Package 'OmopIndices'

November 4, 2026

Type Package

Title Patient-Level Indices from the OMOP Common Data Model

Version 0.3.0

Maintainer Martí Català <marti.catalasabate@ndorms.ox.ac.uk>

Description Provides tools to derive standardised, reproducible patient-level indices and covariates from Observational Medical Outcomes Partnership (OMOP) Common Data Model (CDM) databases. Functions calculate comorbidity and frailty scores, including the Charlson Comorbidity Index, Electronic Frailty Index, and Hospital Frailty Risk Score, as well as body mass index, polypharmacy, ethnicity, location, and socioeconomic status measures.

License Apache License (>= 2)

Encoding UTF-8

Suggests CDMConnector, CodelistGenerator, CohortConstructor, DBI, duckdb, gt, here, htmltools, knitr, odbc, omock, PhenotypeR, rmarkdown, RPostgres, testthat, visOmopResults

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

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

Imports cli, clock, dplyr, lifecycle, OmopConstructor, omopgenerics (>= 1.4.0), PatientProfiles, purrr, rlang

Depends R (>= 4.1.0)

Config/roxygen2/version 8.1.0

Config/Needs/website rmarkdown

NeedsCompilation no

Author Martí Català [aut, cre] (ORCID: <https://orcid.org/0000-0003-3308-9905>), Marta Alcalde-Herraiz [aut] (ORCID: <https://orcid.org/0009-0002-4405-1814>), Maria T Sanchez-Santos [aut] (ORCID: <https://orcid.org/0000-0003-1908-8623>)

Repository CRAN

Date/Publication 2026-11-04 17:30:02 UTC

Contents

addBMI 2
addCharlsonIndex 3
addElectronicFrailtyIndex 6
addEthnicity 8
addHospitalFrailtyRiskScore 10
addIndexOfMultipleDeprivation 12
addLocation 13
addPolypharmacy 14
addPolypharmacyCount 16
addSocioEconomicStatus 17
addTownsend 18
addUpdatedCharlsonIndex 20
charlsonCohort 22
electronicFrailtyIndexCohort 23
getIndexCodelist 25
updatedCharlsonCohort 26

Index **28**

`addBMI` *Add Body Mass Index measurement*

Description

Add Body Mass Index measurement

Usage

```
addBMI(
  x,
  conceptSet = getIndexCodelist("body_mass_index"),
  indexDate = "cohort_start_date",
  window = c(-Inf, 0),
  order = "last",
  nameStyle = "bmi",
  categories = NULL,
  inObservation = TRUE,
  name = tableName(x)
)
```

Arguments

`x`	A `cdm_table` containing a person identifier column named `person_id` or `subject_id`.
`conceptSet`	A named concept set supplied as a `codelist`, `codelist_with_details`, `concept_set_expression`, or named list of concept IDs. It must contain `bmi` as concepts. By default, internal codelists are used.

indexDate	A character string naming the Date column in x that defines the index date.
window	A numeric vector of length two, c(min, max), defining the window for BMI in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.
order	A character string specifying how to select among multiple BMI measurements within the window: last (latest), first (earliest), max (highest), or min (lowest).
nameStyle	A character string specifying the name of the new column.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a BMI interval. An additional column named by appending _categories to nameStyle is added, and missing BMI values are labelled missing.
inObservation	Logical; whether to restrict records to the person's observation period.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.

Value

The table x with a new column containing the selected BMI value.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addBMI(window = c(-365, 0), order = "last") |>
  select(subject_id, cohort_start_date, bmi) |>
  glimpse()
```

---

addCharlsonIndex *Add Charlson Comorbidity Index (CCI) value based on Charlson et al. (1987) (*R*hrefhttps://doi.org/10.1016/0021-9681%2887%2990171-8doi:10.1016/00219681(87)901718) and Charlson et al. (1994) (*R*hrefhttps://doi.org/10.1016/0895-4356%2894%2990129-5doi:10.1016/08954356(94)901295) (age-adjusted) version.*

Description

Add Charlson Comorbidity Index (CCI) value based on Charlson et al. (1987) (doi:10.1016/0021-9681(87)901718) and Charlson et al. (1994) (doi:10.1016/08954356(94)901295) (age-adjusted) version.

Usage

```
addCharlsonIndex(
  x,
  indexDate = "cohort_start_date",
  ageAdjusted = TRUE,
  conceptSet = getIndexCodelist("charlson"),
  window = NULL,
  nameStyle = "charlson",
  categories = NULL,
  name = tableName(x)
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
indexDate	A character string naming the Date column in x that defines the index date.
ageAdjusted	Logical; whether to include age adjustment in the Charlson Comorbidity Index.
conceptSet	A named concept set supplied as a codelist, codelist_with_details, concept_set_expression, or named list of concept IDs. It must contain myocardial_infarction, congestive_heart_failure, peripheral_vascular_disease, cerebrovascular_disease, dementia, chronic_pulmonary_disea connective_tissue_disease, peptic_ulcer_disease, mild_liver_disease, diabetes_without_complication, hemiplegia, severe_chronic_kidney_disease, diabetes_with_complication, any_malignancy, moderate_or_severe_liver_disease, metastatic_solid_tumor, aids as concepts. By default, internal codelists are used.
window	NULL to use the windows defined by the index components, or a numeric vector of length two, c(min, max), to override the component windows in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.
nameStyle	A character string specifying the name of the new column.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. An additional column with the suffix _categories is added. Intervals are evaluated in the order supplied, and missing scores are labelled missing.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.

Value

The table x with a new column containing the Charlson Comorbidity Index value.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

# Using the internal concept sets:
cdm$cohort |>
  addCharlsonIndex(ageAdjusted = TRUE) |>
  select(subject_id, cohort_start_date, charlson) |>
  glimpse()

# This example uses custom concept sets.
customConceptSet <- list(
  myocardial_infarction = 329847L,
  congestive_heart_failure = 319835L,
  peripheral_vascular_disease = 321052L,
  cerebrovascular_disease = 381591L,
  dementia = 4182210L,
  chronic_pulmonary_disease = 255573L,
  connective_tissue_disease = 4134537L,
  peptic_ulcer_disease = 4027663L,
  mild_liver_disease = 194984L,
  moderate_or_severe_liver_disease = 4212540L,
  diabetes_without_complication = 201820L,
  diabetes_with_complication = 42538715L,
  hemiplegia = 374022L,
  severe_chronic_kidney_disease = 46271022L,
  any_malignancy = 4180914L,
  metastatic_solid_tumor = 432851L,
  aids = 4267414L
)

cdm$cohort |>
  addCharlsonIndex(
    conceptSet = customConceptSet,
    nameStyle = "charlson_custom"
  ) |>
  select(subject_id, cohort_start_date, charlson_custom) |>
  glimpse()
```

addElectronicFrailtyIndex *Add Electronic Frailty Index (eFI) value based on Clegg et al. (2016) (*Rhrefhttps://doi.org/10.1093/ageing/afw039doi:10.1093/ageing/afw039*)*

Description

Add Electronic Frailty Index (eFI) value based on Clegg et al. (2016) (doi:10.1093/ageing/afw039)

Usage

```
addElectronicFrailtyIndex(
  x,
  indexDate = "cohort_start_date",
  conceptSet = getIndexCodelist("electronic_frailty_index"),
 categories = list(fit = c(0, 0.12), mild = c(0.12, 0.24), moderate = c(0.24, 0.36),
    severe = c(0.36, 1)),
  window = NULL,
  nameStyle = "efi",
  name = tableName(x)
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
indexDate	A character string naming the Date column in x that defines the index date.
conceptSet	A named concept set supplied as a codelist, codelist_with_details, concept_set_expression, or named list of concept IDs. It must contain activity_limitation, anemia, arthritis, atrial_fibrillation, cerebrovascular_disease, chronic_kidney_disease, diabetes, dizziness, dyspnea, falls, foot_problem, fragility_fracture, hearing_impairment, heart_failure, heart_valve_disorder, housebound, hypertension, hypotension_syncope, ischemic_heart_disease, memory_cognitive_disorder, mobility_problems, osteoporosis, parkinsonism_tremor, peptic_ulcer, peripheral_vascular_disease, care_requirement, respiratory_disease, skin_ulcer, sleep_disturbance, social_vulnerability, thyroid_disease, urinary_incontinence, urinary_system_disease, visual_impairment, weight_loss_anorexia as concepts. By default, internal codelists are used.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. An additional column with the suffix _categories is added. Intervals are evaluated in the order supplied, and missing scores are labelled missing.
window	NULL to use the windows defined by the index components, or a numeric vector of length two, c(min, max), to override the component windows in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.

nameStyle	A character string specifying the name of the new column.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.

Value

The table x with a new column containing the Electronic Frailty Index value.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

# Using the internal concept sets:
cdm$cohort |>
  addElectronicFrailtyIndex() |>
  select(subject_id, cohort_start_date, efi, efi_categories) |>
  glimpse()

# This example uses custom concept sets.
customConceptSet <- list(
  activity_limitation = 763723L,
  anemia = 439777L,
  arthritis = 4291025L,
  atrial_fibrillation = 313217L,
  chronic_kidney_disease = 46271022L,
  cerebrovascular_disease = 381591L,
  dizziness = 4223938L,
  dyspnea = 312437L,
  falls = 4059015L,
  foot_problem = 4101512L,
  fragility_fracture = 3170964L,
  hearing_impairment = 4234647L,
  heart_failure = 316139L,
  heart_valve_disorder = 4281749L,
  housebound = 4052962L,
  hypertension = 319826L,
  hypotension_syncope = 316447L,
  ischemic_heart_disease = 4185932L,
  memory_cognitive_disorder = 4304008L,
  mobility_problems = 4053076L,
  osteoporosis = 80502L,
```

```
  parkinsonism_tremor = 4140090L,
  peptic_ulcer = 4027663L,
  peripheral_vascular_disease = 321052L,
  care_requirement = 3661927L,
  respiratory_disease = 317009L,
  skin_ulcer = 4262920L,
  sleep_disturbance = 435524L,
  social_vulnerability = 4026161L,
  diabetes = 201820L,
  thyroid_disease = 4017052L,
  urinary_incontinence = 197672L,
  urinary_system_disease = 75865L,
  visual_impairment = 4265433L,
  weight_loss_anorexia = 436675L
)

cdm$cohort |>
  addElectronicFrailtyIndex(
    conceptSet = customConceptSet,
    nameStyle = "efi_custom"
  ) |>
  select(subject_id, cohort_start_date, efi_custom, efi_custom_categories) |>
  glimpse()
```

`addEthnicity` *Add the ethnicity of a person to a table*

Description

Add the ethnicity of a person to a table

Usage

```
addEthnicity(
  x,
  from = c("ethnicity_concept_id", "ethnicity_source_concept_id", "race_concept_id",
    "race_source_concept_id"),
  nameStyle = "ethnicity",
  name = tableName(x),
  missingEthnicityValue = "Missing"
)
```

Arguments

`x`	A `cdm_table` containing a person identifier column named `person_id` or `subject_id`.
`from`	A character vector specifying the ethnicity sources to try, in priority order. The first source that provides a value is used. Available options are:

- **ethnicity_concept_id** to assign ethnicity using the concept_name associated with the ethnicity_concept_id column of the person table.
- **ethnicity_source_concept_id** to assign ethnicity using the concept_name associated with the ethnicity_source_concept_id column of the person table.
- **race_concept_id** to assign ethnicity using the concept_name associated with the race_concept_id column of the person table.
- **race_source_concept_id** to assign ethnicity using the concept_name associated with the race_source_concept_id column of the person table.
- **ethnicity_source_value** to assign ethnicity using the value of the column ethnicity_source_value in the person table.
- **race_source_value** to assign ethnicity using the value of the column race_source_value in the person table.
- **nhs-categories** to assign ethnicity using NHS Ethnic Category.
- **nhs-groups** to assign ethnicity using broad groups of NHS Ethnic Category as described in doi:10.1038/s41597024029581.

nameStyle A character string specifying the name of the new column.

name A character string specifying the name of the output table. If NULL, a temporary table is created.

missingEthnicityValue
A character string used to replace missing ethnicity values.

Value

The table x with a new column containing the patient's ethnicity.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addEthnicity() |>
  select(subject_id, cohort_start_date, ethnicity) |>
  glimpse()
```

addHospitalFrailtyRiskScore *Add the hospital frailty risk score as defined in Gilbert et al. (2018) (Rhrefhttps://doi.org/10.1016/S0140-6736%2818%2930668-8doi:10.1016/S01406736(18)306688)*

Description

Add the hospital frailty risk score as defined in Gilbert et al. (2018) (doi:10.1016/S01406736(18)30668-8)

Usage

```
addHospitalFrailtyRiskScore(
  x,
  indexDate = "cohort_start_date",
  conceptSet = getIndexCodelist("hospital_frailty_risk_score"),
  categories = list(low = c(0, 5), intermediate = c(5, 15), high = c(15, Inf)),
  window = NULL,
  nameStyle = "hfrs",
  name = tableName(x)
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
indexDate	A character string naming the Date column in x that defines the index date.
conceptSet	A named concept set supplied as a codelist, codelist_with_details, concept_set_expression, or named list of concept IDs. It must contain abnormal_results_of_function_studies, abnormalities_of_gait_and_mobility, abnormalities_of_heart_beat, acute_renal_failure, agent_resistant_to_penicillin_and_related_antibiotics, alzheimers_disease, artificial_opening_status, blindness_and_low_vision, calculus_of_kidney_and_ureter, care_involving_use_of_rehabilitation_procedures, carrier_of_infectious_disease, cellulitis, cerebral_infarction, cerebral_ischaemic_attacks, chronic_renal_failure, cognitive_functions_and_awareness, convulsions_not_elsewhere_classified, decubitus_ulcer, deficiency_of_other_b_group_vitamins, degenerative_diseases_of_nervou delirium, dementia_in_alzheimers_disease, dependence_on_enabling_machines_and_devices, depressive_episode, diarrhoea_and_gastroenteritis, disorders_of_kidney_and_ureter, disorders_of_mineral_metabolism, duodenal_ulcer, dysphagia, epilepsy, exposure_to_unspecified_factor, fall, fall_involving_bed, fall_on_and_from_stairs_and_s fever_of_unknown_origin, fluid_electrolyte_and_acid_base_balance, fracture_of_femur, fracture_of_lumbar_spine_and_pelvis, fracture_of_ribs_sternum_and_t fracture_of_shoulder_and_upper_arm, gangrene_not_elsewhere_classified, general_sensations_and_perceptions, hemiplegia, hypotension, infections_of_skin, intracranial_injury, mental_and_behavioural_disorders_due_to_use_of_alcohol, nausea_and_vomiting, nervous_and_musculoskeletal_systems_r_29_6_tendency_to_fall,

nosocomial_condition, open_wound_of_forearm, open_wound_of_head, osteoporosis_with_path
osteoporosis_without_pathological_fracture, other_abnormal_findings_of_blood_chemistr
other_anaemias, other_and_unspecified_injuries_of_head, other_arthrosis,
other_bacterial_agents, other_bacterial_intestinal_infections, other_cerebrovascular_d
other_diseases_of_digestive_system, other_disorders_of_pancreatic_internal_secretion,
other_disorders_of_urinary_system, other_fall_on_same_level, other_functional_intestin
other_hearing_loss, other_joint_disorders_not_elsewhere_classified,
other_medical_procedures, other_noninfective_gastroenteritis_and_colitis,
other_septicaemia, other_soft_tissue_disorders_not_elsewhere_classified,
parkinsons_disease, personal_history_of_other_diseases_and_conditions,
personal_history_of_risk_factors, pneumonia_organism_unspecified,
pneumonitis_due_to_solids_and_liquids, polyarthrosis, problems_related_to_care_provide
problems_related_to_life_management_difficulty, problems_related_to_medical_facilitie
problems_related_to_social_environment, prosthetic_devices_implants,
respiratory_failure_not_elsewhere_classified, retention_of_urine,
scoliosis, senility, sequelae_of_cerebrovascular_disease, somnolence_stupor_and_coma,
speech_disturbances_not_elsewhere_classified, spinal_stenosis_secondary_code_only,
streptococcus_and_staphylococcus, superficial_injury_of_head, superficial_injury_of_lo
symptoms_and_signs_concerning_food_and_fluid_intake, symptoms_and_signs_involving_emo
syncope_and_collapse, thyrotoxicosis_hyperthyroidism, ulcer_of_lower_limb_not_elsewher
unknown_and_unspecified_causes_of_morbidity, unspecified_acute_lower_respiratory_infe
unspecified_dementia, unspecified_fall, unspecified_haematuria, unspecified_renal_failu
unspecified_urinary_incontinence, vascular_dementia, vitamin_d_deficiency,
volume_depletion as concepts. By default, internal codelists are used.

categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. An additional column with the suffix _categories is added. Intervals are evaluated in the order supplied, and missing scores are labelled missing.
window	NULL to use the windows defined by the index components, or a numeric vector of length two, c(min, max), to override the component windows in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.
nameStyle	A character string specifying the name of the new column.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.

Value

The table x with a new column containing the Hospital Frailty Risk Score value.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
```

```
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addHospitalFrailtyRiskScore() |>
  select(subject_id, cohort_start_date, hfrs, hfrs_categories) |>
  glimpse()
```

`addIndexOfMultipleDeprivation` *Add socioeconomic status as a column to a table using the* R*hrefhttps://athena.ohdsi.org/search-terms/terms/35812882Index of Multiple Deprivation*

Description

Add socioeconomic status as a column to a table using the Index of Multiple Deprivation

Usage

```
addIndexOfMultipleDeprivation(
  x,
  indexDate = "cohort_start_date",
  window = c(-Inf, Inf),
  order = "last",
  nameStyle = "socio_economic_status",
  name = tableName(x),
  missingSocioEconomicStatusValue = "Missing"
)
```

Arguments

`x`	A `cdm_table` containing a person identifier column named `person_id` or `subject_id`.
`indexDate`	A character string naming the `Date` column in `x` that defines the index date.
`window`	A numeric vector of length two, `c(min, max)`, defining the window for `socio_economic_status` in days relative to `indexDate`. Use `-Inf` or `Inf` for an unbounded lower or upper limit.
`order`	A character string specifying which record to select when multiple records are found: `first` or `last`.
`nameStyle`	A character string specifying the name of the new column.
`name`	A character string specifying the name of the output table. If `NULL`, a temporary table is created.
`missingSocioEconomicStatusValue`	A character string used to replace missing socioeconomic status values.

Value

The table x with a new column containing the Index of Multiple Deprivation value.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addIndexOfMultipleDeprivation() |>
  select(subject_id, cohort_start_date, socio_economic_status) |>
  glimpse()
```

---

addLocation *Add the location to a table*

Description

Add the location to a table

Usage

```
addLocation(
  x,
  from = c("location_id", "care_site_id"),
  nameStyle = "location",
  name = tableName(x),
  locationSource = "location_source_value",
  missingLocationValue = "Missing"
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
from	A character vector specifying the location sources to try, in priority order. The first source that provides a location is used. Supported values are location_id and care_site_id.

nameStyle	A character string specifying the name of the new column.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.
locationSource	A character string specifying the column to retrieve from the location table. Supported values include location_source_value, city, state, zip, county, country_concept_id, and country_source_value.
missingLocationValue	A character string used to replace missing location values.

Value

The table x with a new column containing the patient's location.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addLocation() |>
  select(subject_id, cohort_start_date, location) |>
  glimpse()
```

---

addPolypharmacy *Add the maximum number of ingredients to which an individual is simultaneously exposed within a specified window*

Description

Add the maximum number of ingredients to which an individual is simultaneously exposed within a specified window

Usage

```
addPolypharmacy(
  x,
  indexDate = "cohort_start_date",
  window = c(0, 0),
  overlap = TRUE,
  group = "ingredient drug_era",
  categories = NULL,
  nameStyle = "polypharmacy",
  name = tableName(x)
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
indexDate	A character string naming the Date column in x that defines the index date.
window	A numeric vector of length two, c(min, max), defining the window for polypharmacy in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.
overlap	Logical; if TRUE, count drug records that overlap in time. If FALSE, count drug records that occur within the window without requiring them to overlap one another.
group	Group at which drug records will be considered, it must be a choice between: • "concept_id" Records are not grouped, each drug_concept_id counts as 1. • "ingredient drug_era" Records are grouped using the drug_era table. • "ingredient drug_exposure" Records are grouped at ingredient level using the drug_exposure table. • "ATC 1st", "ATC 2nd", "ATC 3rd", "ATC 4th", "ATC 5th" Records are grouped using different ATC groups levels. ATC codelists are retrieved using CodelistGenerator::getATCCodes(). • Custom conceptSet a custom codelist, codelist_with_details or concept_set_expression concept set definition, drug_concept_ids not included in the conceptSet will be omitted.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. An additional column with the suffix _categories is added. Intervals are evaluated in the order supplied, and missing scores are labelled missing.
nameStyle	A character string specifying the name of the new column.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.

Value

The table x with a new column containing the maximum number of simultaneous ingredients in the window of interest.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addPolypharmacy(window = c(-30, 0)) |>
  select(subject_id, cohort_start_date, polypharmacy) |>
  glimpse()
```

`addPolypharmacyCount` *Add the maximum number of ingredients to which an individual is simultaneously exposed within a specified window*

Description

[Deprecated]

Usage

```
addPolypharmacyCount(
  x,
  indexDate = "cohort_start_date",
  window = c(0, 0),
  overlap = TRUE,
  categories = NULL,
  nameStyle = "polypharmacy",
  name = tableName(x)
)
```

Arguments

`x`	A `cdm_table` containing a person identifier column named `person_id` or `subject_id`.
`indexDate`	A character string naming the `Date` column in `x` that defines the index date.
`window`	A numeric vector of length two, `c(min, max)`, defining the window for `polypharmacy` in days relative to `indexDate`. Use `-Inf` or `Inf` for an unbounded lower or upper limit.

overlap	Logical; if TRUE, count drug eras that overlap in time. If FALSE, count drug eras that occur within the window without requiring them to overlap one another.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. An additional column with the suffix _categories is added. Intervals are evaluated in the order supplied, and missing scores are labelled missing.
nameStyle	A character string specifying the name of the new column.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.

Details

The function was renamed to addPolypharmacy()

Value

The table x with a new column containing the maximum number of simultaneous ingredients in the window of interest.

---

addSocioEconomicStatus

Add socioeconomic status as a column to a table

Description

Add socioeconomic status as a column to a table

Usage

```
addSocioEconomicStatus(
  x,
  indexDate = "cohort_start_date",
  window = c(-Inf, Inf),
  order = "last",
  from = c("imd", "townsend"),
  nameStyle = "socio_economic_status",
  name = tableName(x),
  missingSocioEconomicStatusValue = "Missing"
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
indexDate	A character string naming the Date column in x that defines the index date.
window	A numeric vector of length two, c(min, max), defining the window for socio_economic_status in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.

order	A character string specifying which record to select when multiple records are found: first or last.
from	A character vector specifying the socioeconomic status sources to try, in priority order. Supported values are imd, for the Index of Multiple Deprivation, and townsend, for the Townsend deprivation index.
nameStyle	A character string specifying the name of the new column.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.
missingSocioEconomicStatusValue	A character string used to replace missing socioeconomic status values.

Value

The table x with a new column containing the patient's socioeconomic status.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addSocioEconomicStatus() |>
  select(subject_id, cohort_start_date, socio_economic_status) |>
  glimpse()
```

---

addTownsend *Add socioeconomic status as a column to a table using the* R*hrefhttps://athena.ohdsi.org/search-terms/terms/715996Townsend deprivation index*

Description

Add socioeconomic status as a column to a table using the Townsend deprivation index

Usage

```
addTownsend(
  x,
  indexDate = "cohort_start_date",
  window = c(-Inf, Inf),
  order = "last",
  nameStyle = "socio_economic_status",
  name = tableName(x),
  missingSocioEconomicStatusValue = "Missing"
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
indexDate	A character string naming the Date column in x that defines the index date.
window	A numeric vector of length two, c(min, max), defining the window for socio_economic_status in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.
order	A character string specifying which record to select when multiple records are found: first or last.
nameStyle	A character string specifying the name of the new column.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.
missingSocioEconomicStatusValue	A character string used to replace missing socioeconomic status values.

Value

The table x with a new column containing the Townsend deprivation index.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

cdm$cohort |>
  addTownsend() |>
  select(subject_id, cohort_start_date, socio_economic_status) |>
```

```
glimpse()
```

---

addUpdatedCharlsonIndex

*Add Updated Charlson Comorbidity Index (CCI) value based on Quan et al. (2011) (*Rhrefhttps://doi.org/10.1093/aje/kwq433doi:10.1093/aje/kwq433*)*

Description

Add Updated Charlson Comorbidity Index (CCI) value based on Quan et al. (2011) (doi:10.1093/aje/kwq433)

Usage

```
addUpdatedCharlsonIndex(
  x,
  indexDate = "cohort_start_date",
  ageAdjusted = TRUE,
  conceptSet = getIndexCodelist("updated_charlson"),
  window = NULL,
  nameStyle = "updated_charlson",
  categories = NULL,
  name = tableName(x)
)
```

Arguments

x	A cdm_table containing a person identifier column named person_id or subject_id.
indexDate	A character string naming the Date column in x that defines the index date.
ageAdjusted	Logical; whether to include age adjustment in the updated Charlson Comorbidity Index.
conceptSet	A named concept set supplied as a codelist, codelist_with_details, concept_set_expression, or named list of concept IDs. It must contain congestive_heart_failure, dementia, chronic_pulmonary_disease, connective_tissue_disease, mild_liver_disease, hemiplegia, severe_chronic_kidney_disease, diabetes_with_complication, any_malignancy, moderate_or_severe_liver_disease, metastatic_solid_tumor, aids as concepts. By default, internal codelists are used.
window	NULL to use the windows defined by the index components, or a numeric vector of length two, c(min, max), to override the component windows in days relative to indexDate. Use -Inf or Inf for an unbounded lower or upper limit.
nameStyle	A character string specifying the name of the new column.

categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. An additional column with the suffix _categories is added. Intervals are evaluated in the order supplied, and missing scores are labelled missing.
name	A character string specifying the name of the output table. If NULL, a temporary table is created.

Value

The table x with a new column containing the updated Charlson Comorbidity Index value.

Examples

```
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)
library(CohortConstructor)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")
cdm$cohort <- conceptCohort(
  cdm = cdm,
  conceptSet = list(sinusitis = c(257012L, 4283893L, 4294548L, 40481087L)),
  name = "cohort"
)

# Using the internal concept sets:
cdm$cohort |>
  addUpdatedCharlsonIndex(ageAdjusted = TRUE) |>
  select(subject_id, cohort_start_date, updated_charlson) |>
  glimpse()

# This example uses custom concept sets.
customConceptSet <- list(
  congestive_heart_failure = 319835L,
  dementia = 4182210L,
  chronic_pulmonary_disease = 255573L,
  connective_tissue_disease = 4134537L,
  mild_liver_disease = 194984L,
  moderate_or_severe_liver_disease = 4212540L,
  diabetes_with_complication = 42538715L,
  hemiplegia = 374022L,
  severe_chronic_kidney_disease = 46271022L,
  any_malignancy = 4180914L,
  metastatic_solid_tumor = 432851L,
  aids = 4267414L
)

cdm$cohort |>
  addUpdatedCharlsonIndex(
    conceptSet = customConceptSet,
    nameStyle = "updated_charlson_custom"
```

```
) |>
select(subject_id, cohort_start_date, updated_charlson_custom) |>
glimpse()
```

---

charlsonCohort *Create a cohort of groups of the Charlson Comorbidity Index (CCI)*

Description

[Experimental]

Usage

```
charlsonCohort(
  cdm,
  categories,
  name,
  ageAdjusted = TRUE,
  conceptSet = getIndexCodelist("charlson"),
  subsetCohort = NULL,
  subsetCohortId = NULL
)
```

Arguments

cdm	A cdm_reference object.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. Upper and lower bounds are both included. The name will be used to name the resultant cohort.
name	Name of the new cohort table created in the cdm object.
ageAdjusted	Logical; whether to include age adjustment in the Charlson Comorbidity Index.
conceptSet	A named concept set supplied as a codelist, codelist_with_details, concept_set_expression, or named list of concept IDs. It must contain myocardial_infarction, congestive_heart_failure, peripheral_vascular_disease, cerebrovascular_disease, dementia, chronic_pulmonary_disea connective_tissue_disease, peptic_ulcer_disease, mild_liver_disease, diabetes_without_complication, hemiplegia, severe_chronic_kidney_disease, diabetes_with_complication, any_malignancy, moderate_or_severe_liver_disease, metastatic_solid_tumor, aids as concepts. By default, internal codelists are used.
subsetCohort	A character refering to a cohort table containing individuals for whom cohorts will be generated. Only individuals in this table will appear in the generated cohort. If NULL no subsetting will be performed.
subsetCohortId	Specifies cohort IDs from the subsetCohort table to include. If none are provided, all cohorts from the subsetCohort are included.

Details

CCI is based on Charlson et al. (1987) (doi:10.1016/00219681(87)901718) and Charlson et al. (1994) (doi:10.1016/08954356(94)901295) (age-adjusted) version.

Value

A cohort table.

Examples

```
## Not run:
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")

# Using the internal concept sets:
cdm$my_cohort <- charlsonCohort(
  cdm = cdm,
  categories = list(
    "none" = c(0, 0),
    "mild" = c(1, 2),
    "moderate" = c(3, 4),
    "severe" = c(5, Inf)
  ),
  name = "my_cohort"
)

cdm$my_cohort

settings(cdm$my_cohort)


## End(Not run)
```

---

electronicFrailtyIndexCohort

Create a cohort of groups of the Electronic Frailty Index (eFI)

Description

[Experimental]

Usage

```
electronicFrailtyIndexCohort(
  cdm,
  name,
 categories = list(fit = c(0, 0.12), mild = c(0.12, 0.24), moderate = c(0.24, 0.36),
    severe = c(0.36, 1)),
  conceptSet = getIndexCodelist("electronic_frailty_index"),
  subsetCohort = NULL,
  subsetCohortId = NULL
)
```

Arguments

cdm	A cdm_reference object.
name	Name of the new cohort table created in the cdm object.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. Upper and lower bounds are both included. The name will be used to name the resultant cohort.
conceptSet	A named concept set supplied as a codelist, codelist_with_details, concept_set_expression, or named list of concept IDs. It must contain activity_limitation, anemia, arthritis, atrial_fibrillation, cerebrovascular_disease, chronic_kidney_disease, diabetes, dizziness, dyspnea, falls, foot_problem, fragility_fracture, hearing_impairment, heart_failure, heart_valve_disorder, housebound, hypertension, hypotension_syncope, ischemic_heart_disease, memory_cognitive_disorder, mobility_problems, osteoporosis, parkinsonism_tremor, peptic_ulcer, peripheral_vascular_disease, care_requirement, respiratory_disease, skin_ulcer, sleep_disturbance, social_vulnerability, thyroid_disease, urinary_incontinence, urinary_system_disease, visual_impairment, weight_loss_anorexia as concepts. By default, internal codelists are used.
subsetCohort	A character refering to a cohort table containing individuals for whom cohorts will be generated. Only individuals in this table will appear in the generated cohort. If NULL no subsetting will be performed.
subsetCohortId	Specifies cohort IDs from the subsetCohort table to include. If none are provided, all cohorts from the subsetCohort are included.

Details

eFI calculation is baed on Clegg et al. (2016) (doi:10.1093/ageing/afw039)

Value

A cohort table.

Examples

```
## Not run:
library(omock)
library(duckdb)
```

```
library(OmopIndices)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")

# Using the internal concept sets:
cdm$my_cohort <- electronicFrailtyIndexCohort(
  cdm = cdm,
  name = "my_cohort"
)

cdm$my_cohort

settings(cdm$my_cohort)

## End(Not run)
```

`getIndexCodelist` *Get the codelists used for a specified index calculation*

Description

Get the codelists used for a specified index calculation

Usage

```
getIndexCodelist(index)
```

Arguments

`index`	A character string identifying the index for which to retrieve internal codelists. Supported values are `"body_mass_index"`, `"charlson"`, `"electronic_frailty_index"`, `"hospital_frailty_risk_score"`, and `"updated_charlson"`.

Value

A codelist containing the concepts used for the selected index.

Examples

```
library(OmopIndices)

getIndexCodelist("charlson")
```

updatedCharlsonCohort *Create a cohort of groups of the updated Charlson Comorbidity Index (CCI)*

Description

[Experimental]

Usage

```
updatedCharlsonCohort(
  cdm,
  categories,
  name,
  ageAdjusted = TRUE,
  conceptSet = getIndexCodelist("updated_charlson"),
  subsetCohort = NULL,
  subsetCohortId = NULL
)
```

Arguments

cdm	A cdm_reference object.
categories	A named list of numeric vectors, each containing the lower and upper bounds of a score interval. Upper and lower bounds are both included. The name will be used to name the resultant cohort.
name	Name of the new cohort table created in the cdm object.
ageAdjusted	Logical; whether to include age adjustment in the updated Charlson Comorbidity Index.
conceptSet	A named concept set supplied as a codelist, codelist_with_details, concept_set_expression, or named list of concept IDs. It must contain congestive_heart_failure, dementia, chronic_pulmonary_disease, connective_tissue_disease, mild_liver_disease, hemiplegia, severe_chronic_kidney_disease, diabetes_with_complication, any_malignancy, moderate_or_severe_liver_disease, metastatic_solid_tumor, aids as concepts. By default, internal codelists are used.
subsetCohort	A character refering to a cohort table containing individuals for whom cohorts will be generated. Only individuals in this table will appear in the generated cohort. If NULL no subsetting will be performed.
subsetCohortId	Specifies cohort IDs from the subsetCohort table to include. If none are provided, all cohorts from the subsetCohort are included.

Details

The updated CCI is based on Quan et al. (2011) (doi:10.1093/aje/kwq433).

Value

A cohort table.

Examples

```
## Not run:
library(omock)
library(duckdb)
library(OmopIndices)
library(dplyr)

cdm <- mockCdmFromDataset(datasetName = "GiBleed", source = "duckdb")

cdm$my_cohort <- updatedCharlsonCohort(
  cdm = cdm,
  categories = list(
    "none" = c(0, 0),
    "mild" = c(1, 2),
    "moderate" = c(3, 4),
    "severe" = c(5, Inf)
  ),
  name = "my_cohort"
)

cdm$my_cohort
settings(cdm$my_cohort)

## End(Not run)
```

Index

addBMI, 2
addCharlsonIndex, 3
addElectronicFrailtyIndex, 6
addEthnicity, 8
addHospitalFrailtyRiskScore, 10
addIndexOfMultipleDeprivation, 12
addLocation, 13
addPolypharmacy, 14
addPolypharmacyCount, 16
addSocioEconomicStatus, 17
addTownsend, 18
addUpdatedCharlsonIndex, 20

charlsonCohort, 22

electronicFrailtyIndexCohort, 23

getIndexCodelist, 25

updatedCharlsonCohort, 26