Package: OmopSketch (via r-universe)

October 21, 2024

Title Characterise Tables of an OMOP Common Data Model Instance

Version 0.1.1

Maintainer Marta Alcalde-Herraiz
<marta.alcaldeherraiz@ndorms.ox.ac.uk>

Description Summarises key information in data mapped to the
Observational Medical Outcomes Partnership (OMOP) common data
model. Assess suitability to perform specific epidemiological
studies and explore the different domains to obtain feasibility
counts and trends.

License Apache License (>= 2)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Suggests CodelistGenerator, DBI, duckdb, flextable, gt, here, knitr,
odbc, remotes, rmarkdown, RPostgres, testthat (>= 3.0.0),
withr, omock (>= 0.3.0)

Config/testthat/edition 3

Config/testthat/parallel true

Imports CDMConnector (>= 1.3.0), cli, clock, CohortConstructor (>=
0.3.1), dplyr, ggplot2, omopgenerics (>= 0.3.1),
PatientProfiles (>= 1.2.0.900), purrr, rlang, stringr, tibble,
tidyr, visOmopResults (>= 0.4.0)

Depends R (>= 2.10)

URL https://OHDSI.github.io/OmopSketch/

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

VignetteBuilder knitr

Remotes darwin-eu-dev/omopgenerics@mah_vl,
darwin-eu-dev/PatientProfiles@issue_705

Repository https://ohdsi.r-universe.dev

RemoteUrl https://github.com/ohdsi/omopsketch

RemoteRef HEAD

RemoteSha b575776183c4c935addfe6eef3b27b0d7a7dee91

Contents

mockOmopSketch	*Creates a mock database to test OmopSketch package.*

Description

Creates a mock database to test OmopSketch package.

Usage

```
mockOmopSketch(
  con = NULL,
  writeSchema = NULL,
  numberIndividuals = 100,
  seed = NULL
)
```

Arguments

con A DBI connection to create the cdm mock object. By default, the connection
would be a 'duckdb' one.

writeSchema Name of an schema of the DBI connection with writing permissions.

numberIndividuals
Number of individuals to create in the cdm reference object.

seed | An optional integer used to set the seed for random number generation, ensuring reproducibility of the generated data. If provided, this seed allows the function to produce consistent results each time it is run with the same inputs. If 'NULL', the seed is not set, which can lead to different outputs on each run.

Value

A mock cdm_reference object.

Examples

```
mockOmopSketch(numberIndividuals = 100)
```

---

plotConceptCounts *Plot the concept counts of a summariseConceptCounts output.*

Description

Plot the concept counts of a summariseConceptCounts output.

Usage

```
plotConceptCounts(result, facet = NULL, colour = NULL)
```

Arguments

result | A summarised_result object (output of summariseConceptCounts).

facet | Columns to face by. Formula format can be provided. See possible columns to face by with: `visOmopResults::tidyColumns()`.

colour | Columns to colour by. See possible columns to colour by with: `visOmopResults::tidyColumns()`.

Value

A ggplot2 object showing the concept counts.

Examples

```
library(dplyr)

cdm <- mockOmopSketch()

result <- cdm |>
  summariseConceptCounts(
    conceptId = list(
      "Renal agenesis" = 194152,
      "Manic mood" = c(4226696, 4304866, 37110496, 40371897)
    )
  )
```

```
result |>
  filter(variable_name == "Number subjects") |>
  plotConceptCounts(facet = "codelist_name", colour = "codelist_name")

PatientProfiles::mockDisconnect(cdm)
```

---

plotInObservation   *Create a ggplot2 plot from the output of summariseInObservation( ).*

Description

Create a ggplot2 plot from the output of summariseInObservation().

Usage

```
plotInObservation(result, facet = NULL, colour = NULL)
```

Arguments

result	A summarised_result object (output of summariseInObservation).
facet	Columns to face by. Formula format can be provided. See possible columns to face by with: visOmopResults::tidyColumns().
colour	Columns to colour by. See possible columns to colour by with: visOmopResults::tidyColumns().

Value

A ggplot showing the table counts

Examples

```
library(dplyr)

cdm <- mockOmopSketch()

result <- summariseInObservation(
  cdm$observation_period,
  output = c("person-days","records"),
  ageGroup = list("<=40" = c(0, 40), ">40" = c(41, Inf)),
  sex = TRUE
)

result |>
  filter(variable_name == "Number person-days") |>
  plotInObservation(facet = "sex", colour = "age_group")

PatientProfiles::mockDisconnect(cdm)
```

---

plotObservationPeriod   *Create a plot from the output of summariseObservationPeriod().*

Description

Create a plot from the output of summariseObservationPeriod().

Usage

```
plotObservationPeriod(
  result,
  variableName = "number subjects",
  plotType = "barplot",
  facet = NULL,
  colour = NULL
)
```

Arguments

result	A summarised_result object.
variableName	The variable to plot it can be: "number subjects", "records per person", "duration" or "days to next observation period".
plotType	The plot type, it can be: "barplot", "boxplot" or "densityplot".
facet	Columns to colour by. See possible columns to colour by with: `visOmopResults::tidyColumns()`.
colour	Columns to colour by. See possible columns to colour by with: `visOmopResults::tidyColumns()`.

Value

A ggplot2 object.

Examples

```
cdm <- mockOmopSketch(numberIndividuals = 100)

result <- summariseObservationPeriod(cdm$observation_period)

result |>
  plotObservationPeriod(
    variableName = "duration in days",
    plotType = "boxplot"
  )

PatientProfiles::mockDisconnect(cdm)
```

---

plotRecordCount   *Create a ggplot of the records' count trend.*

Description

Create a ggplot of the records' count trend.

Usage

```
plotRecordCount(result, facet = NULL, colour = NULL)
```

Arguments

result   Output from summariseRecordCount().

facet   Columns to face by. Formula format can be provided. See possible columns to
                face by with: `visOmopResults::tidyColumns()`.

colour   Columns to colour by. See possible columns to colour by with: `visOmopResults::tidyColumns()`.

Value

A ggplot showing the table counts

Examples

```
cdm <- mockOmopSketch()

summarisedResult <- summariseRecordCount(
  cdm = cdm,
  omopTableName = "condition_occurrence",
  ageGroup = list("<=20" = c(0,20), ">20" = c(21, Inf)),
  sex = TRUE
)

plotRecordCount(summarisedResult, colour = "age_group", facet = sex ~ .)

PatientProfiles::mockDisconnect(cdm = cdm)
```

---

summariseClinicalRecords

Summarise an omop table from a cdm object. You will obtain information related to the number of records, number of subjects, whether the records are in observation, number of present domains and number of present concepts.

Description

Summarise an omop table from a cdm object. You will obtain information related to the number of records, number of subjects, whether the records are in observation, number of present domains and number of present concepts.

Usage

```
summariseClinicalRecords(
  cdm,
  omopTableName,
  recordsPerPerson = c("mean", "sd", "median", "q25", "q75", "min", "max"),
  inObservation = TRUE,
  standardConcept = TRUE,
  sourceVocabulary = FALSE,
  domainId = TRUE,
  typeConcept = TRUE,
  sex = FALSE,
  ageGroup = NULL
)
```

Arguments

cdm	A cdm_reference object.
omopTableName	A character vector of the names of the tables to summarise in the cdm object.
recordsPerPerson	
	Generates summary statistics for the number of records per person. Set to NULL if no summary statistics are required.
inObservation	Boolean variable. Whether to include the percentage of records in observation.
standardConcept	
	Boolean variable. Whether to summarise standard concept information.
sourceVocabulary	
	Boolean variable. Whether to summarise source vocabulary information.
domainId	Boolean variable. Whether to summarise domain id of standard concept id information.
typeConcept	Boolean variable. Whether to summarise type concept id field information.
sex	Boolean variable. Whether to stratify by sex (TRUE) or not (FALSE).
ageGroup	A list of age groups to stratify results by.

Value

A summarised_result object.

Examples

```
cdm <- mockOmopSketch()

summarisedResult <- summariseClinicalRecords(
```

```
  cdm = cdm,
  omopTableName = "condition_occurrence",
  recordsPerPerson = c("mean", "sd"),
  inObservation = TRUE,
  standardConcept = TRUE,
  sourceVocabulary = TRUE,
  domainId = TRUE,
  typeConcept = TRUE
)

summarisedResult

PatientProfiles::mockDisconnect(cdm = cdm)
```

---

summariseConceptCounts

> *Summarise concept counts in patient-level data. Only concepts recorded during observation period are counted.*

Description

Summarise concept counts in patient-level data. Only concepts recorded during observation period are counted.

Usage

```
summariseConceptCounts(
  cdm,
  conceptId,
  countBy = c("record", "person"),
  concept = TRUE,
  interval = "overall",
  sex = FALSE,
  ageGroup = NULL
)
```

Arguments

cdm	A cdm object
conceptId	List of concept IDs to summarise.
countBy	Either "record" for record-level counts or "person" for person-level counts
concept	TRUE or FALSE. If TRUE code use will be summarised by concept.
interval	Time interval to stratify by. It can either be "years", "quarters", "months" or "overall".
sex	TRUE or FALSE. If TRUE code use will be summarised by sex.
ageGroup	A list of ageGroup vectors of length two. Code use will be thus summarised by age groups.

Value

A summarised_result object with results overall and, if specified, by strata.

Examples

```
library(OmopSketch)

cdm <- mockOmopSketch()

cs <- list(sumatriptan = c(35604883, 35604879, 35604880, 35604884))

results <- summariseConceptCounts(cdm, conceptId = cs)

results

PatientProfiles::mockDisconnect(cdm)
```

---

summariseInObservation

>*Summarise the number of people in observation during a specific interval of time.*

Description

Summarise the number of people in observation during a specific interval of time.

Usage

```
summariseInObservation(
  observationPeriod,
  interval = "overall",
  output = "records",
  ageGroup = NULL,
  sex = FALSE
)
```

Arguments

observationPeriod
>An observation_period omop table. It must be part of a cdm_reference object.

interval
>Time interval to stratify by. It can either be "years", "quarters", "months" or "overall".

output
>Output format. It can be either the number of records ("records") that are in observation in the specific interval of time, the number of person-days ("person-days"), or both c("records","person-days").

ageGroup
>A list of age groups to stratify results by.

sex
>Boolean variable. Whether to stratify by sex (TRUE) or not (FALSE).

Value

A summarised_result object.

Examples

```
library(dplyr, warn.conflicts = FALSE)

cdm <- mockOmopSketch()

result <- summariseInObservation(
  cdm$observation_period,
  interval = "months",
  output = c("person-days","records"),
  ageGroup = list("<=60" = c(0,60), ">60" = c(61, Inf)),
  sex = TRUE
)

result |>
  glimpse()

PatientProfiles::mockDisconnect(cdm)
```

---

summariseObservationPeriod

Summarise the observation period table getting some overall statistics in a summarised_result object.

Description

Summarise the observation period table getting some overall statistics in a summarised_result object.

Usage

```
summariseObservationPeriod(
  observationPeriod,
  estimates = c("mean", "sd", "min", "q05", "q25", "median", "q75", "q95", "max",
    "density"),
  ageGroup = NULL,
  sex = FALSE
)
```

Arguments

observationPeriod
                observation_period omop table.

estimates	Estimates to summarise the variables of interest (records per person, duration in days and days to next observation period).
ageGroup	A list of age groups to stratify results by.
sex	Boolean variable. Whether to stratify by sex (TRUE) or not (FALSE).

Value

A summarised_result object with the summarised data.

Examples

```
library(dplyr, warn.conflicts = FALSE)

cdm <- mockOmopSketch(numberIndividuals = 100)

result <- summariseObservationPeriod(cdm$observation_period)

result |>
  glimpse()

PatientProfiles::mockDisconnect(cdm)
```

---

summariseOmopSnapshot *Summarise a cdm_reference object creating a snapshot with the meta-data of the cdm_reference object.*

Description

Summarise a cdm_reference object creating a snapshot with the metadata of the cdm_reference object.

Usage

```
summariseOmopSnapshot(cdm)
```

Arguments

cdm	A cdm_reference object.

Value

A summarised_result object.

Examples

```
cdm <- mockOmopSketch(numberIndividuals = 10)

summariseOmopSnapshot(cdm)
```

---

summariseRecordCount   *Summarise record counts of an omop_table using a specific time inter-*
                              val. Only records that fall within the observation period are consid-
                              ered.

Description

Summarise record counts of an omop_table using a specific time interval. Only records that fall
within the observation period are considered.

Usage

```
summariseRecordCount(
  cdm,
  omopTableName,
  interval = "overall",
  ageGroup = NULL,
  sex = FALSE
)
```

Arguments

cdm	A cdm_reference object.
omopTableName	A character vector of omop tables from the cdm.
interval	Time interval to stratify by. It can either be "years", "quarters", "months" or "overall".
ageGroup	A list of age groups to stratify results by.
sex	Whether to stratify by sex (TRUE) or not (FALSE).

Value

A summarised_result object.

Examples

```
library(dplyr, warn.conflicts = FALSE)

cdm <- mockOmopSketch()

summarisedResult <- summariseRecordCount(
  cdm = cdm,
  omopTableName = c("condition_occurrence", "drug_exposure"),
  interval = "years",
  ageGroup = list("<=20" = c(0,20), ">20" = c(21, Inf)),
  sex = TRUE
)
```

```
summarisedResult |>
  glimpse()

PatientProfiles::mockDisconnect(cdm = cdm)
```

---

tableClinicalRecords   *Create a visual table from a summariseClinicalRecord() output.*

Description

Create a visual table from a summariseClinicalRecord() output.

Usage

```
tableClinicalRecords(result, type = "gt")
```

Arguments

result	Output from summariseClinicalRecords().
type	Type of formatting output table, either "gt" or "flextable".

Value

A gt or flextable object with the summarised data.

Examples

```
cdm <- mockOmopSketch()

summarisedResult <- summariseClinicalRecords(
  cdm = cdm,
  omopTableName = c("condition_occurrence", "drug_exposure"),
  recordsPerPerson = c("mean", "sd"),
  inObservation = TRUE,
  standardConcept = TRUE,
  sourceVocabulary = TRUE,
  domainId = TRUE,
  typeConcept = TRUE
)

summarisedResult |>
  suppress(minCellCount = 5) |>
  tableClinicalRecords()

PatientProfiles::mockDisconnect(cdm)
```

---

tableObservationPeriod

Create a visual table from a summariseObservationPeriod() result.

Description

Create a visual table from a summariseObservationPeriod() result.

Usage

```
tableObservationPeriod(result, type = "gt")
```

Arguments

result	A summarised_result object.
type	Type of formatting output table, either "gt" or "flextable".

Value

A gt or flextable object with the summarised data.

Examples

```
cdm <- mockOmopSketch(numberIndividuals = 100)

result <- summariseObservationPeriod(cdm$observation_period)

tableObservationPeriod(result)

PatientProfiles::mockDisconnect(cdm)
```

---

tableOmopSnapshot   *Create a visual table from a summarise_omop_snapshot result.*

Description

Create a visual table from a summarise_omop_snapshot result.

Usage

```
tableOmopSnapshot(result, type = "gt")
```

Arguments

result	Output from summariseOmopSnapshot().
type	Type of formatting output table, either "gt" or "flextable".

Value

A gt or flextable object with the summarised data.

Examples

```
cdm <- mockOmopSketch(numberIndividuals = 10)

result <- summariseOmopSnapshot(cdm)

result |>
  tableOmopSnapshot()

PatientProfiles::mockDisconnect(cdm)
```

Index