interest: acute liver injury, acute kidney injury, acute myocardial infarction, and GI bleeding.

Usage

```
data(omopReferenceSet)
```

Format

A data frame with 399 rows and 10 variables:

exposureConceptId Concept ID identifying the exposure

exposureConceptName Name of the exposure

outcomeConceptId Concept ID identifying the outcome

outcomeConceptName Name of the outcome

groundTruth 0 = negative control, 1 = positive control

indicationConceptId Concept Id identifying the (primary) indication of the drug. To be used when one wants to nest the analysis within the indication

indicationConceptName Name of the indication

comparatorDrugConceptId Concept ID identifying a comparator drug that can be used as a counterfactual

comparatorDrugConceptName Name of the comparator drug

comparatorType How the comparator was selected

References

Ryan PB, Schuemie MJ, Welebob E, Duke J, Valentine S, Hartzema AG. Defining a reference set to support methodological research in drug safety. Drug Safety 36 Suppl 1:S33-47, 2013

---

plotBias *Plot the error distribution*

Description

Plot the error distribution

Usage

```
plotBias()
```

plotRoc *Plot the ROC curve*

Description

Plot the ROC curve

Usage

```
plotRoc()
```

Index