Package 'FeatureExtraction'

June 26, 2026

Type Package

Title Generating Features for a Cohort

Version 3.14.0

Date 2026-06-25

Maintainer Ger Inberg <g.inberg@erasmusmc.nl>

Description An R interface for generating features for a cohort using data in the Common Data Model. Features can be constructed using default or custom made feature definitions. Furthermore it's possible to aggregate features and get the summary statistics.

Depends R (>= 3.2.2),
DatabaseConnector (>= 3.0.0),
Andromeda (>= 1.0.0)

Imports methods,
dplyr,
rJava,
jsonlite,
SqlRender (>= 1.18.0),
ParallelLogger (>= 2.0.2),
cli,
pillar,
readr,
rlang,
RSQLite,
DBI,
checkmate,
vroom

Suggests testthat,
knitr,
rmarkdown,
Eunomia (>= 2.0.0),
withr,
curl,
httr

License Apache License 2.0

VignetteBuilder knitr

URL https://github.com/OHDSI/FeatureExtraction

BugReports https://github.com/OHDSI/FeatureExtraction/issues

NeedsCompilation no

RoxygenNote 7.3.3

Encoding UTF-8

Language en-US

Contents

.createLooCovariateSettings . . . 3
.getDbLooCovariateData . . . 3
aggregateCovariates . . . 4
computeStandardizedDifference . . . 5
convertPrespecSettingsToDetailedSettings . . . 6
CovariateData-class . . . 7
createAnalysisDetails . . . 7
createCohortAttrCovariateSettings . . . 8
createCohortBasedCovariateSettings . . . 10
createCohortBasedTemporalCovariateSettings . . . 11
createCovariateSettings . . . 12
createDefaultCovariateSettings . . . 25
createDefaultTemporalCovariateSettings . . . 26
createDetailedCovariateSettings . . . 27
createDetailedTemporalCovariateSettings . . . 28
createEmptyCovariateData . . . 29
createTable1 . . . 29
createTable1CovariateSettings . . . 31
createTemporalCovariateSettings . . . 32
createTemporalSequenceCovariateSettings . . . 38
filterByCohortDefinitionId . . . 41
filterByRowId . . . 41
getDbCohortAttrCovariatesData . . . 42
getDbCohortBasedCovariatesData . . . 45
getDbCovariateData . . . 47
getDbDefaultCovariateData . . . 50
getDefaultTable1Specifications . . . 53
isAggregatedCovariateData . . . 53
isCovariateData . . . 54
isTemporalCovariateData . . . 54
loadCovariateData . . . 55
replaceCovariateSettingsCohortSchemaTable . . . 56
saveCovariateData . . . 56
tidyCovariateData . . . 57

Index **59**

.createLooCovariateSettings *Get covariate settings*

Description

Get covariate settings

Usage

```
.createLooCovariateSettings(useLengthOfObs = TRUE)
```

Arguments

useLengthOfObs	if length of observations should be used

Value

Returns an object of type covariateSettings, containing settings for the covariates.

.getDbLooCovariateData *Get covariate information from the database*

Description

Get covariate information from the database

Usage

```
.getDbLooCovariateData(
  connection,
  tempEmulationSchema = NULL,
  cdmDatabaseSchema,
  cohortTable = "#cohort_person",
  cohortIds = c(-1),
  cdmVersion = "5",
  rowIdField = "subject_id",
  covariateSettings,
  aggregated = FALSE,
  minCharacterizationMean = 0,
  minCharacterizationCount = 0,
  ...
)
```

Arguments

connection	A connection to the server containing the schema as created using the connect function in the DatabaseConnector package. Either the connection or connectionDetails argument should be specified.
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.
cdmDatabaseSchema	The name of the database schema that contains the OMOP CDM instance. Requires read permissions to this database. On SQL Server, this should specify both the database and the schema, so for example 'cdm_instance.dbo'.
cohortTable	Name of the (temp) table holding the cohort for which we want to construct covariates
cohortIds	For which cohort ID(s) should covariates be constructed? If set to -1, covariates will be constructed for all cohorts in the specified cohort table.
cdmVersion	Define the OMOP CDM version used: currently supported is "5".
rowIdField	The name of the field in the cohort table that is to be used as the row_id field in the output table. This can be especially usefull if there is more than one period per person.
covariateSettings	Either an object of type covariateSettings as created using one of the createCovariate functions, or a list of such objects.
aggregated	Should aggregate statistics be computed instead of covariates per cohort entry?
minCharacterizationMean	The minimum mean value for binary characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that have very low values. The default is 0.
minCharacterizationCount	The minimum count value for binary characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that occur in very few cohort entries. The default is 0.
...	Additional arguments, not used.

Value

Returns an object of type covariateData, containing information on the covariates.

---

aggregateCovariates *Aggregate covariate data*

Description

Aggregate covariate data

Usage

```
aggregateCovariates(covariateData)
```

Arguments

covariateData	An object of type covariateData as generated using getDbCovariateData.

Value

An object of class covariateData.

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = 1,
  aggregated = FALSE,
  temporal = FALSE
)
aggregatedCovariateData <- aggregateCovariates(covariateData)
```

---

computeStandardizedDifference

Compute standardized difference of mean for all covariates.

Description

Computes the standardized difference for all covariates between two cohorts. The standardized difference is defined as the difference between the mean divided by the overall standard deviation.

Usage

```
computeStandardizedDifference(
  covariateData1,
  covariateData2,
  cohortId1 = NULL,
  cohortId2 = NULL
)
```

Arguments

covariateData1	The covariate data of the first cohort. Needs to be in aggregated format.
covariateData2	The covariate data of the second cohort. Needs to be in aggregated format.
cohortId1	If provided, covariateData1 will be restricted to this cohort. If not provided, covariateData1 is assumed to contain data on only 1 cohort.
cohortId2	If provided, covariateData2 will be restricted to this cohort. If not provided, covariateData2 is assumed to contain data on only 1 cohort.

Value

A data frame with means and standard deviations per cohort as well as the standardized difference of mean.

Examples

```
binaryCovDataFile <- system.file("testdata/binaryCovariateData.zip",
  package = "FeatureExtraction"
)
covariateData1 <- loadCovariateData(binaryCovDataFile)
covariateData2 <- loadCovariateData(binaryCovDataFile)
covDataDiff <- computeStandardizedDifference(
  covariateData1,
  covariateData2,
  cohortId1 = 1,
  cohortId2 = 2
)
```

---

convertPrespecSettingsToDetailedSettings

Convert prespecified covariate settings into detailed covariate settings

---

Description

Convert prespecified covariate settings into detailed covariate settings

Usage

```
convertPrespecSettingsToDetailedSettings(covariateSettings)
```

Arguments

covariateSettings	An object of type covariateSettings as created for example by the createCovariateSettings function.

Details

For advanced users only.

Value

An object of type covariateSettings, to be used in other functions.

Examples

```
covSettings <- createDefaultCovariateSettings()
detailedSettings <- convertPrespecSettingsToDetailedSettings(covariateSettings = covSettings)
```

CovariateData-class *Covariate Data*

Description

CovariateData is an S4 class that inherits from Andromeda. It contains information on covariates, which can be either captured on a per-person basis, or aggregated across the cohort(s).

By default covariates refer to a specific time period, with for example different covariate IDs for whether a diagnosis code was observed in the year before and month before index date. However, a CovariateData can also be temporal, meaning that next to a covariate ID there is also a time ID, which identifies the (user specified) time window the covariate was captured.

A CovariateData object is typically created using getDbCovariateData, can only be saved using saveCovariateData, and loaded using loadCovariateData.

Usage

```
## S4 method for signature 'CovariateData'
show(object)

## S4 method for signature 'CovariateData'
summary(object)
```

Arguments

object	An object of class 'CovariateData'.

See Also

isCovariateData, isAggregatedCovariateData, isTemporalCovariateData

createAnalysisDetails *Create detailed covariate settings*

Description

Create detailed covariate settings

Usage

```
createAnalysisDetails(
  analysisId,
  sqlFileName,
  parameters,
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

Arguments

analysisId	An integer between 0 and 999 that uniquely identifies this analysis.
sqlFileName	The name of the parameterized SQL file embedded in the featureExtraction package.
parameters	The list of parameter values used to render the template SQL.
includedCovariateConceptIds	A list of concept IDs that should be used to construct covariates.
addDescendantsToInclude	Should descendant concept IDs be added to the list of concepts to include?
excludedCovariateConceptIds	A list of concept IDs that should NOT be used to construct covariates.
addDescendantsToExclude	Should descendant concept IDs be added to the list of concepts to exclude?
includedCovariateIds	A list of covariate IDs that should be restricted to.

Details

creates an object specifying in detail how covariates should be constructed from data in the CDM model. Warning: this function is for advanced users only.

Value

An object of type analysisDetail, to be used in createDetailedCovariateSettings or createDetailedTemporalC

Examples

```
analysisDetails <- createAnalysisDetails(
  analysisId = 1,
  sqlFileName = "DemographicsGender.sql",
  parameters = list(
    analysisId = 1,
    analysisName = "Gender",
    domainId = "Demographics"
  ),
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

---

createCohortAttrCovariateSettings

Create cohort attribute covariate settings

Description

Create cohort attribute covariate settings

Usage

```
createCohortAttrCovariateSettings(
  analysisId = -1,
  attrDatabaseSchema,
  attrDefinitionTable = "attribute_definition",
  cohortAttrTable = "cohort_attribute",
  includeAttrIds = c(),
  isBinary = FALSE,
  missingMeansZero = FALSE
)
```

Arguments

analysisId	A unique identifier for this analysis.
attrDatabaseSchema	The database schema where the attribute definition and cohort attribute table can be found.
attrDefinitionTable	The name of the attribute definition table.
cohortAttrTable	The name of the cohort attribute table.
includeAttrIds	(optional) A list of attribute definition IDs to restrict to.
isBinary	Needed for aggregation: Are these binary variables? Binary variables should only have the values 0 or 1.
missingMeansZero	Needed for aggregation: For continuous values, should missing values be interpreted as 0?

Details

Creates an object specifying where the cohort attributes can be found to construct covariates. The attributes should be defined in a table with the same structure as the attribute_definition table in the Common Data Model. It should at least have these columns:

attribute_definition_id A unique identifier of type integer.

attribute_name A short description of the attribute.

The cohort attributes themselves should be stored in a table with the same format as the cohort_attribute table in the Common Data Model. It should at least have these columns:

cohort_definition_id A key to link to the cohort table.

subject_id A key to link to the cohort table.

cohort_start_date A key to link to the cohort table.

attribute_definition_id An foreign key linking to the attribute definition table.

value_as_number A real number.

Value

An object of type `covariateSettings`, to be used in other functions.

Examples

```
covariateSettings <- createCohortAttrCovariateSettings(
  analysisId = 1,
  attrDatabaseSchema = "main",
  attrDefinitionTable = "attribute_definition",
  cohortAttrTable = "cohort_attribute",
  includeAttrIds = c(1),
  isBinary = FALSE,
  missingMeansZero = FALSE
)
```

---

createCohortBasedCovariateSettings

Create settings for covariates based on other cohorts

Description

Create settings for covariates based on other cohorts

Usage

```
createCohortBasedCovariateSettings(
  analysisId,
  covariateCohortDatabaseSchema = NULL,
  covariateCohortTable = NULL,
  covariateCohorts,
  valueType = "binary",
  startDay = -365,
  endDay = 0,
  includedCovariateIds = c(),
  warnOnAnalysisIdOverlap = TRUE
)
```

Arguments

analysisId	A unique identifier for this analysis.
covariateCohortDatabaseSchema	The database schema where the cohorts used to define the covariates can be found. If set to NULL, the database schema will be guessed, for example using the same one as for the main cohorts.
covariateCohortTable	The table where the cohorts used to define the covariates can be found. If set to NULL, the table will be guessed, for example using the same one as for the main cohorts.
covariateCohorts	A data frame with at least two columns: 'cohortId' and 'cohortName'. The cohort ID should correspond to the cohort_definition_id of the cohort to use for creating a covariate.

`valueType` Either 'binary' or 'count'. When `valueType = 'count'`, the covariate value will be the number of times the cohort was observed in the window.

`startDay` What is the start day (relative to the index date) of the covariate window?

`endDay` What is the end day (relative to the index date) of the covariate window?

`includedCovariateIds`
A list of covariate IDs that should be restricted to.

`warnOnAnalysisIdOverlap`
Warn if the provided 'analysisId' overlaps with any predefined analysis as available in the 'createCovariateSettings()' function.

Details

Creates an object specifying covariates to be constructed based on the presence of other cohorts.

Value

An object of type `covariateSettings`, to be used in other functions.

`createCohortBasedTemporalCovariateSettings`
Create settings for temporal covariates based on other cohorts

---

Description

Create settings for temporal covariates based on other cohorts

Usage

```
createCohortBasedTemporalCovariateSettings(
  analysisId,
  covariateCohortDatabaseSchema = NULL,
  covariateCohortTable = NULL,
  covariateCohorts,
  valueType = "binary",
  temporalStartDays = -365:-1,
  temporalEndDays = -365:-1,
  includedCovariateIds = c(),
  warnOnAnalysisIdOverlap = TRUE
)
```

Arguments

`analysisId` A unique identifier for this analysis.

`covariateCohortDatabaseSchema`
The database schema where the cohorts used to define the covariates can be found. If set to `NULL`, the database schema will be guessed, for example using the same one as for the main cohorts.

`covariateCohortTable`
The table where the cohorts used to define the covariates can be found. If set to `NULL`, the table will be guessed, for example using the same one as for the main cohorts.

`covariateCohorts`
: A data frame with at least two columns: 'cohortId' and 'cohortName'. The cohort ID should correspond to the `cohort_definition_id` of the cohort to use for creating a covariate.

`valueType`
: Either 'binary' or 'count'. When `valueType = 'count'`, the covariate value will be the number of times the cohort was observed in the window.

`temporalStartDays`
: A list of integers representing the start of a time period, relative to the index date. 0 indicates the index date, -1 indicates the day before the index date, etc. The start day is included in the time period.

`temporalEndDays`
: A list of integers representing the end of a time period, relative to the index date. 0 indicates the index date, -1 indicates the day before the index date, etc. The end day is included in the time period.

`includedCovariateIds`
: A list of covariate IDs that should be restricted to.

`warnOnAnalysisIdOverlap`
: Warn if the provided 'analysisId' overlaps with any predefined analysis as available in the 'createTemporalCovariateSettings()' function.

Details

Creates an object specifying temporal covariates to be constructed based on the presence of other cohorts.

Value

An object of type `covariateSettings`, to be used in other functions.

`createCovariateSettings` *Create covariate settings*

Description

Create covariate settings

Usage

```
createCovariateSettings(
  useDemographicsGender = FALSE,
  useDemographicsAge = FALSE,
  useDemographicsAgeGroup = FALSE,
  useDemographicsRace = FALSE,
  useDemographicsEthnicity = FALSE,
  useDemographicsIndexYear = FALSE,
  useDemographicsIndexMonth = FALSE,
  useDemographicsPriorObservationTime = FALSE,
  useDemographicsPostObservationTime = FALSE,
  useDemographicsTimeInCohort = FALSE,
  useDemographicsIndexYearMonth = FALSE,
```

```
    useCareSiteId = FALSE,
    useConditionOccurrenceAnyTimePrior = FALSE,
    useConditionOccurrenceLongTerm = FALSE,
    useConditionOccurrenceMediumTerm = FALSE,
    useConditionOccurrenceShortTerm = FALSE,
    useConditionOccurrencePrimaryInpatientAnyTimePrior = FALSE,
    useConditionOccurrencePrimaryInpatientLongTerm = FALSE,
    useConditionOccurrencePrimaryInpatientMediumTerm = FALSE,
    useConditionOccurrencePrimaryInpatientShortTerm = FALSE,
    useConditionEraAnyTimePrior = FALSE,
    useConditionEraLongTerm = FALSE,
    useConditionEraMediumTerm = FALSE,
    useConditionEraShortTerm = FALSE,
    useConditionEraOverlapping = FALSE,
    useConditionEraStartLongTerm = FALSE,
    useConditionEraStartMediumTerm = FALSE,
    useConditionEraStartShortTerm = FALSE,
    useConditionGroupEraAnyTimePrior = FALSE,
    useConditionGroupEraLongTerm = FALSE,
    useConditionGroupEraMediumTerm = FALSE,
    useConditionGroupEraShortTerm = FALSE,
    useConditionGroupEraOverlapping = FALSE,
    useConditionGroupEraStartLongTerm = FALSE,
    useConditionGroupEraStartMediumTerm = FALSE,
    useConditionGroupEraStartShortTerm = FALSE,
    useDrugExposureAnyTimePrior = FALSE,
    useDrugExposureLongTerm = FALSE,
    useDrugExposureMediumTerm = FALSE,
    useDrugExposureShortTerm = FALSE,
    useDrugEraAnyTimePrior = FALSE,
    useDrugEraLongTerm = FALSE,
    useDrugEraMediumTerm = FALSE,
    useDrugEraShortTerm = FALSE,
    useDrugEraOverlapping = FALSE,
    useDrugEraStartLongTerm = FALSE,
    useDrugEraStartMediumTerm = FALSE,
    useDrugEraStartShortTerm = FALSE,
    useDrugGroupEraAnyTimePrior = FALSE,
    useDrugGroupEraLongTerm = FALSE,
    useDrugGroupEraMediumTerm = FALSE,
    useDrugGroupEraShortTerm = FALSE,
    useDrugGroupEraOverlapping = FALSE,
    useDrugGroupEraStartLongTerm = FALSE,
    useDrugGroupEraStartMediumTerm = FALSE,
    useDrugGroupEraStartShortTerm = FALSE,
    useProcedureOccurrenceAnyTimePrior = FALSE,
    useProcedureOccurrenceLongTerm = FALSE,
    useProcedureOccurrenceMediumTerm = FALSE,
    useProcedureOccurrenceShortTerm = FALSE,
    useDeviceExposureAnyTimePrior = FALSE,
    useDeviceExposureLongTerm = FALSE,
    useDeviceExposureMediumTerm = FALSE,
```

```
useDeviceExposureShortTerm = FALSE,
useMeasurementAnyTimePrior = FALSE,
useMeasurementLongTerm = FALSE,
useMeasurementMediumTerm = FALSE,
useMeasurementShortTerm = FALSE,
useMeasurementValueAnyTimePrior = FALSE,
useMeasurementValueLongTerm = FALSE,
useMeasurementValueMediumTerm = FALSE,
useMeasurementValueShortTerm = FALSE,
useMeasurementRangeGroupAnyTimePrior = FALSE,
useMeasurementRangeGroupLongTerm = FALSE,
useMeasurementRangeGroupMediumTerm = FALSE,
useMeasurementRangeGroupShortTerm = FALSE,
useMeasurementValueAsConceptAnyTimePrior = FALSE,
useMeasurementValueAsConceptLongTerm = FALSE,
useMeasurementValueAsConceptMediumTerm = FALSE,
useMeasurementValueAsConceptShortTerm = FALSE,
useObservationAnyTimePrior = FALSE,
useObservationLongTerm = FALSE,
useObservationMediumTerm = FALSE,
useObservationShortTerm = FALSE,
useObservationValueAsConceptAnyTimePrior = FALSE,
useObservationValueAsConceptLongTerm = FALSE,
useObservationValueAsConceptMediumTerm = FALSE,
useObservationValueAsConceptShortTerm = FALSE,
useObservationValueAnyTimePrior = FALSE,
useObservationValueLongTerm = FALSE,
useObservationValueMediumTerm = FALSE,
useObservationValueShortTerm = FALSE,
useCharlsonIndex = FALSE,
useDcsi = FALSE,
useChads2 = FALSE,
useChads2Vasc = FALSE,
useHfrs = FALSE,
useDistinctConditionCountLongTerm = FALSE,
useDistinctConditionCountMediumTerm = FALSE,
useDistinctConditionCountShortTerm = FALSE,
useDistinctIngredientCountLongTerm = FALSE,
useDistinctIngredientCountMediumTerm = FALSE,
useDistinctIngredientCountShortTerm = FALSE,
useDistinctProcedureCountLongTerm = FALSE,
useDistinctProcedureCountMediumTerm = FALSE,
useDistinctProcedureCountShortTerm = FALSE,
useDistinctMeasurementCountLongTerm = FALSE,
useDistinctMeasurementCountMediumTerm = FALSE,
useDistinctMeasurementCountShortTerm = FALSE,
useDistinctObservationCountLongTerm = FALSE,
useDistinctObservationCountMediumTerm = FALSE,
useDistinctObservationCountShortTerm = FALSE,
useVisitCountLongTerm = FALSE,
useVisitCountMediumTerm = FALSE,
useVisitCountShortTerm = FALSE,
```

```
  useVisitConceptCountLongTerm = FALSE,
  useVisitConceptCountMediumTerm = FALSE,
  useVisitConceptCountShortTerm = FALSE,
  longTermStartDays = -365,
  mediumTermStartDays = -180,
  shortTermStartDays = -30,
  endDays = 0,
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

Arguments

`useDemographicsGender`
: Gender of the subject. (analysis ID 1)

`useDemographicsAge`
: Age of the subject on the index date (in years). (analysis ID 2)

`useDemographicsAgeGroup`
: Age of the subject on the index date (in 5 year age groups) (analysis ID 3)

`useDemographicsRace`
: Race of the subject. (analysis ID 4)

`useDemographicsEthnicity`
: Ethnicity of the subject. (analysis ID 5)

`useDemographicsIndexYear`
: Year of the index date. (analysis ID 6)

`useDemographicsIndexMonth`
: Month of the index date. (analysis ID 7)

`useDemographicsPriorObservationTime`
: Number of continuous days of observation time preceding the index date. (analysis ID 8)

`useDemographicsPostObservationTime`
: Number of continuous days of observation time following the index date. (analysis ID 9)

`useDemographicsTimeInCohort`
: Number of days of observation time during cohort period. (analysis ID 10)

`useDemographicsIndexYearMonth`
: Both calendar year and month of the index date in a single variable. (analysis ID 11)

`useCareSiteId`
: Care site associated with the cohort start, pulled from the visit_detail, visit_occurrence, or person table, in that order. (analysis ID 12)

`useConditionOccurrenceAnyTimePrior`
: One covariate per condition in the condition_occurrence table starting any time prior to index. (analysis ID 101)

`useConditionOccurrenceLongTerm`
: One covariate per condition in the condition_occurrence table starting in the long term window. (analysis ID 102)

useConditionOccurrenceMediumTerm
: One covariate per condition in the condition_occurrence table starting in the medium term window. (analysis ID 103)

useConditionOccurrenceShortTerm
: One covariate per condition in the condition_occurrence table starting in the short term window. (analysis ID 104)

useConditionOccurrencePrimaryInpatientAnyTimePrior
: One covariate per condition observed as a primary diagnosis in an inpatient setting in the condition_occurrence table starting any time prior to index. (analysis ID 105)

useConditionOccurrencePrimaryInpatientLongTerm
: One covariate per condition observed as a primary diagnosis in an inpatient setting in the condition_occurrence table starting in the long term window. (analysis ID 106)

useConditionOccurrencePrimaryInpatientMediumTerm
: One covariate per condition observed as a primary diagnosis in an inpatient setting in the condition_occurrence table starting in the medium term window. (analysis ID 107)

useConditionOccurrencePrimaryInpatientShortTerm
: One covariate per condition observed as a primary diagnosis in an inpatient setting in the condition_occurrence table starting in the short term window. (analysis ID 108)

useConditionEraAnyTimePrior
: One covariate per condition in the condition_era table overlapping with any time prior to index. (analysis ID 201)

useConditionEraLongTerm
: One covariate per condition in the condition_era table overlapping with any part of the long term window. (analysis ID 202)

useConditionEraMediumTerm
: One covariate per condition in the condition_era table overlapping with any part of the medium term window. (analysis ID 203)

useConditionEraShortTerm
: One covariate per condition in the condition_era table overlapping with any part of the short term window. (analysis ID 204)

useConditionEraOverlapping
: One covariate per condition in the condition_era table overlapping with the end of the risk window. (analysis ID 205)

useConditionEraStartLongTerm
: One covariate per condition in the condition_era table starting in the long term window. (analysis ID 206)

useConditionEraStartMediumTerm
: One covariate per condition in the condition_era table starting in the medium term window. (analysis ID 207)

useConditionEraStartShortTerm
: One covariate per condition in the condition_era table starting in the short term window. (analysis ID 208)

useConditionGroupEraAnyTimePrior
: One covariate per condition era rolled up to groups in the condition_era table overlapping with any time prior to index. (analysis ID 209)

useConditionGroupEraLongTerm
: One covariate per condition era rolled up to groups in the condition_era table overlapping with any part of the long term window. (analysis ID 210)

useConditionGroupEraMediumTerm
: One covariate per condition era rolled up to groups in the condition_era table overlapping with any part of the medium term window. (analysis ID 211)

useConditionGroupEraShortTerm
: One covariate per condition era rolled up to groups in the condition_era table overlapping with any part of the short term window. (analysis ID 212)

useConditionGroupEraOverlapping
: One covariate per condition era rolled up to groups in the condition_era table overlapping with the end of the risk window. (analysis ID 213)

useConditionGroupEraStartLongTerm
: One covariate per condition era rolled up to groups in the condition_era table starting in the long term window. (analysis ID 214)

useConditionGroupEraStartMediumTerm
: One covariate per condition era rolled up to groups in the condition_era table starting in the medium term window. (analysis ID 215)

useConditionGroupEraStartShortTerm
: One covariate per condition era rolled up to groups in the condition_era table starting in the short term window. (analysis ID 216)

useDrugExposureAnyTimePrior
: One covariate per drug in the drug_exposure table starting any time prior to index. (analysis ID 301)

useDrugExposureLongTerm
: One covariate per drug in the drug_exposure table starting in the long term window. (analysis ID 302)

useDrugExposureMediumTerm
: One covariate per drug in the drug_exposure table starting in the medium term window. (analysis ID 303)

useDrugExposureShortTerm
: One covariate per drug in the drug_exposure table starting in the short term window. (analysis ID 304)

useDrugEraAnyTimePrior
: One covariate per drug in the drug_era table overlapping with any time prior to index. (analysis ID 401)

useDrugEraLongTerm
: One covariate per drug in the drug_era table overlapping with any part of the long term window. (analysis ID 402)

useDrugEraMediumTerm
: One covariate per drug in the drug_era table overlapping with any part of the medium term window. (analysis ID 403)

useDrugEraShortTerm
: One covariate per drug in the drug_era table overlapping with any part of the short term window. (analysis ID 404)

useDrugEraOverlapping
: One covariate per drug in the drug_era table overlapping with the end of the risk window. (analysis ID 405)

useDrugEraStartLongTerm
: One covariate per drug in the drug_era table starting in the long term window. (analysis ID 406)

useDrugEraStartMediumTerm
: One covariate per drug in the drug_era table starting in the medium term window. (analysis ID 407)

useDrugEraStartShortTerm
: One covariate per drug in the drug_era table starting in the short term window. (analysis ID 408)

useDrugGroupEraAnyTimePrior
: One covariate per drug rolled up to ATC groups in the drug_era table overlapping with any time prior to index. (analysis ID 409)

useDrugGroupEraLongTerm
: One covariate per drug rolled up to ATC groups in the drug_era table overlapping with any part of the long term window. (analysis ID 410)

useDrugGroupEraMediumTerm
: One covariate per drug rolled up to ATC groups in the drug_era table overlapping with any part of the medium term window. (analysis ID 411)

useDrugGroupEraShortTerm
: One covariate per drug rolled up to ATC groups in the drug_era table overlapping with any part of the short term window. (analysis ID 412)

useDrugGroupEraOverlapping
: One covariate per drug rolled up to ATC groups in the drug_era table overlapping with the end of the risk window. (analysis ID 413)

useDrugGroupEraStartLongTerm
: One covariate per drug rolled up to ATC groups in the drug_era table starting in the long term window. (analysis ID 414)

useDrugGroupEraStartMediumTerm
: One covariate per drug rolled up to ATC groups in the drug_era table starting in the medium term window. (analysis ID 415)

useDrugGroupEraStartShortTerm
: One covariate per drug rolled up to ATC groups in the drug_era table starting in the short term window. (analysis ID 416)

useProcedureOccurrenceAnyTimePrior
: One covariate per procedure in the procedure_occurrence table any time prior to index. (analysis ID 501)

useProcedureOccurrenceLongTerm
: One covariate per procedure in the procedure_occurrence table in the long term window. (analysis ID 502)

useProcedureOccurrenceMediumTerm
: One covariate per procedure in the procedure_occurrence table in the medium term window. (analysis ID 503)

useProcedureOccurrenceShortTerm
: One covariate per procedure in the procedure_occurrence table in the short term window. (analysis ID 504)

useDeviceExposureAnyTimePrior
: One covariate per device in the device exposure table starting any time prior to index. (analysis ID 601)

useDeviceExposureLongTerm
: One covariate per device in the device exposure table starting in the long term window. (analysis ID 602)

useDeviceExposureMediumTerm
: One covariate per device in the device exposure table starting in the medium term window. (analysis ID 603)

useDeviceExposureShortTerm
: One covariate per device in the device exposure table starting in the short term window. (analysis ID 604)

useMeasurementAnyTimePrior
: One covariate per measurement in the measurement table any time prior to index. (analysis ID 701)

useMeasurementLongTerm
: One covariate per measurement in the measurement table in the long term window. (analysis ID 702)

useMeasurementMediumTerm
: One covariate per measurement in the measurement table in the medium term window. (analysis ID 703)

useMeasurementShortTerm
: One covariate per measurement in the measurement table in the short term window. (analysis ID 704)

useMeasurementValueAnyTimePrior
: One covariate containing the value per measurement-unit combination any time prior to index. (analysis ID 705)

useMeasurementValueLongTerm
: One covariate containing the value per measurement-unit combination in the long term window. (analysis ID 706)

useMeasurementValueMediumTerm
: One covariate containing the value per measurement-unit combination in the medium term window. (analysis ID 707)

useMeasurementValueShortTerm
: One covariate containing the value per measurement-unit combination in the short term window. (analysis ID 708)

useMeasurementRangeGroupAnyTimePrior
: Covariates indicating whether measurements are below, within, or above normal range any time prior to index. (analysis ID 709)

useMeasurementRangeGroupLongTerm
: Covariates indicating whether measurements are below, within, or above normal range in the long term window. (analysis ID 710)

useMeasurementRangeGroupMediumTerm
: Covariates indicating whether measurements are below, within, or above normal range in the medium term window. (analysis ID 711)

useMeasurementRangeGroupShortTerm
: Covariates indicating whether measurements are below, within, or above normal range in the short term window. (analysis ID 712)

useMeasurementValueAsConceptAnyTimePrior
: One covariate per measurement-value concept combination any time prior to index. (analysis ID 713)

useMeasurementValueAsConceptLongTerm
: One covariate per measurement-value concept combination in the long term window. (analysis ID 714)

useMeasurementValueAsConceptMediumTerm
: One covariate per measurement-value concept combination in the medium term window. (analysis ID 715)

useMeasurementValueAsConceptShortTerm
: One covariate per measurement-value concept combination in the short term window. (analysis ID 716)

useObservationAnyTimePrior
: One covariate per observation in the observation table any time prior to index. (analysis ID 801)

useObservationLongTerm
: One covariate per observation in the observation table in the long term window. (analysis ID 802)

useObservationMediumTerm
: One covariate per observation in the observation table in the medium term window. (analysis ID 803)

useObservationShortTerm
: One covariate per observation in the observation table in the short term window. (analysis ID 804)

useObservationValueAsConceptAnyTimePrior
: One covariate per observation-value concept combination any time prior to index. (analysis ID 805)

useObservationValueAsConceptLongTerm
: One covariate per observation-value concept combination in the long term window. (analysis ID 806)

useObservationValueAsConceptMediumTerm
: One covariate per observation-value concept combination in the medium term window. (analysis ID 807)

useObservationValueAsConceptShortTerm
: One covariate per observation-value concept combination in the short term window. (analysis ID 808)

useObservationValueAnyTimePrior
: One covariate containing the value per observation-unit combination any time prior to index. (analysis ID 809)

useObservationValueLongTerm
: One covariate containing the value per observation-unit combination in the long term window. (analysis ID 810)

useObservationValueMediumTerm
: One covariate containing the value per observation-unit combination in the medium term window. (analysis ID 811)

useObservationValueShortTerm
: One covariate containing the value per observation-unit combination in the short term window. (analysis ID 812)

useCharlsonIndex
: The Charlson comorbidity index (Romano adaptation) using all conditions prior to the window end. (analysis ID 901)

useDcsi
: The Diabetes Comorbidity Severity Index (DCSI) using all conditions prior to the window end. (analysis ID 902)

useChads2
: The CHADS2 score using all conditions prior to the window end. (analysis ID 903)

useChads2Vasc
: The CHADS2VASc score using all conditions prior to the window end. (analysis ID 904)

useHfrs
: The Hospital Frailty Risk Score score using all conditions prior to the window end. (analysis ID 926)

useDistinctConditionCountLongTerm
: The number of distinct condition concepts observed in the long term window. (analysis ID 905)

useDistinctConditionCountMediumTerm
: The number of distinct condition concepts observed in the medium term window. (analysis ID 906)

useDistinctConditionCountShortTerm
: The number of distinct condition concepts observed in the short term window. (analysis ID 907)

useDistinctIngredientCountLongTerm
: The number of distinct ingredients observed in the long term window. (analysis ID 908)

useDistinctIngredientCountMediumTerm
: The number of distinct ingredients observed in the medium term window. (analysis ID 909)

useDistinctIngredientCountShortTerm
: The number of distinct ingredients observed in the short term window. (analysis ID 910)

useDistinctProcedureCountLongTerm
: The number of distinct procedures observed in the long term window. (analysis ID 911)

useDistinctProcedureCountMediumTerm
: The number of distinct procedures observed in the medium term window. (analysis ID 912)

useDistinctProcedureCountShortTerm
: The number of distinct procedures observed in the short term window. (analysis ID 913)

useDistinctMeasurementCountLongTerm
: The number of distinct measurements observed in the long term window. (analysis ID 914)

useDistinctMeasurementCountMediumTerm
: The number of distinct measurements observed in the medium term window. (analysis ID 915)

useDistinctMeasurementCountShortTerm
: The number of distinct measurements observed in the short term window. (analysis ID 916)

useDistinctObservationCountLongTerm
: The number of distinct observations observed in the long term window. (analysis ID 917)

useDistinctObservationCountMediumTerm
: The number of distinct observations observed in the medium term window. (analysis ID 918)

useDistinctObservationCountShortTerm
: The number of distinct observations observed in the short term window. (analysis ID 919)

useVisitCountLongTerm
: The number of visits observed in the long term window. (analysis ID 920)

useVisitCountMediumTerm
: The number of visits observed in the medium term window. (analysis ID 921)

useVisitCountShortTerm
: The number of visits observed in the short term window. (analysis ID 922)

useVisitConceptCountLongTerm
: The number of visits observed in the long term window, stratified by visit concept ID. (analysis ID 923)

useVisitConceptCountMediumTerm
: The number of visits observed in the medium term window, stratified by visit concept ID. (analysis ID 924)

useVisitConceptCountShortTerm
: The number of visits observed in the short term window, stratified by visit concept ID. (analysis ID 925)

longTermStartDays
: What is the start day (relative to the index date) of the long-term window?

mediumTermStartDays
: What is the start day (relative to the index date) of the medium-term window?

shortTermStartDays
: What is the start day (relative to the index date) of the short-term window?

endDays
: What is the end day (relative to the index date) of the window?

includedCovariateConceptIds
: A list of concept IDs that should be used to construct covariates.

addDescendantsToInclude
: Should descendant concept IDs be added to the list of concepts to include?

excludedCovariateConceptIds
: A list of concept IDs that should NOT be used to construct covariates.

addDescendantsToExclude
: Should descendant concept IDs be added to the list of concepts to exclude?

includedCovariateIds
: A list of covariate IDs that should be restricted to.

Details

creates an object specifying how covariates should be constructed from data in the CDM model.

Value

An object of type `covariateSettings`, to be used in other functions.

Examples

```
settings <- createCovariateSettings(
  useDemographicsGender = TRUE,
  useDemographicsAge = FALSE,
  useDemographicsAgeGroup = TRUE,
  useDemographicsRace = TRUE,
  useDemographicsEthnicity = TRUE,
  useDemographicsIndexYear = TRUE,
  useDemographicsIndexMonth = TRUE,
  useDemographicsPriorObservationTime = FALSE,
  useDemographicsPostObservationTime = FALSE,
  useDemographicsTimeInCohort = FALSE,
  useDemographicsIndexYearMonth = FALSE,
  useCareSiteId = FALSE,
  useConditionOccurrenceAnyTimePrior = FALSE,
  useConditionOccurrenceLongTerm = FALSE,
  useConditionOccurrenceMediumTerm = FALSE,
  useConditionOccurrenceShortTerm = FALSE,
  useConditionOccurrencePrimaryInpatientAnyTimePrior = FALSE,
  useConditionOccurrencePrimaryInpatientLongTerm = FALSE,
```

```
    useConditionOccurrencePrimaryInpatientMediumTerm = FALSE,
    useConditionOccurrencePrimaryInpatientShortTerm = FALSE,
    useConditionEraAnyTimePrior = FALSE,
    useConditionEraLongTerm = FALSE,
    useConditionEraMediumTerm = FALSE,
    useConditionEraShortTerm = FALSE,
    useConditionEraOverlapping = FALSE,
    useConditionEraStartLongTerm = FALSE,
    useConditionEraStartMediumTerm = FALSE,
    useConditionEraStartShortTerm = FALSE,
    useConditionGroupEraAnyTimePrior = FALSE,
    useConditionGroupEraLongTerm = TRUE,
    useConditionGroupEraMediumTerm = FALSE,
    useConditionGroupEraShortTerm = TRUE,
    useConditionGroupEraOverlapping = FALSE,
    useConditionGroupEraStartLongTerm = FALSE,
    useConditionGroupEraStartMediumTerm = FALSE,
    useConditionGroupEraStartShortTerm = FALSE,
    useDrugExposureAnyTimePrior = FALSE,
    useDrugExposureLongTerm = FALSE,
    useDrugExposureMediumTerm = FALSE,
    useDrugExposureShortTerm = FALSE,
    useDrugEraAnyTimePrior = FALSE,
    useDrugEraLongTerm = FALSE,
    useDrugEraMediumTerm = FALSE,
    useDrugEraShortTerm = FALSE,
    useDrugEraOverlapping = FALSE,
    useDrugEraStartLongTerm = FALSE,
    useDrugEraStartMediumTerm = FALSE,
    useDrugEraStartShortTerm = FALSE,
    useDrugGroupEraAnyTimePrior = FALSE,
    useDrugGroupEraLongTerm = TRUE,
    useDrugGroupEraMediumTerm = FALSE,
    useDrugGroupEraShortTerm = TRUE,
    useDrugGroupEraOverlapping = TRUE,
    useDrugGroupEraStartLongTerm = FALSE,
    useDrugGroupEraStartMediumTerm = FALSE,
    useDrugGroupEraStartShortTerm = FALSE,
    useProcedureOccurrenceAnyTimePrior = FALSE,
    useProcedureOccurrenceLongTerm = TRUE,
    useProcedureOccurrenceMediumTerm = FALSE,
    useProcedureOccurrenceShortTerm = TRUE,
    useDeviceExposureAnyTimePrior = FALSE,
    useDeviceExposureLongTerm = TRUE,
    useDeviceExposureMediumTerm = FALSE,
    useDeviceExposureShortTerm = TRUE,
    useMeasurementAnyTimePrior = FALSE,
    useMeasurementLongTerm = TRUE,
    useMeasurementMediumTerm = FALSE,
    useMeasurementShortTerm = TRUE,
    useMeasurementValueAnyTimePrior = FALSE,
    useMeasurementValueLongTerm = FALSE,
    useMeasurementValueMediumTerm = FALSE,
    useMeasurementValueShortTerm = FALSE,
    useMeasurementRangeGroupAnyTimePrior = FALSE,
    useMeasurementRangeGroupLongTerm = TRUE,
    useMeasurementRangeGroupMediumTerm = FALSE,
```

```
  useMeasurementRangeGroupShortTerm = TRUE,
  useMeasurementValueAsConceptAnyTimePrior = FALSE,
  useMeasurementValueAsConceptLongTerm = TRUE,
  useMeasurementValueAsConceptMediumTerm = FALSE,
  useMeasurementValueAsConceptShortTerm = TRUE,
  useObservationAnyTimePrior = FALSE,
  useObservationLongTerm = TRUE,
  useObservationMediumTerm = FALSE,
  useObservationShortTerm = TRUE,
  useObservationValueAsConceptAnyTimePrior = FALSE,
  useObservationValueAsConceptLongTerm = TRUE,
  useObservationValueAsConceptMediumTerm = FALSE,
  useObservationValueAsConceptShortTerm = TRUE,
  useObservationValueAnyTimePrior = FALSE,
  useObservationValueLongTerm = FALSE,
  useObservationValueMediumTerm = FALSE,
  useObservationValueShortTerm = FALSE,
  useCharlsonIndex = TRUE,
  useDcsi = TRUE,
  useChads2 = TRUE,
  useChads2Vasc = TRUE,
  useHfrs = FALSE,
  useDistinctConditionCountLongTerm = FALSE,
  useDistinctConditionCountMediumTerm = FALSE,
  useDistinctConditionCountShortTerm = FALSE,
  useDistinctIngredientCountLongTerm = FALSE,
  useDistinctIngredientCountMediumTerm = FALSE,
  useDistinctIngredientCountShortTerm = FALSE,
  useDistinctProcedureCountLongTerm = FALSE,
  useDistinctProcedureCountMediumTerm = FALSE,
  useDistinctProcedureCountShortTerm = FALSE,
  useDistinctMeasurementCountLongTerm = FALSE,
  useDistinctMeasurementCountMediumTerm = FALSE,
  useDistinctMeasurementCountShortTerm = FALSE,
  useDistinctObservationCountLongTerm = FALSE,
  useDistinctObservationCountMediumTerm = FALSE,
  useDistinctObservationCountShortTerm = FALSE,
  useVisitCountLongTerm = FALSE,
  useVisitCountMediumTerm = FALSE,
  useVisitCountShortTerm = FALSE,
  useVisitConceptCountLongTerm = FALSE,
  useVisitConceptCountMediumTerm = FALSE,
  useVisitConceptCountShortTerm = FALSE,
  longTermStartDays = -365,
  mediumTermStartDays = -180,
  shortTermStartDays = -30,
  endDays = 0,
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

createDefaultCovariateSettings
Create default covariate settings

Description

Create default covariate settings

Usage

```
createDefaultCovariateSettings(
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

Arguments

includedCovariateConceptIds	A list of concept IDs that should be used to construct covariates.
addDescendantsToInclude	Should descendant concept IDs be added to the list of concepts to include?
excludedCovariateConceptIds	A list of concept IDs that should NOT be used to construct covariates.
addDescendantsToExclude	Should descendant concept IDs be added to the list of concepts to exclude?
includedCovariateIds	A list of covariate IDs that should be restricted to.

Value

An object of type `covariateSettings`, to be used in other functions.

Examples

```
covSettings <- createDefaultCovariateSettings(
  includedCovariateConceptIds = c(1),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(2),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c(1)
)
```

createDefaultTemporalCovariateSettings
Create default covariate settings

Description

Create default covariate settings

Usage

```
createDefaultTemporalCovariateSettings(
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

Arguments

includedCovariateConceptIds	A list of concept IDs that should be used to construct covariates.
addDescendantsToInclude	Should descendant concept IDs be added to the list of concepts to include?
excludedCovariateConceptIds	A list of concept IDs that should NOT be used to construct covariates.
addDescendantsToExclude	Should descendant concept IDs be added to the list of concepts to exclude?
includedCovariateIds	A list of covariate IDs that should be restricted to.

Value

An object of type `covariateSettings`, to be used in other functions.

Examples

```
covSettings <- createDefaultTemporalCovariateSettings(
  includedCovariateConceptIds = c(1),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(2),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c(1)
)
```

createDetailedCovariateSettings
Create detailed covariate settings

Description

Create detailed covariate settings

Usage

```
createDetailedCovariateSettings(analyses = list())
```

Arguments

analyses	A list of analysisDetail objects as created using createAnalysisDetails.

Details

creates an object specifying in detail how covariates should be constructed from data in the CDM model. Warning: this function is for advanced users only.

Value

An object of type covariateSettings, to be used in other functions.

Examples

```
analysisDetails <- createAnalysisDetails(
  analysisId = 1,
  sqlFileName = "DemographicsGender.sql",
  parameters = list(
    analysisId = 1,
    analysisName = "Gender",
    domainId = "Demographics"
  ),
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
covSettings <- createDetailedCovariateSettings(analyses = analysisDetails)
```

createDetailedTemporalCovariateSettings
Create detailed temporal covariate settings

Description

Create detailed temporal covariate settings

Usage

```
createDetailedTemporalCovariateSettings(
  analyses = list(),
  temporalStartDays = -365:-1,
  temporalEndDays = -365:-1
)
```

Arguments

analyses	A list of analysis detail objects as created using createAnalysisDetails.
temporalStartDays	A list of integers representing the start of a time period, relative to the index date. 0 indicates the index date, -1 indicates the day before the index date, etc. The start day is included in the time period.
temporalEndDays	A list of integers representing the end of a time period, relative to the index date. 0 indicates the index date, -1 indicates the day before the index date, etc. The end day is included in the time period.

Details

creates an object specifying in detail how temporal covariates should be constructed from data in the CDM model. Warning: this function is for advanced users only.

Value

An object of type covariateSettings, to be used in other functions.

Examples

```
analysisDetails <- createAnalysisDetails(
  analysisId = 1,
  sqlFileName = "DemographicsGender.sql",
  parameters = list(
    analysisId = 1,
    analysisName = "Gender",
    domainId = "Demographics"
  ),
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
```

```
)
covSettings <- createDetailedTemporalCovariateSettings(
  analyses = analysisDetails,
  temporalStartDays = -365:-1,
  temporalEndDays = -365:-1
)
```

---

createEmptyCovariateData *Creates an empty covariate data object*

Description

Creates an empty covariate data object

Usage

```
createEmptyCovariateData(cohortIds, aggregated, temporal)
```

Arguments

cohortIds	For which cohort IDs should the covariate data be created?
aggregated	if the data should be aggregated
temporal	if the data is temporary

Value

an empty object of class CovariateData

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = 1,
  aggregated = FALSE,
  temporal = FALSE
)
```

---

createTable1 *Create a table 1*

Description

Creates a formatted table of cohort characteristics, to be included in publications or reports. Allows for creating a table describing a single cohort, or a table comparing two cohorts.

Usage

```
createTable1(
  covariateData1,
  covariateData2 = NULL,
  cohortId1 = NULL,
  cohortId2 = NULL,
  specifications = getDefaultTable1Specifications(),
  output = "two columns",
  showCounts = FALSE,
  showPercent = TRUE,
  percentDigits = 1,
  valueDigits = 1,
  stdDiffDigits = 2
)
```

Arguments

covariateData1	The covariate data of the cohort to be included in the table.
covariateData2	The covariate data of the cohort to also be included, when comparing two cohorts.
cohortId1	If provided, covariateData1 will be restricted to this cohort. If not provided, covariateData1 is assumed to contain data on only 1 cohort.
cohortId2	If provided, covariateData2 will be restricted to this cohort. If not provided, covariateData2 is assumed to contain data on only 1 cohort.
specifications	Specifications of which covariates to display, and how.
output	The output format for the table. Options are output = "two columns", output = "one column", or output = "list".
showCounts	Show the number of cohort entries having the binary covariate?
showPercent	Show the percentage of cohort entries having the binary covariate?
percentDigits	Number of digits to be used for percentages.
valueDigits	Number of digits to be used for the values of continuous variables.
stdDiffDigits	Number of digits to be used for the standardized differences.

Value

A data frame, or, when output = "list" a list of two data frames.

Examples

```
eunomiaConnectionDetails <- Eunomia::getEunomiaConnectionDetails()
covSettings <- createDefaultCovariateSettings()
Eunomia::createCohorts(
  connectionDetails = eunomiaConnectionDetails,
  cdmDatabaseSchema = "main",
  cohortDatabaseSchema = "main",
  cohortTable = "cohort"
)
covData1 <- getDbCovariateData(
  connectionDetails = eunomiaConnectionDetails,
  tempEmulationSchema = NULL,
```

```
  cdmDatabaseSchema = "main",
  cdmVersion = "5",
  cohortTable = "cohort",
  cohortDatabaseSchema = "main",
  cohortTableIsTemp = FALSE,
  cohortId = 1,
  rowIdField = "subject_id",
  covariateSettings = covSettings,
  aggregated = TRUE
)
covData2 <- getDbCovariateData(
  connectionDetails = eunomiaConnectionDetails,
  tempEmulationSchema = NULL,
  cdmDatabaseSchema = "main",
  cdmVersion = "5",
  cohortTable = "cohort",
  cohortDatabaseSchema = "main",
  cohortTableIsTemp = FALSE,
  cohortId = 2,
  rowIdField = "subject_id",
  covariateSettings = covSettings,
  aggregated = TRUE
)
table1 <- createTable1(
  covariateData1 = covData1,
  covariateData2 = covData2,
  cohortId1 = 1,
  cohortId2 = 2,
  specifications = getDefaultTable1Specifications(),
  output = "one column",
  showCounts = FALSE,
  showPercent = TRUE,
  percentDigits = 1,
  valueDigits = 1,
  stdDiffDigits = 2
)
```

`createTable1CovariateSettings` *Create covariate settings for a table 1*

Description

Creates a covariate settings object for generating only those covariates that will be included in a table 1. This function works by filtering the `covariateSettings` object for the covariates in the `specifications` object.

Usage

```
createTable1CovariateSettings(
  specifications = getDefaultTable1Specifications(),
  covariateSettings = createDefaultCovariateSettings(),
```

```
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

Arguments

specifications	A specifications object for generating a table using the createTable1 function.
covariateSettings	The covariate settings object to use as the basis for the filtered covariate settings.
includedCovariateConceptIds	A list of concept IDs that should be used to construct covariates.
addDescendantsToInclude	Should descendant concept IDs be added to the list of concepts to include?
excludedCovariateConceptIds	A list of concept IDs that should NOT be used to construct covariates.
addDescendantsToExclude	Should descendant concept IDs be added to the list of concepts to exclude?
includedCovariateIds	A list of covariate IDs that should be restricted to.

Value

A covariate settings object, for example to be used when calling the getDbCovariateData function.

Examples

```
table1CovSettings <- createTable1CovariateSettings(
  specifications = getDefaultTable1Specifications(),
  covariateSettings = createDefaultCovariateSettings(),
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

---

createTemporalCovariateSettings

Create covariate settings

Description

Create covariate settings

Usage

```
createTemporalCovariateSettings(
  useDemographicsGender = FALSE,
  useDemographicsAge = FALSE,
  useDemographicsAgeGroup = FALSE,
  useDemographicsRace = FALSE,
  useDemographicsEthnicity = FALSE,
  useDemographicsIndexYear = FALSE,
  useDemographicsIndexMonth = FALSE,
  useDemographicsPriorObservationTime = FALSE,
  useDemographicsPostObservationTime = FALSE,
  useDemographicsTimeInCohort = FALSE,
  useDemographicsIndexYearMonth = FALSE,
  useCareSiteId = FALSE,
  useConditionOccurrence = FALSE,
  useConditionOccurrencePrimaryInpatient = FALSE,
  useConditionEraStart = FALSE,
  useConditionEraOverlap = FALSE,
  useConditionEraGroupStart = FALSE,
  useConditionEraGroupOverlap = FALSE,
  useDrugExposure = FALSE,
  useDrugEraStart = FALSE,
  useDrugEraOverlap = FALSE,
  useDrugEraGroupStart = FALSE,
  useDrugEraGroupOverlap = FALSE,
  useProcedureOccurrence = FALSE,
  useDeviceExposure = FALSE,
  useMeasurement = FALSE,
  useMeasurementValue = FALSE,
  useMeasurementRangeGroup = FALSE,
  useMeasurementValueAsConcept = FALSE,
  useObservation = FALSE,
  useObservationValueAsConcept = FALSE,
  useObservationValue = FALSE,
  useCharlsonIndex = FALSE,
  useDcsi = FALSE,
  useChads2 = FALSE,
  useChads2Vasc = FALSE,
  useHfrs = FALSE,
  useDistinctConditionCount = FALSE,
  useDistinctIngredientCount = FALSE,
  useDistinctProcedureCount = FALSE,
  useDistinctMeasurementCount = FALSE,
  useDistinctObservationCount = FALSE,
  useVisitCount = FALSE,
  useVisitConceptCount = FALSE,
  temporalStartDays = -365:-1,
  temporalEndDays = -365:-1,
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
```

```
  includedCovariateIds = c()
)
```

Arguments

`useDemographicsGender`
: Gender of the subject. (analysis ID 1)

`useDemographicsAge`
: Age of the subject on the index date (in years). (analysis ID 2)

`useDemographicsAgeGroup`
: Age of the subject on the index date (in 5 year age groups) (analysis ID 3)

`useDemographicsRace`
: Race of the subject. (analysis ID 4)

`useDemographicsEthnicity`
: Ethnicity of the subject. (analysis ID 5)

`useDemographicsIndexYear`
: Year of the index date. (analysis ID 6)

`useDemographicsIndexMonth`
: Month of the index date. (analysis ID 7)

`useDemographicsPriorObservationTime`
: Number of days of observation time preceding the index date. (analysis ID 8)

`useDemographicsPostObservationTime`
: Number of days of observation time preceding the index date. (analysis ID 9)

`useDemographicsTimeInCohort`
: Number of days of observation time preceding the index date. (analysis ID 10)

`useDemographicsIndexYearMonth`
: Calendar month of the index date. (analysis ID 11)

`useCareSiteId`
: Care site associated with the cohort start, pulled from the visit_detail, visit_occurrence, or person table, in that order. (analysis ID 12)

`useConditionOccurrence`
: One covariate per condition in the condition_occurrence table starting in the time window. (analysis ID 101)

`useConditionOccurrencePrimaryInpatient`
: One covariate per condition observed as a primary diagnosis in an inpatient setting in the condition_occurrence table starting in the time window. (analysis ID 102)

`useConditionEraStart`
: One covariate per condition in the condition_era table starting in the time window. (analysis ID 201)

`useConditionEraOverlap`
: One covariate per condition in the condition_era table overlapping with any part of the time window. (analysis ID 202)

`useConditionEraGroupStart`
: One covariate per condition era rolled up to SNOMED groups in the condition_era table starting in the time window. (analysis ID 203)

`useConditionEraGroupOverlap`
: One covariate per condition era rolled up to SNOMED groups in the condition_era table overlapping with any part of the time window. (analysis ID 204)

useDrugExposure
: One covariate per drug in the drug_exposure table starting in the time window. (analysis ID 301)

useDrugEraStart
: One covariate per drug in the drug_era table starting in the time window. (analysis ID 401)

useDrugEraOverlap
: One covariate per drug in the drug_era table overlapping with any part of the time window. (analysis ID 402)

useDrugEraGroupStart
: One covariate per drug rolled up to ATC groups in the drug_era table starting in the time window. (analysis ID 403)

useDrugEraGroupOverlap
: One covariate per drug rolled up to ATC groups in the drug_era table overlapping with any part of the time window. (analysis ID 404)

useProcedureOccurrence
: One covariate per procedure in the procedure_occurrence table in the time window. (analysis ID 501)

useDeviceExposure
: One covariate per device in the device exposure table starting in the timewindow. (analysis ID 601)

useMeasurement
: One covariate per measurement in the measurement table in the time window. (analysis ID 701)

useMeasurementValue
: One covariate containing the value per measurement-unit combination in the time window. If multiple values are found, the last is taken. (analysis ID 702)

useMeasurementRangeGroup
: Covariates indicating whether measurements are below, within, or above normal range within the time period. (analysis ID 703)

useMeasurementValueAsConcept
: One covariate per measurement-value concept combination within the time period. (analysis ID 704)

useObservation
: One covariate per observation in the observation table in the time window. (analysis ID 801)

useObservationValueAsConcept
: One covariate per observation-value concept combination within the time period. (analysis ID 802)

useObservationValue
: One covariate containing the value per observation-unit combination in the time window. If multiple values are found, the last is taken. (analysis ID 803)

useCharlsonIndex
: The Charlson comorbidity index (Romano adaptation) using all conditions prior to the window end. (analysis ID 901)

useDcsi
: The Diabetes Comorbidity Severity Index (DCSI) using all conditions prior to the window end. (analysis ID 902)

useChads2
: The CHADS2 score using all conditions prior to the window end. (analysis ID 903)

useChads2Vasc
: The CHADS2VASc score using all conditions prior to the window end. (analysis ID 904)

useHfrs The Hospital Frailty Risk Score score using all conditions prior to the window end. (analysis ID 926)

useDistinctConditionCount
The number of distinct condition concepts observed in the time window. (analysis ID 905)

useDistinctIngredientCount
The number of distinct ingredients observed in the time window. (analysis ID 906)

useDistinctProcedureCount
The number of distinct procedures observed in the time window. (analysis ID 907)

useDistinctMeasurementCount
The number of distinct measurements observed in the time window. (analysis ID 908)

useDistinctObservationCount
The number of distinct observations in the time window. (analysis ID 909)

useVisitCount The number of visits observed in the time window. (analysis ID 910)

useVisitConceptCount
The number of visits observed in the time window, stratified by visit concept ID. (analysis ID 911)

temporalStartDays
A list of integers representing the start of a time period, relative to the index date. 0 indicates the index date, -1 indicates the day before the index date, etc. The start day is included in the time period.

temporalEndDays
A list of integers representing the end of a time period, relative to the index date. 0 indicates the index date, -1 indicates the day before the index date, etc. The end day is included in the time period.

includedCovariateConceptIds
A list of concept IDs that should be used to construct covariates.

addDescendantsToInclude
Should descendant concept IDs be added to the list of concepts to include?

excludedCovariateConceptIds
A list of concept IDs that should NOT be used to construct covariates.

addDescendantsToExclude
Should descendant concept IDs be added to the list of concepts to exclude?

includedCovariateIds
A list of covariate IDs that should be restricted to.

Details

creates an object specifying how covariates should be constructed from data in the CDM model.

Value

An object of type covariateSettings, to be used in other functions.

Examples

```
settings <- createTemporalCovariateSettings(
  useDemographicsGender = TRUE,
  useDemographicsAge = FALSE,
  useDemographicsAgeGroup = TRUE,
  useDemographicsRace = TRUE,
  useDemographicsEthnicity = TRUE,
  useDemographicsIndexYear = TRUE,
  useDemographicsIndexMonth = TRUE,
  useDemographicsPriorObservationTime = FALSE,
  useDemographicsPostObservationTime = FALSE,
  useDemographicsTimeInCohort = FALSE,
  useDemographicsIndexYearMonth = FALSE,
  useCareSiteId = FALSE,
  useConditionOccurrence = FALSE,
  useConditionOccurrencePrimaryInpatient = FALSE,
  useConditionEraStart = FALSE,
  useConditionEraOverlap = FALSE,
  useConditionEraGroupStart = FALSE,
  useConditionEraGroupOverlap = TRUE,
  useDrugExposure = FALSE,
  useDrugEraStart = FALSE,
  useDrugEraOverlap = FALSE,
  useDrugEraGroupStart = FALSE,
  useDrugEraGroupOverlap = TRUE,
  useProcedureOccurrence = TRUE,
  useDeviceExposure = TRUE,
  useMeasurement = TRUE,
  useMeasurementValue = FALSE,
  useMeasurementRangeGroup = TRUE,
  useMeasurementValueAsConcept = TRUE,
  useObservation = TRUE,
  useObservationValueAsConcept = TRUE,
  useObservationValue = FALSE,
  useCharlsonIndex = TRUE,
  useDcsi = TRUE,
  useChads2 = TRUE,
  useChads2Vasc = TRUE,
  useHfrs = FALSE,
  useDistinctConditionCount = FALSE,
  useDistinctIngredientCount = FALSE,
  useDistinctProcedureCount = FALSE,
  useDistinctMeasurementCount = FALSE,
  useDistinctObservationCount = FALSE,
  useVisitCount = FALSE,
  useVisitConceptCount = FALSE,
  temporalStartDays = -365:-1,
  temporalEndDays = -365:-1,
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

createTemporalSequenceCovariateSettings
Create covariate settings

Description

Create covariate settings

Usage

```
createTemporalSequenceCovariateSettings(
  useDemographicsGender = FALSE,
  useDemographicsAge = FALSE,
  useDemographicsAgeGroup = FALSE,
  useDemographicsRace = FALSE,
  useDemographicsEthnicity = FALSE,
  useDemographicsIndexYear = FALSE,
  useDemographicsIndexMonth = FALSE,
  useConditionOccurrence = FALSE,
  useConditionOccurrencePrimaryInpatient = FALSE,
  useConditionEraStart = FALSE,
  useConditionEraGroupStart = FALSE,
  useDrugExposure = FALSE,
  useDrugEraStart = FALSE,
  useDrugEraGroupStart = FALSE,
  useProcedureOccurrence = FALSE,
  useDeviceExposure = FALSE,
  useMeasurement = FALSE,
  useMeasurementValue = FALSE,
  useObservation = FALSE,
  timePart = "month",
  timeInterval = 1,
  sequenceEndDay = -1,
  sequenceStartDay = -730,
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

Arguments

useDemographicsGender
: Gender of the subject. (analysis ID 1)

useDemographicsAge
: Age of the subject on the index date (in years). (analysis ID 2)

useDemographicsAgeGroup
: Age of the subject on the index date (in 5 year age groups) (analysis ID 3)

useDemographicsRace
: Race of the subject. (analysis ID 4)

useDemographicsEthnicity
: Ethnicity of the subject. (analysis ID 5)

useDemographicsIndexYear
: Year of the index date. (analysis ID 6)

useDemographicsIndexMonth
: Month of the index date. (analysis ID 7)

useConditionOccurrence
: One covariate per condition in the condition_occurrence table starting in the time window. (analysis ID 101)

useConditionOccurrencePrimaryInpatient
: One covariate per condition observed as a primary diagnosis in an inpatient setting in the condition_occurrence table starting in the time window. (analysis ID 102)

useConditionEraStart
: One covariate per condition in the condition_era table starting in the time window. (analysis ID 201)

useConditionEraGroupStart
: One covariate per condition era rolled up to SNOMED groups in the condition_era table starting in the time window. (analysis ID 203)

useDrugExposure
: One covariate per drug in the drug_exposure table starting in the time window. (analysis ID 301)

useDrugEraStart
: One covariate per drug in the drug_era table starting in the time window. (analysis ID 401)

useDrugEraGroupStart
: One covariate per drug rolled up to ATC groups in the drug_era table starting in the time window. (analysis ID 403)

useProcedureOccurrence
: One covariate per procedure in the procedure_occurrence table in the time window. (analysis ID 501)

useDeviceExposure
: One covariate per device in the device exposure table starting in the timewindow. (analysis ID 601)

useMeasurement
: One covariate per measurement in the measurement table in the time window. (analysis ID 701)

useMeasurementValue
: One covariate containing the value per measurement-unit combination in the time window. If multiple values are found, the last is taken. (analysis ID 702)

useObservation
: One covariate per observation in the observation table in the time window. (analysis ID 801)

timePart
: The interval scale ('DAY', 'MONTH', 'YEAR')

timeInterval
: Fixed interval length for timeId using the 'timePart' scale. For example, a 'timePart' of DAY with 'timeInterval' 30 has timeIds where timeId 1 is day 0 to day 29, timeId 2 is day 30 to day 59, etc.

sequenceEndDay
: What is the end day (relative to the index date) of the data extraction?

sequenceStartDay
: What is the start day (relative to the index date) of the data extraction?

`includedCovariateConceptIds`
: A list of concept IDs that should be used to construct covariates.

`addDescendantsToInclude`
: Should descendant concept IDs be added to the list of concepts to include?

`excludedCovariateConceptIds`
: A list of concept IDs that should NOT be used to construct covariates.

`addDescendantsToExclude`
: Should descendant concept IDs be added to the list of concepts to exclude?

`includedCovariateIds`
: A list of covariate IDs that should be restricted to.

Details

creates an object specifying how covariates should be constructed from data in the CDM model.

Value

An object of type `covariateSettings`, to be used in other functions.

Examples

```
settings <- createTemporalSequenceCovariateSettings(
  useDemographicsGender = TRUE,
  useDemographicsAge = FALSE,
  useDemographicsAgeGroup = TRUE,
  useDemographicsRace = TRUE,
  useDemographicsEthnicity = TRUE,
  useDemographicsIndexYear = TRUE,
  useDemographicsIndexMonth = TRUE,
  useConditionOccurrence = FALSE,
  useConditionOccurrencePrimaryInpatient = FALSE,
  useConditionEraStart = FALSE,
  useConditionEraGroupStart = FALSE,
  useDrugExposure = FALSE,
  useDrugEraStart = FALSE,
  useDrugEraGroupStart = FALSE,
  useProcedureOccurrence = TRUE,
  useDeviceExposure = TRUE,
  useMeasurement = TRUE,
  useMeasurementValue = FALSE,
  useObservation = TRUE,
  timePart = "DAY",
  timeInterval = 1,
  sequenceEndDay = -1,
  sequenceStartDay = -730,
  includedCovariateConceptIds = c(),
  addDescendantsToInclude = FALSE,
  excludedCovariateConceptIds = c(),
  addDescendantsToExclude = FALSE,
  includedCovariateIds = c()
)
```

filterByCohortDefinitionId *Filter covariates by cohort definition IDs*

Description

Filter covariates by cohort definition IDs

Usage

```
filterByCohortDefinitionId(covariateData, cohortId = 1, cohortIds = c(1))
```

Arguments

covariateData	An object of type CovariateData
cohortId	DEPRECATED The cohort definition IDs to keep.
cohortIds	The cohort definition IDs to keep.

Value

An object of type covariateData.

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = c(1, 2),
  aggregated = TRUE,
  temporal = FALSE
)

covData <- filterByCohortDefinitionId(
  covariateData = covariateData,
  cohortIds = c(1)
)
```

filterByRowId *Filter covariates by row ID*

Description

Filter covariates by row ID

Usage

```
filterByRowId(covariateData, rowIds)
```

Arguments

covariateData	An object of type CovariateData
rowIds	A vector containing the rowIds to keep.

Value

An object of type covariateData.

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = 1,
  aggregated = FALSE,
  temporal = FALSE
)

covData <- filterByRowId(
  covariateData = covariateData,
  rowIds = 1
)
```

---

getDbCohortAttrCovariatesData

Getcovariate information from the database through the cohort_attribute table

---

Description

Constructs covariates using the cohort_attribute table.

Usage

```
getDbCohortAttrCovariatesData(
  connection,
  oracleTempSchema = NULL,
  cdmDatabaseSchema,
  cohortTable = "#cohort_person",
  cohortId = -1,
  cohortIds = c(-1),
  cdmVersion = "5",
  rowIdField = "subject_id",
  covariateSettings,
  targetDatabaseSchema = NULL,
  targetCovariateTable = NULL,
  targetCovariateContinuousTable = NULL,
  targetCovariateRefTable = NULL,
  targetAnalysisRefTable = NULL,
  targetTimeRefTable = NULL,
  aggregated = FALSE,
```

```
  minCharacterizationMean = 0,
  minCharacterizationCount = 0,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema")
)
```

Arguments

connection	A connection to the server containing the schema as created using the connect function in the DatabaseConnector package.
oracleTempSchema	DEPRECATED: use tempEmulationSchema instead.
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'.
cohortTable	Name of the table holding the cohort for which we want to construct covariates. If it is a temp table, the name should have a hash prefix, e.g. '#temp_table'. If it is a non-temp table, it should include the database schema, e.g. 'cdm_database.cohort'.
cohortId	DEPRECATED:For which cohort ID should covariates be constructed? If set to -1, covariates will be constructed for all cohorts in the specified cohort table.
cohortIds	For which cohort ID(s) should covariates be constructed? If set to c(-1), covariates will be constructed for all cohorts in the specified cohort table.
cdmVersion	The version of the Common Data Model used. Currently only cdmVersion = "5" is supported.
rowIdField	The name of the field in the cohort temp table that is to be used as the row_id field in the output table. This can be especially usefull if there is more than one period per person.
covariateSettings	An object of type covariateSettings as created using the createCohortAttrCovariateSettings function.
targetDatabaseSchema	Optional target database schema. Ignored because cohort attribute covariates do not support exporting directly to target tables.
targetCovariateTable	Optional target covariate table. Ignored because cohort attribute covariates do not support exporting directly to target tables.
targetCovariateContinuousTable	Optional target continuous covariate table. Ignored because cohort attribute covariates do not support exporting directly to target tables.
targetCovariateRefTable	Optional target covariate reference table. Ignored because cohort attribute covariates do not support exporting directly to target tables.
targetAnalysisRefTable	Optional target analysis reference table. Ignored because cohort attribute covariates do not support exporting directly to target tables.
targetTimeRefTable	Optional target time reference table. Ignored because cohort attribute covariates do not support exporting directly to target tables.
aggregated	Should aggregate statistics be computed instead of covariates per cohort entry?

`minCharacterizationMean`
: The minimum mean value for characterization output. Ignored because aggregation is not supported for cohort attribute covariates.

`minCharacterizationCount`
: The minimum count value for characterization output. Ignored because aggregation is not supported for cohort attribute covariates.

`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.

Details

This function uses the data in the CDM to construct a large set of covariates for the provided cohort. The cohort is assumed to be in an existing temp table with these fields: 'subject_id', 'cohort_definition_id', 'cohort_start_date'. Optionally, an extra field can be added containing the unique identifier that will be used as rowID in the output. Typically, users don't call this function directly but rather use the `getDbCovariateData` function instead.

Value

Returns an object of type `CovariateData`, which is an Andromeda object containing information on the baseline covariates. Information about multiple outcomes can be captured at once for efficiency reasons. This object is a list with the following components:

covariates An ffdf object listing the baseline covariates per person in the cohorts. This is done using a sparse representation: covariates with a value of 0 are omitted to save space. The covariates object will have three columns: rowId, covariateId, and covariateValue. The rowId is usually equal to the person_id, unless specified otherwise in the rowIdField argument.

covariateRef A table describing the covariates that have been extracted.

. The CovariateData object will also have a `metaData` attribute, a list of objects with information on how the covariateData object was constructed.

Examples

```
connectionDetails <- Eunomia::getEunomiaConnectionDetails()
Eunomia::createCohorts(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = "main",
  cohortDatabaseSchema = "main",
  cohortTable = "cohort"
)
connection <- DatabaseConnector::connect(connectionDetails)
covariateSettings <- createCohortAttrCovariateSettings(
  attrDatabaseSchema = "main",
  cohortAttrTable = "cohort_attribute",
  attrDefinitionTable = "attribute_definition",
  includeAttrIds = c(1),
  isBinary = FALSE,
  missingMeansZero = FALSE
)

covData <- getDbCohortAttrCovariatesData(
  connection = connection,
```

```
  tempEmulationSchema = NULL,
  cdmDatabaseSchema = "main",
  cdmVersion = "5",
  cohortTable = "cohort",
  cohortIds = 1,
  rowIdField = "subject_id",
  covariateSettings = covariateSettings,
  aggregated = FALSE
)
```

`getDbCohortBasedCovariatesData` *Get covariate information from the database based on other cohorts*

Description

Constructs covariates using other cohorts.

Usage

```
getDbCohortBasedCovariatesData(
  connection,
  oracleTempSchema = NULL,
  cdmDatabaseSchema,
  cohortTable = "#cohort_person",
  cohortId = -1,
  cohortIds = c(-1),
  cdmVersion = "5",
  rowIdField = "subject_id",
  covariateSettings,
  targetDatabaseSchema = NULL,
  targetCovariateTable = NULL,
  targetCovariateContinuousTable = NULL,
  targetCovariateRefTable = NULL,
  targetAnalysisRefTable = NULL,
  targetTimeRefTable = NULL,
  aggregated = FALSE,
  minCharacterizationMean = 0,
  minCharacterizationCount = 0,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema")
)
```

Arguments

`connection`	A connection to the server containing the schema as created using the `connect` function in the `DatabaseConnector` package.
`oracleTempSchema`	DEPRECATED: use `tempEmulationSchema` instead.

`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'.

`cohortTable`
: Name of the table holding the cohort for which we want to construct covariates. If it is a temp table, the name should have a hash prefix, e.g. '#temp_table'. If it is a non-temp table, it should include the database schema, e.g. 'cdm_database.cohort'.

`cohortId`
: DEPRECATED:For which cohort ID should covariates be constructed? If set to -1, covariates will be constructed for all cohorts in the specified cohort table.

`cohortIds`
: For which cohort ID(s) should covariates be constructed? If set to c(-1), covariates will be constructed for all cohorts in the specified cohort table.

`cdmVersion`
: The version of the Common Data Model used. Currently only `cdmVersion = "5"` is supported.

`rowIdField`
: The name of the field in the cohort temp table that is to be used as the row_id field in the output table. This can be especially usefull if there is more than one period per person.

`covariateSettings`
: An object of type `covariateSettings` as created using the `createCohortBasedCovariateSetting` or `createCohortBasedTemporalCovariateSettings` functions.

`targetDatabaseSchema`
: (Optional) The name of the database schema where the resulting covariates should be stored. If not provided, results will be fetched to R.

`targetCovariateTable`
: (Optional) The name of the table where the resulting covariates will be stored. If not provided, results will be fetched to R. The table can be a permanent table in the `targetDatabaseSchema` or a temp table. If it is a temp table, do not specify `targetDatabaseSchema`.

`targetCovariateContinuousTable`
: (Optional) The name of the table where the resulting continuous covariates should be stored.

`targetCovariateRefTable`
: (Optional) The name of the table where the covariate reference will be stored.

`targetAnalysisRefTable`
: (Optional) The name of the table where the analysis reference will be stored.

`targetTimeRefTable`
: (Optional) The name of the table for the time reference

`aggregated`
: Should aggregate statistics be computed instead of covariates per cohort entry?

`minCharacterizationMean`
: The minimum mean value for binary characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that have very low values. The default is 0.

`minCharacterizationCount`
: The minimum count value for binary characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that occur in very few cohort entries. The default is 0.

`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.

Details

This function uses the data in the CDM to construct a large set of covariates for the provided cohort. The cohort is assumed to be in an existing temp table with these fields: 'subject_id', 'cohort_definition_id', 'cohort_start_date'. Optionally, an extra field can be added containing the unique identifier that will be used as rowID in the output. Typically, users don't call this function directly but rather use the `getDbCovariateData` function instead.

Value

Returns an object of type `CovariateData`, which is an Andromeda object containing information on the baseline covariates. Information about multiple outcomes can be captured at once for efficiency reasons. This object is a list with the following components:

covariates An ffdf object listing the baseline covariates per person in the cohorts. This is done using a sparse representation: covariates with a value of 0 are omitted to save space. The covariates object will have three columns: rowId, covariateId, and covariateValue. The rowId is usually equal to the person_id, unless specified otherwise in the rowIdField argument.

covariateRef A table describing the covariates that have been extracted.

. The CovariateData object will also have a `metaData` attribute, a list of objects with information on how the covariateData object was constructed.

`getDbCovariateData` *Get covariate information from the database*

Description

Uses one or several covariate builder functions to construct covariates.

Usage

```
getDbCovariateData(
  connectionDetails = NULL,
  connection = NULL,
  oracleTempSchema = NULL,
  cdmDatabaseSchema,
  cdmVersion = "5",
  cohortTable = "cohort",
  cohortDatabaseSchema = cdmDatabaseSchema,
  cohortTableIsTemp = FALSE,
  cohortId = -1,
  cohortIds = c(-1),
  rowIdField = "subject_id",
  covariateSettings,
  exportToTable = FALSE,
  createTable = exportToTable,
  dropTableIfExists = exportToTable,
  targetDatabaseSchema = NULL,
  targetCovariateTable = NULL,
  targetCovariateContinuousTable = NULL,
  targetCovariateRefTable = NULL,
```

```
  targetAnalysisRefTable = NULL,
  targetTimeRefTable = NULL,
  aggregated = FALSE,
  minCharacterizationMean = 0,
  minCharacterizationCount = 0,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema"),
  covariateCohortDatabaseSchema = NULL,
  covariateCohortTable = NULL
)
```

Arguments

connectionDetails	An R object of type connectionDetails created using the function createConnectionDetails in the DatabaseConnector package. Either the connection or connectionDetails argument should be specified.
connection	A connection to the server containing the schema as created using the connect function in the DatabaseConnector package. Either the connection or connectionDetails argument should be specified.
oracleTempSchema	DEPRECATED: use tempEmulationSchema instead.
cdmDatabaseSchema	The name of the database schema that contains the OMOP CDM instance. Requires read permissions to this database. On SQL Server, this should specify both the database and the schema, so for example 'cdm_instance.dbo'.
cdmVersion	Define the OMOP CDM version used: currently supported is "5".
cohortTable	Name of the (temp) table holding the cohort for which we want to construct covariates
cohortDatabaseSchema	If the cohort table is not a temp table, specify the database schema where the cohort table can be found. On SQL Server, this should specify both the database and the schema, so for example 'cdm_instance.dbo'.
cohortTableIsTemp	Is the cohort table a temp table?
cohortId	DEPRECATED:For which cohort ID(s) should covariates be constructed? If set to -1, covariates will be constructed for all cohorts in the specified cohort table.
cohortIds	For which cohort ID(s) should covariates be constructed? If set to c(-1), covariates will be constructed for all cohorts in the specified cohort table.
rowIdField	The name of the field in the cohort table that is to be used as the row_id field in the output table. This can be especially usefull if there is more than one period per person.
covariateSettings	Either an object of type covariateSettings as created using one of the createCovariate functions, or a list of such objects.
exportToTable	Whether to export to a table rather than Andromeda object
createTable	Run sql to create table? Code does not check if table exists.
dropTableIfExists	If targetDatabaseSchema, drop any existing tables. Otherwise, results are merged into existing table data. Overides createTable.

`targetDatabaseSchema`
: (Optional) The name of the database schema where the resulting covariates should be stored as a table. If not provided, results will be fetched to R.

`targetCovariateTable`
: (Optional) The name of the table where the resulting covariates will be stored. If not provided, results will be fetched to R. The table can be a permanent table in the `targetDatabaseSchema` or a temp table. If it is a temp table, do not specify `targetDatabaseSchema`.

`targetCovariateContinuousTable`
: (Optional) The name of the table where the resulting continuous covariates will be stored. If not provided, results will be fetched to R. The table can be a permanent table in the `targetDatabaseSchema` or a temp table. If it is a temp table, do not specify `targetDatabaseSchema`.

`targetCovariateRefTable`
: (Optional) The name of the table where the covariate reference will be stored. If it is a temp table, do not specify `targetDatabaseSchema`.

`targetAnalysisRefTable`
: (Optional) The name of the table where the analysis reference will be stored. If it is a temp table, do not specify `targetDatabaseSchema`.

`targetTimeRefTable`
: (Optional) The name of the table for the time reference. If it is a temp table, do not specify `targetDatabaseSchema`.

`aggregated`
: Should aggregate statistics be computed instead of covariates per cohort entry? If aggregated is set to FALSE, the results returned will be based on each subject_id and cohort_start_date in your cohort table. If your cohort contains multiple entries for the same subject_id (due to different cohort_start_date values), you must carefully set the rowIdField so you can identify the patients properly. See issue #229 for more discussion on this parameter.

`minCharacterizationMean`
: The minimum mean value for characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that have very low values. The default is 0.

`minCharacterizationCount`
: The minimum count value for characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that occur in very few cohort entries. The default is 0.

`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.

`covariateCohortDatabaseSchema`
: The database schema where the cohorts used to define the covariates can be found.

`covariateCohortTable`
: The table where the cohorts used to define the covariates can be found.

Details

This function uses the data in the CDM to construct a large set of covariates for the provided cohort. The cohort is assumed to be in an existing table with these fields: 'subject_id', 'co-

hort_definition_id', 'cohort_start_date'. Optionally, an extra field can be added containing the unique identifier that will be used as rowID in the output.

Value

Returns an object of type `covariateData`, containing information on the covariates.

Examples

```
eunomiaConnectionDetails <- Eunomia::getEunomiaConnectionDetails()
covSettings <- createDefaultCovariateSettings()
Eunomia::createCohorts(
  connectionDetails = eunomiaConnectionDetails,
  cdmDatabaseSchema = "main",
  cohortDatabaseSchema = "main",
  cohortTable = "cohort"
)
covData <- getDbCovariateData(
  connectionDetails = eunomiaConnectionDetails,
  tempEmulationSchema = NULL,
  cdmDatabaseSchema = "main",
  cdmVersion = "5",
  cohortTable = "cohort",
  cohortDatabaseSchema = "main",
  cohortTableIsTemp = FALSE,
  cohortIds = -1,
  rowIdField = "subject_id",
  covariateSettings = covSettings,
  aggregated = FALSE
)
```

---

getDbDefaultCovariateData

Get default covariate information from the database

Description

Constructs a large default set of covariates for one or more cohorts using data in the CDM schema. Includes covariates for all drugs, drug classes, condition, condition classes, procedures, observations, etc.

Usage

```
getDbDefaultCovariateData(
  connection,
  oracleTempSchema = NULL,
  cdmDatabaseSchema,
  cohortTable = "#cohort_person",
  cohortId = -1,
  cohortIds = c(-1),
  cdmVersion = "5",
```

```
  rowIdField = "subject_id",
  covariateSettings,
  targetDatabaseSchema = NULL,
  targetCovariateTable = NULL,
  targetCovariateContinuousTable = NULL,
  targetCovariateRefTable = NULL,
  targetAnalysisRefTable = NULL,
  targetTimeRefTable = NULL,
  aggregated = FALSE,
  minCharacterizationMean = 0,
  minCharacterizationCount = 0,
  tempEmulationSchema = getOption("sqlRenderTempEmulationSchema")
)
```

Arguments

connection	A connection to the server containing the schema as created using the connect function in the DatabaseConnector package.
oracleTempSchema	DEPRECATED: use tempEmulationSchema instead.
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'.
cohortTable	Name of the table holding the cohort for which we want to construct covariates. If it is a temp table, the name should have a hash prefix, e.g. '#temp_table'. If it is a non-temp table, it should include the database schema, e.g. 'cdm_database.cohort'.
cohortId	DEPRECATED:For which cohort ID should covariates be constructed? If set to -1, covariates will be constructed for all cohorts in the specified cohort table.
cohortIds	For which cohort ID(s) should covariates be constructed? If set to c(-1), covariates will be constructed for all cohorts in the specified cohort table.
cdmVersion	The version of the Common Data Model used. Currently only cdmVersion = "5" is supported.
rowIdField	The name of the field in the cohort temp table that is to be used as the row_id field in the output table. This can be especially usefull if there is more than one period per person.
covariateSettings	Either an object of type covariateSettings as created using one of the createCovariate functions, or a list of such objects.
targetDatabaseSchema	(Optional) The name of the database schema where the resulting covariates should be stored. If not provided, results will be fetched to R.
targetCovariateTable	(Optional) The name of the table where the resulting covariates will be stored. If not provided, results will be fetched to R. The table can be a permanent table in the targetDatabaseSchema or a temp table. If it is a temp table, do not specify targetDatabaseSchema.
targetCovariateContinuousTable	(Optional) The name of the table where the resulting continuous covariates should be stored.

`targetCovariateRefTable`
: (Optional) The name of the table where the covariate reference will be stored.

`targetAnalysisRefTable`
: (Optional) The name of the table where the analysis reference will be stored.

`targetTimeRefTable`
: (Optional) The name of the table for the time reference

`aggregated`
: Should aggregate statistics be computed instead of covariates per cohort entry?

`minCharacterizationMean`
: The minimum mean value for binary characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that have very low values. The default is 0.

`minCharacterizationCount`
: The minimum count value for binary characterization output. Values below this will be cut off from output. This will help reduce the file size of the characterization output, but will remove information on covariates that occur in very few cohort entries. The default is 0.

`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.

Details

This function uses the data in the CDM to construct a large set of covariates for the provided cohort. The cohort is assumed to be in an existing temp table with these fields: 'subject_id', 'cohort_definition_id', 'cohort_start_date'. Optionally, an extra field can be added containing the unique identifier that will be used as rowID in the output. Typically, users don't call this function directly but rather use the `getDbCovariateData` function instead.

Value

Returns an object of type `CovariateData`, which is an Andromeda object containing information on the baseline covariates. Information about multiple outcomes can be captured at once for efficiency reasons. This object is a list with the following components:

covariates An ffdf object listing the baseline covariates per person in the cohorts. This is done using a sparse representation: covariates with a value of 0 are omitted to save space. The covariates object will have three columns: rowId, covariateId, and covariateValue. The rowId is usually equal to the person_id, unless specified otherwise in the rowIdField argument.

covariateRef A table describing the covariates that have been extracted.

. The CovariateData object will also have a `metaData` attribute, a list of objects with information on how the covariateData object was constructed.

Examples

```
connectionDetails <- Eunomia::getEunomiaConnectionDetails()
Eunomia::createCohorts(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = "main",
  cohortDatabaseSchema = "main",
  cohortTable = "cohort"
```

```
)
connection <- DatabaseConnector::connect(connectionDetails)

results <- getDbDefaultCovariateData(
  connection = connection,
  cdmDatabaseSchema = "main",
  cohortTable = "cohort",
  covariateSettings = createDefaultCovariateSettings()
)
```

---

getDefaultTable1Specifications

Get the default table 1 specifications

Description

Loads the default specifications for a table 1, to be used with the `createTable1` function.

Usage

```
getDefaultTable1Specifications()
```

Value

A specifications objects.

Examples

```
defaultTable1Specs <- getDefaultTable1Specifications()
```

---

isAggregatedCovariateData

Check whether covariate data is aggregated

Description

Check whether covariate data is aggregated

Usage

```
isAggregatedCovariateData(x)
```

Arguments

x	The covariate data object to check.

Value

A logical value.

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = 1,
  aggregated = FALSE,
  temporal = FALSE
)
isAggrCovData <- isAggregatedCovariateData(covariateData)
```

---

isCovariateData *Check whether an object is a CovariateData object*

Description

Check whether an object is a CovariateData object

Usage

```
isCovariateData(x)
```

Arguments

x	The object to check.

Value

A logical value.

Examples

```
binaryCovDataFile <- system.file("testdata/binaryCovariateData.zip",
  package = "FeatureExtraction"
)
covData <- loadCovariateData(binaryCovDataFile)
isCovData <- isCovariateData(covData)
```

---

isTemporalCovariateData *Check whether covariate data is temporal*

Description

Check whether covariate data is temporal

Usage

```
isTemporalCovariateData(x)
```

Arguments

x	The covariate data object to check.

Value

A logical value.

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = 1,
  aggregated = FALSE,
  temporal = FALSE
)
isTempCovData <- isTemporalCovariateData(covariateData)
```

---

loadCovariateData *Load the covariate data from a folder*

Description

loadCovariateData loads an object of type covariateData from a folder in the file system.

Usage

```
loadCovariateData(file, readOnly)
```

Arguments

file	The name of the folder containing the data.
readOnly	DEPRECATED: 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 CovariateData.

Examples

```
binaryCovDataFile <- system.file("testdata/binaryCovariateData.zip",
  package = "FeatureExtraction"
)
covData <- loadCovariateData(binaryCovDataFile)
```

replaceCovariateSettingsCohortSchemaTable *Utility function to set the cohort table & schema on createCohortBasedCovariateSettings with information from the execution settings*

Description

Utility function to set the cohort table & schema on createCohortBasedCovariateSettings with information from the execution settings

Usage

```
replaceCovariateSettingsCohortSchemaTable(
  covariateSettings,
  covariateCohortDatabaseSchema,
  covariateCohortTable
)
```

Arguments

covariateSettings	An object of type covariateSettings
covariateCohortDatabaseSchema	The database schema where the cohorts used to define the covariates can be found.
covariateCohortTable	The table where the cohorts used to define the covariates can be found.

Value

An object of type covariateSettings

saveCovariateData *Save the covariate data to folder*

Description

saveCovariateData saves an object of type covariateData to folder.

Usage

```
saveCovariateData(covariateData, file)
```

Arguments

covariateData	An object of type covariateData as generated using getDbCovariateData.
file	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 folder specified by the user.

Value

No return value, called for side effects.

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = 1,
  aggregated = FALSE,
  temporal = FALSE
)
# For this example we'll use a temporary file location:
fileName <- tempfile()
saveCovariateData(covariateData = covariateData, file = fileName)
# Cleaning up the file used in this example:
unlink(fileName)
```

---

tidyCovariateData *Tidy covariate data*

Description

Tidy covariate data

Usage

```
tidyCovariateData(
  covariateData,
  minFraction = 0.001,
  normalize = TRUE,
  removeRedundancy = TRUE
)
```

Arguments

covariateData	An object as generated using the getDbCovariateData function.
minFraction	Minimum fraction of the population that should have a non-zero value for a covariate for that covariate to be kept. Set to 0 to don't filter on frequency.
normalize	Normalize the covariates? (dividing by the max).
removeRedundancy	Should redundant covariates be removed?

Details

Normalize covariate values by dividing by the max and/or remove redundant covariates and/or remove infrequent covariates. For temporal covariates, redundancy is evaluated per time ID.

Value

An object of class `CovariateData`.

Examples

```
covariateData <- FeatureExtraction::createEmptyCovariateData(
  cohortIds = 1,
  aggregated = FALSE,
  temporal = FALSE
)

covData <- tidyCovariateData(
  covariateData = covariateData,
  minFraction = 0.001,
  normalize = TRUE,
  removeRedundancy = TRUE
)
```

Index

.createLooCovariateSettings, 3
.getDbLooCovariateData, 3

aggregateCovariates, 4
Andromeda, *7*

computeStandardizedDifference, 5
convertPrespecSettingsToDetailedSettings, 6
CovariateData *(*CovariateData-class*)*, 7
CovariateData-class, 7
createAnalysisDetails, 7, *27, 28*
createCohortAttrCovariateSettings, 8, *43*
createCohortBasedCovariateSettings, 10, *46*
createCohortBasedTemporalCovariateSettings, 11, *46*
createCovariateSettings, *6*, 12
createDefaultCovariateSettings, 25
createDefaultTemporalCovariateSettings, 26
createDetailedCovariateSettings, *8*, 27
createDetailedTemporalCovariateSettings, *8*, 28
createEmptyCovariateData, 29
createTable1, 29, *32*, *53*
createTable1CovariateSettings, 31
createTemporalCovariateSettings, 32
createTemporalSequenceCovariateSettings, 38

filterByCohortDefinitionId, 41
filterByRowId, 41

getDbCohortAttrCovariatesData, 42
getDbCohortBasedCovariatesData, 45
getDbCovariateData, *7*, *32*, *44*, *47*, 47, *52*, *57*
getDbDefaultCovariateData, 50
getDefaultTable1Specifications, 53

isAggregatedCovariateData, *7*, 53
isCovariateData, *7*, 54
isTemporalCovariateData, *7*, 54

loadCovariateData, *7*, 55

replaceCovariateSettingsCohortSchemaTable, 56

saveCovariateData, *7*, 56
show,CovariateData-method *(*CovariateData-class*)*, 7
summary,CovariateData-method *(*CovariateData-class*)*, 7

tidyCovariateData, 57