Package 'FeatureExtraction'

August 28, 2017

Type Package

Title Generating Features for a Cohort

Version 2.0.0

Date 2017-06-01

Author Martijn J. Schuemie [aut, cre],
Marc A. Suchard [aut],
Patrick B. Ryan [aut],
Jenna Reps [aut]

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

Description An R package for generating features (covariates) for a cohort using data in the Common Data Model.

License Apache License 2.0

Depends R (>= 3.2.2),
DatabaseConnector (>= 2.0.0),

Imports bit,
ff,
ffbase (>= 0.12.1),
plyr,
Rcpp (>= 0.11.2),
rJava,
jsonlite,
SqlRender (>= 1.1.3),

Suggests testthat,
knitr,
rmarkdown

LinkingTo Rcpp

NeedsCompilation yes

RoxygenNote 6.0.1

R topics documented:

`byMaxFf` *Compute max of values binned by a second variable*

Description

Compute max of values binned by a second variable

Usage

```
byMaxFf(values, bins)
```

Arguments

`values`	An ff object containing the numeric values to take the max of.
`bins`	An ff object containing the numeric values to bin by.

Examples

```
values <- ff::as.ff(c(1, 1, 2, 2, 1))
bins <- ff::as.ff(c(1, 1, 1, 2, 2))
byMaxFf(values, bins)
```

bySumFf *Compute sum of values binned by a second variable*

Description

Compute sum of values binned by a second variable

Usage

```
bySumFf(values, bins)
```

Arguments

values	An ff object containing the numeric values to take the sum of.
bins	An ff object containing the numeric values to bin by.

Examples

```
values <- ff::as.ff(c(1, 1, 2, 2, 1))
bins <- ff::as.ff(c(1, 1, 1, 2, 2))
bySumFf(values, bins)
```

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.

`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.
`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.
`sqFileName`	The name of the paramterized SQL file embedded in the `featureExtraction` package.

Details

creates an object specifying in detail how covariates should be contructed 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(attrDatabaseSchema,
  attrDefinitionTable = "attribute_definition",
  cohortAttrTable = "cohort_attribute", includeAttrIds = c())
```

Arguments

`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.

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. On CDM v4, this field should be called `cohort_concept_id`.

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.

createCovariateSettings *Create covariate settings*

Description

Create covariate settings

Usage

```
createCovariateSettings(useDemographicsGender = FALSE,
  useDemographicsAge = FALSE, useDemographicsIndexYear = FALSE,
  useDemographicsIndexMonth = FALSE, useConditionOccurrenceLongTerm = FALSE,
  useConditionOccurrenceShortTerm = FALSE, useConditionEraLongTerm = FALSE,
  useConditionEraShortTerm = FALSE, useConditionGroupEraShortTerm = FALSE,
  useConditionGroupEraLongTerm = FALSE, useDrugExposureLongTerm = FALSE,
  useDrugExposureShortTerm = FALSE, useDrugEraLongTerm = FALSE,
  useDrugEraShortTerm = FALSE, useDrugGroupEraLongTerm = FALSE,
  useDrugGroupEraShortTerm = FALSE, useProcedureOccurrenceLongTerm = FALSE,
  useProcedureOccurrenceShortTerm = FALSE,
  useDeviceExposureLongTerm = FALSE, useDeviceExposureShortTerm = FALSE,
  useMeasurementLongTerm = FALSE, useMeasurementShortTerm = FALSE,
  useObservationLongTerm = FALSE, useObservationShortTerm = FALSE,
  useCharlsonIndex = 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.
useDemographicsAge	Age of the subject on the index date (in 5 year increments).
useDemographicsIndexYear	Year of the index date.
useDemographicsIndexMonth	Month of the index date.
useConditionOccurrenceLongTerm	One covariate per condition in the condition_occurrence table starting in the long term window.
useConditionOccurrenceShortTerm	One covariate per condition in the condition_occurrence table starting in the short term window.
useConditionEraLongTerm	One covariate per condition in the condition_era table overlapping with any part of the long term window.
useConditionEraShortTerm	One covariate per condition in the condition_era table overlapping with any part of the short term window.

useConditionGroupEraShortTerm
: One covariate per condition era rolled up to SNOMED groups in the condition_era table overlapping with any part of the long term window.

useConditionGroupEraLongTerm
: One covariate per condition era rolled up to SNOMED groups in the condition_era table overlapping with any part of the short term window.

useDrugExposureLongTerm
: One covariate per drug in the drug_exposure table starting in the long term window.

useDrugExposureShortTerm
: One covariate per drug in the drug_exposure table starting in the short term window.

useDrugEraLongTerm
: One covariate per drug in the drug_era table overlapping with any part of the long term window.

useDrugEraShortTerm
: One covariate per drug in the drug_era table overlapping with any part of the long short window.

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

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

useProcedureOccurrenceLongTerm
: One covariate per procedure in the procedure_occurrence table in the long term window.

useProcedureOccurrenceShortTerm
: One covariate per procedure in the procedure_occurrence table in the short term window.

useDeviceExposureLongTerm
: One covariate per device in the device exposure table starting in the long term window.

useDeviceExposureShortTerm
: One covariate per device in the device exposure table starting in the short term window.

useMeasurementLongTerm
: One covariate per measurement in the measurement table in the long term window.

useMeasurementShortTerm
: One covariate per measurement in the measurement table in the short term window.

useObservationLongTerm
: One covariate per observation in the observation table in the long term window.

useObservationShortTerm
: One covariate per observation in the observation table in the short term window.

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

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 contructed from data in the CDM model.

Value

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

`createDefaultCovariateSettings` *Create default covariate settings*

Description

Create default covariate settings

Usage

```
createDefaultCovariateSettings()
```

Value

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

createDefaultTemporalCovariateSettings
Create default covariate settings

Description

Create default covariate settings

Usage

```
createDefaultTemporalCovariateSettings()
```

Value

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

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 contructed 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.

`createDetailedTemporalCovariateSettings` *Create detailed temporal covariate settings*

Description

Create detailed temporal covariate settings

Usage

```
createDetailedTemporalCovariateSettings(analyses = list(),
  temporalStartDays = c(-365, -364, -363, -362, -361, -360, -359, -358, -357,
  -356, -355, -354, -353, -352, -351, -350, -349, -348, -347, -346, -345, -344,
  -343, -342, -341, -340, -339, -338, -337, -336, -335, -334, -333, -332, -331,
  -330, -329, -328, -327, -326, -325, -324, -323, -322, -321, -320, -319, -318,
  -317, -316, -315, -314, -313, -312, -311, -310, -309, -308, -307, -306, -305,
  -304, -303, -302, -301, -300, -299, -298, -297, -296, -295, -294, -293, -292,
  -291, -290, -289, -288, -287, -286, -285, -284, -283, -282,      -281, -280,
  -279, -278, -277, -276, -275, -274, -273, -272, -271, -270, -269, -268, -267,
  -266, -265, -264, -263, -262, -261, -260, -259, -258, -257, -256, -255, -254,
  -253, -252, -251, -250, -249, -248, -247, -246, -245, -244, -243, -242, -241,
  -240, -239, -238, -237, -236, -235, -234, -233, -232, -231, -230, -229, -228,
  -227, -226, -225, -224, -223, -222, -221, -220, -219, -218, -217, -216, -215,
  -214, -213, -212, -211, -210, -209, -208, -207, -206, -205, -204, -203, -202,
  -201, -200, -199,      -198, -197, -196, -195, -194, -193, -192, -191, -190,
  -189, -188, -187, -186, -185, -184, -183, -182, -181, -180, -179, -178, -177,
  -176, -175, -174, -173, -172, -171, -170, -169, -168, -167, -166, -165, -164,
  -163, -162, -161, -160, -159, -158, -157, -156, -155, -154, -153, -152, -151,
  -150, -149, -148, -147, -146, -145, -144, -143, -142, -141, -140, -139, -138,
  -137, -136, -135, -134, -133, -132, -131, -130, -129, -128, -127, -126, -125,
  -124, -123, -122, -121, -120, -119, -118, -117, -116,      -115, -114, -113,
  -112, -111, -110, -109, -108, -107, -106, -105, -104, -103, -102, -101, -100,
  -99, -98, -97, -96, -95, -94, -93, -92, -91, -90, -89, -88, -87, -86, -85,
  -84, -83, -82, -81, -80, -79, -78, -77, -76, -75, -74, -73, -72, -71, -70,
  -69, -68, -67, -66, -65, -64, -63, -62, -61, -60, -59, -58, -57, -56, -55,
  -54, -53, -52, -51, -50, -49, -48, -47, -46, -45, -44, -43, -42, -41, -40,
  -39, -38, -37, -36, -35, -34, -33, -32, -31, -30, -29, -28, -27, -26, -25,
  -24, -23, -22, -21, -20, -19,      -18, -17, -16, -15, -14, -13, -12, -11,
  -10, -9, -8, -7, -6, -5, -4, -3, -2, -1), temporalEndDays = c(-365, -364,
  -363, -362, -361, -360, -359, -358, -357, -356, -355, -354, -353, -352, -351,
  -350, -349, -348, -347, -346, -345, -344, -343, -342, -341, -340, -339, -338,
  -337, -336, -335, -334, -333, -332, -331, -330, -329, -328, -327, -326, -325,
  -324, -323, -322, -321, -320, -319, -318, -317, -316, -315, -314, -313, -312,
  -311, -310, -309, -308, -307, -306, -305, -304, -303, -302, -301, -300, -299,
  -298, -297, -296, -295, -294, -293, -292, -291, -290, -289, -288, -287, -286,
  -285, -284, -283, -282,      -281, -280, -279, -278, -277, -276, -275, -274,
  -273, -272, -271, -270, -269, -268, -267, -266, -265, -264, -263, -262, -261,
  -260, -259, -258, -257, -256, -255, -254, -253, -252, -251, -250, -249, -248,
  -247, -246, -245, -244, -243, -242, -241, -240, -239, -238, -237, -236, -235,
  -234, -233, -232, -231, -230, -229, -228, -227, -226, -225, -224, -223, -222,
  -221, -220, -219, -218, -217, -216, -215, -214, -213, -212, -211, -210, -209,
```

```
  -208, -207, -206, -205, -204, -203, -202, -201, -200, -199,      -198, -197,
  -196, -195, -194, -193, -192, -191, -190, -189, -188, -187, -186, -185, -184,
  -183, -182, -181, -180, -179, -178, -177, -176, -175, -174, -173, -172, -171,
  -170, -169, -168, -167, -166, -165, -164, -163, -162, -161, -160, -159, -158,
  -157, -156, -155, -154, -153, -152, -151, -150, -149, -148, -147, -146, -145,
  -144, -143, -142, -141, -140, -139, -138, -137, -136, -135, -134, -133, -132,
  -131, -130, -129, -128, -127, -126, -125, -124, -123, -122, -121, -120, -119,
  -118, -117, -116,      -115, -114, -113, -112, -111, -110, -109, -108, -107,
  -106, -105, -104, -103, -102, -101, -100, -99, -98, -97, -96, -95, -94, -93,
  -92, -91, -90, -89, -88, -87, -86, -85, -84, -83, -82, -81, -80, -79, -78,
  -77, -76, -75, -74, -73, -72, -71, -70, -69, -68, -67, -66, -65, -64, -63,
  -62, -61, -60, -59, -58, -57, -56, -55, -54, -53, -52, -51, -50, -49, -48,
  -47, -46, -45, -44, -43, -42, -41, -40, -39, -38, -37, -36, -35, -34, -33,
  -32, -31, -30, -29, -28, -27, -26, -25, -24, -23, -22, -21, -20, -19,
  -18, -17, -16, -15, -14, -13, -12, -11, -10, -9, -8, -7, -6, -5, -4, -3, -2,
  -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 contructed 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.

---

createTemporalCovariateSettings

Create covariate settings

Description

Create covariate settings

Usage

```
createTemporalCovariateSettings(useDemographicsGender = FALSE,
  useDemographicsAge = FALSE, useDemographicsIndexYear = FALSE,
  useDemographicsIndexMonth = FALSE, useConditionOccurrence = 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,
  useObservation = FALSE, useCharlsonIndex = FALSE,
  temporalStartDays = c(-365, -364, -363, -362, -361, -360, -359, -358, -357,
  -356, -355, -354, -353, -352, -351, -350, -349, -348, -347, -346, -345, -344,
  -343, -342, -341, -340, -339, -338, -337, -336, -335, -334, -333, -332, -331,
  -330, -329, -328, -327, -326, -325, -324, -323, -322, -321, -320, -319, -318,
  -317, -316, -315, -314, -313, -312, -311, -310, -309, -308, -307, -306, -305,
  -304, -303, -302, -301, -300, -299, -298, -297, -296, -295, -294, -293, -292,
  -291, -290, -289, -288, -287, -286, -285, -284, -283, -282,      -281, -280,
  -279, -278, -277, -276, -275, -274, -273, -272, -271, -270, -269, -268, -267,
  -266, -265, -264, -263, -262, -261, -260, -259, -258, -257, -256, -255, -254,
  -253, -252, -251, -250, -249, -248, -247, -246, -245, -244, -243, -242, -241,
  -240, -239, -238, -237, -236, -235, -234, -233, -232, -231, -230, -229, -228,
  -227, -226, -225, -224, -223, -222, -221, -220, -219, -218, -217, -216, -215,
  -214, -213, -212, -211, -210, -209, -208, -207, -206, -205, -204, -203, -202,
  -201, -200, -199,      -198, -197, -196, -195, -194, -193, -192, -191, -190,
  -189, -188, -187, -186, -185, -184, -183, -182, -181, -180, -179, -178, -177,
  -176, -175, -174, -173, -172, -171, -170, -169, -168, -167, -166, -165, -164,
  -163, -162, -161, -160, -159, -158, -157, -156, -155, -154, -153, -152, -151,
  -150, -149, -148, -147, -146, -145, -144, -143, -142, -141, -140, -139, -138,
  -137, -136, -135, -134, -133, -132, -131, -130, -129, -128, -127, -126, -125,
  -124, -123, -122, -121, -120, -119, -118, -117, -116,      -115, -114, -113,
  -112, -111, -110, -109, -108, -107, -106, -105, -104, -103, -102, -101, -100,
  -99, -98, -97, -96, -95, -94, -93, -92, -91, -90, -89, -88, -87, -86, -85,
  -84, -83, -82, -81, -80, -79, -78, -77, -76, -75, -74, -73, -72, -71, -70,
  -69, -68, -67, -66, -65, -64, -63, -62, -61, -60, -59, -58, -57, -56, -55,
  -54, -53, -52, -51, -50, -49, -48, -47, -46, -45, -44, -43, -42, -41, -40,
  -39, -38, -37, -36, -35, -34, -33, -32, -31, -30, -29, -28, -27, -26, -25,
  -24, -23, -22, -21, -20, -19,      -18, -17, -16, -15, -14, -13, -12, -11,
  -10, -9, -8, -7, -6, -5, -4, -3, -2, -1), temporalEndDays = c(-365, -364,
  -363, -362, -361, -360, -359, -358, -357, -356, -355, -354, -353, -352, -351,
  -350, -349, -348, -347, -346, -345, -344, -343, -342, -341, -340, -339, -338,
  -337, -336, -335, -334, -333, -332, -331, -330, -329, -328, -327, -326, -325,
  -324, -323, -322, -321, -320, -319, -318, -317, -316, -315, -314, -313, -312,
  -311, -310, -309, -308, -307, -306, -305, -304, -303, -302, -301, -300, -299,
  -298, -297, -296, -295, -294, -293, -292, -291, -290, -289, -288, -287, -286,
  -285, -284, -283, -282,      -281, -280, -279, -278, -277, -276, -275, -274,
  -273, -272, -271, -270, -269, -268, -267, -266, -265, -264, -263, -262, -261,
  -260, -259, -258, -257, -256, -255, -254, -253, -252, -251, -250, -249, -248,
  -247, -246, -245, -244, -243, -242, -241, -240, -239, -238, -237, -236, -235,
  -234, -233, -232, -231, -230, -229, -228, -227, -226, -225, -224, -223, -222,
  -221, -220, -219, -218, -217, -216, -215, -214, -213, -212, -211, -210, -209,
  -208, -207, -206, -205, -204, -203, -202, -201, -200, -199,      -198, -197,
```

```
-196, -195, -194, -193, -192, -191, -190, -189, -188, -187, -186, -185, -184,
-183, -182, -181, -180, -179, -178, -177, -176, -175, -174, -173, -172, -171,
-170, -169, -168, -167, -166, -165, -164, -163, -162, -161, -160, -159, -158,
-157, -156, -155, -154, -153, -152, -151, -150, -149, -148, -147, -146, -145,
-144, -143, -142, -141, -140, -139, -138, -137, -136, -135, -134, -133, -132,
-131, -130, -129, -128, -127, -126, -125, -124, -123, -122, -121, -120, -119,
-118, -117, -116,      -115, -114, -113, -112, -111, -110, -109, -108, -107,
-106, -105, -104, -103, -102, -101, -100, -99, -98, -97, -96, -95, -94, -93,
-92, -91, -90, -89, -88, -87, -86, -85, -84, -83, -82, -81, -80, -79, -78,
-77, -76, -75, -74, -73, -72, -71, -70, -69, -68, -67, -66, -65, -64, -63,
-62, -61, -60, -59, -58, -57, -56, -55, -54, -53, -52, -51, -50, -49, -48,
-47, -46, -45, -44, -43, -42, -41, -40, -39, -38, -37, -36, -35, -34, -33,
-32, -31, -30, -29, -28, -27, -26, -25, -24, -23, -22, -21, -20, -19,
-18, -17, -16, -15, -14, -13, -12, -11, -10, -9, -8, -7, -6, -5, -4, -3, -2,
-1), includedCovariateConceptIds = c(), addDescendantsToInclude = FALSE,
excludedCovariateConceptIds = c(), addDescendantsToExclude = FALSE,
includedCovariateIds = c())
```

Arguments

`useDemographicsGender`
: Gender of the subject.

`useDemographicsAge`
: Age of the subject on the index date (in 5 year increments).

`useDemographicsIndexYear`
: Year of the index date.

`useDemographicsIndexMonth`
: Month of the index date.

`useConditionOccurrence`
: One covariate per condition in the condition_occurrence table starting in the time window.

`useConditionEraStart`
: One covariate per condition in the condition_era table starting in the time window.

`useConditionEraOverlap`
: One covariate per condition in the condition_era table overlapping with any part of the time window.

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

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

`useDrugExposure`
: One covariate per drug in the drug_exposure table starting in the time window.

`useDrugEraStart`
: One covariate per drug in the drug_era table starting in the time window.

`useDrugEraOverlap`
: One covariate per drug in the drug_era table overlapping with any part of the time window.

`useDrugEraGroupStart`
: One covariate per drug rolled up to ATC groups in the drug_era table starting in thetime window.

`useDrugEraGroupOverlap`
: One covariate per drug rolled up to ATC groups in the drug_era table overlapping with any part of thetime window.

`useProcedureOccurrence`
: One covariate per procedure in the procedure_occurrence table in the time window.

`useDeviceExposure`
: One covariate per device in the device exposure table starting in the timewindow.

`useMeasurement` One covariate per measurement in the measurement table in the time window.

`useObservation` One covariate per observation in the observation table in the time window.

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

`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 contructed from data in the CDM model.

Value

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

`createTextCovariateSettings` *Create text covariate settings*

Description

Create text covariate settings

Usage

```
createTextCovariateSettings(language = "eng", removeNegations = TRUE,
  deleteCovariatesSmallCount = 100)
```

Arguments

`language`	Specify the language of the free-text.
`removeNegations`	Remove negated text prior to constructing features.
`deleteCovariatesSmallCount`	A numeric value used to remove covariates that occur in both cohorts fewer than deleteCovariateSmallCounts time.

Details

creates an object specifying how covariates should be constructed from text in notes table in the CDM model.

Value

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

`FeatureExtraction` *FeatureExtraction*

Description

FeatureExtraction

getDbCohortAttrCovariatesData *Getcovariate information from the database through the cohort_attribute table*

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

```
getDbCohortAttrCovariatesData(connection, oracleTempSchema = NULL,
  cdmDatabaseSchema, cdmVersion = "4", cohortTempTable = "cohort_person",
  rowIdField = "subject_id", covariateSettings)
```

Arguments

connection	A connection to the server containing the schema as created using the connect function in the DatabaseConnector package.
oracleTempSchema	A schema where temp tables can be created in Oracle.
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'.
cdmVersion	Define the OMOP CDM version used: currently support "4" and "5".
cohortTempTable	Name of the temp table holding the cohort for which we want to construct covaraites
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.

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, 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 An ffdf object describing the covariates that have been extracted.

metaData 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, cohortIds = c(), rowIdField = "subject_id",
  covariateSettings, aggregated = FALSE)
```

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	A schema where temp tables can be created in Oracle.
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'.
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 specifiy both the database and the schema, so for example 'cdm_instance.dbo'.

`cohortTableIsTemp`
: Is the cohort table a temp table?

`cohortIds`
: For which cohort IDs should covariates be constructed? If left empty, 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.

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', '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.

Value

Returns an object of type `covariateData`, 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 An ffdf object describing the covariates that have been extracted.

metaData A list of objects with information on how the covariateData object was constructed.

`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, cohortTempTable = "#cohort_person",
  rowIdField = "subject_id", covariateSettings, aggregated = FALSE)
```

Arguments

connection	A connection to the server containing the schema as created using the connect function in the DatabaseConnector package.
oracleTempSchema	A schema where temp tables can be created in Oracle.
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'.
cohortTempTable	Name of the temp table holding the cohort for which we want to construct covaraites
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 defaultCovariateSettings as created using the createCovariateSettings function.
cdmVersion	Define the OMOP CDM version used: currently support "4" and "5".

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, 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 An ffdf object describing the covariates that have been extracted.

metaData A list of objects with information on how the covariateData object was constructed.

---

getDbTextCovariateData

Get text covariate information from the database

Description

Uses a bag-of-words approach to construct covariates based on free-text.

Usage

```
getDbTextCovariateData(connection, oracleTempSchema = NULL, cdmDatabaseSchema,
  cdmVersion = "4", cohortTempTable = "cohort_person",
  rowIdField = "subject_id", covariateSettings)
```

Arguments

`connection`	A connection to the server containing the schema as created using the `connect` function in the `DatabaseConnector` package.
`oracleTempSchema`	A schema where temp tables can be created in Oracle.
`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'.
`cdmVersion`	Define the OMOP CDM version used: currently support "4" and "5".
`cohortTempTable`	Name of the temp table holding the cohort for which we want to construct covaraites
`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 `createTextCovariateSettings` function.

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`, 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 An ffdf object describing the covariates that have been extracted.

metaData A list of objects with information on how the covariateData object was constructed.

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 = FALSE)
```

Arguments

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

Details

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

Value

An object of class covariateData

Examples

```
# todo
```

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.

Examples

```
# todo
```

`tidyCovariateData` *Tidy covariate data*

Description

Tidy covariate data

Usage

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

Arguments

`covariateData`	An object as generated using the `getDbCovariateData` function.
`normalize`	Normalize the coviariates? (dividing by the max)
`removeRedundancy`	Should redundant covariates be removed?

Details

Normalize covariate values by dividing by the max and/or remove redundant covariates.

Index