How to Use PhenotypeLibrary R Package

Gowtham A. Rao

2023-09-19

Contents

PhenotypeLibrary is part of HADES

1 Installation

- This is an installable R-package that may be installed as follows:

```
remotes::install_github("OHDSI/PhenotypeLibrary")
```

2 Retrieval

- The list of cohort definitions available may be retrieved as follows:

```
PhenotypeLibrary::listPhenotypes()
```

```
#> # A tibble: 569 x 55
#>    cohortId cohortName          cohortNameFormatted cohortNameLong librarian status addedVersion logi
#>       <dbl> <chr>               <chr>               <chr>          <chr>     <chr>  <chr>        <chr
#>  1        2 [W] COVID-19 diagn~ COVID-19 diagnosis~ COVID-19 diag~ rao@ohds~ Withd~ <NA>         Witl
#>  2        3 [P] Cough or Sputum Cough or Sputum     Cough or Sput~ rao@ohds~ Pendi~ <NA>         All
#>  3        4 [P] Diarrhea        Diarrhea            Diarrhea       rao@ohds~ Pendi~ <NA>         All
#>  4        5 [P] Dyspnea         Dyspnea             Dyspnea        rao@ohds~ Pendi~ <NA>         All
#>  5        6 [P] Fever           Fever               Fever          rao@ohds~ Pendi~ <NA>         All
#>  6        7 [P] Headache or He~ Headache or Headac~ Headache or H~ rao@ohds~ Pendi~ <NA>         All
#>  7        8 [P] Altered smell ~ Altered smell or t~ Altered smell~ rao@ohds~ Pendi~ <NA>         Alte
#>  8        9 [P] Sore throat     Sore throat         Sore throat    rao@ohds~ Pendi~ <NA>         All
#>  9       10 [P] Nausea or Vomi~ Nausea or Vomiting  Nausea or Vom~ rao@ohds~ Pendi~ <NA>         All
#> 10       11 [P] Malaise and or~ Malaise and or fat~ Malaise and o~ rao@ohds~ Pendi~ <NA>         All
#> # i 559 more rows
```

```
#> # i 46 more variables: isCirceJson <dbl>, contributors <chr>, contributorOrcIds <chr>, contributorOr
#> #   peerReviewers <chr>, peerReviewerOrcIds <lgl>, recommendedReferentConceptIds <chr>, ohdsiForumPo
#> #   createdDate <date>, modifiedDate <date>, lastModifiedBy <lgl>, replaces <dbl>, isReferenceCohort
#> #   censorWindowStartDate <lgl>, censorWindowEndDate <lgl>, collapseSettingsType <chr>, collapseEraP
#> #   exitStrategy <chr>, exitDateOffSetField <chr>, exitDateOffSet <dbl>, numberOfInclusionRules <dbl
#> #   qualifyingLimitType <chr>, primaryCriteriaLimit <chr>, numberOfCohortEntryEvents <dbl>, ...
```

- You can extract one or more cohort definitions into a cohortDefinitionSet object as

```
cohortDefinitionSet <- PhenotypeLibrary::getPlCohortDefinitionSet(cohortIds = c(1, 2, 3))

cohortDefinitionSet
```

```
#> # A tibble: 2 x 4
#>   cohortId cohortName                                          json
#>      <dbl> <chr>                                               <chr>
#> 1        2 [W] COVID-19 diagnosis or SARS-CoV-2 test (1pos) "{\n\t\"cdmVersionRange\" : \">=5.0.0\",'
#> 2        3 [P] Cough or Sputum                              "{\n\t\"cdmVersionRange\" : \">=5.0.0\",'
```

- cohortDefinitionSet is now a data.frame with specifications for the cohort ids 1, 2 and 3. For cohorts that conform to OHDSI Circe specifications, the field json is the cohort json specification that may be posted into your Atlas instance. The SQL is the SQL rendered from the JSON. For cohorts that do not conform to OHDSI Circe specification, only the SQL is provided and the json is left empty.

3 Use

- You can instantiate the cohorts in your environment as follows using (OHDSI/CohortGenerator)[https://github.com/OHDSI/CohortGenerator].

```
connectionDetails <-
  DatabaseConnector::createConnectionDetails(
    dbms = "postgresql",
    server = "some.server.com/ohdsi",
    user = "joe",
    password = "secret"
  )
cdmDatabaseSchema <- "cdm_synpuf"
cohortDatabaseSchema <- "scratch.dbo"
cohortTables <- CohortGenerator::getCohortTableNames()
CohortGenerator::generateCohortSet(
  connectionDetails = connectionDetails,
  cdmDatabaseSchema = cdmDatabaseSchema,
  cohortDatabaseSchema = cohortDatabaseSchema,
  cohortTableNames = cohortTables,
  cohortDefinitionSet = cohortDefinitionSet
)
```

- You can also run cohort diagnostics on this cohortDefinitionSet object as follows:

```r
databaseId <- "synpuf"

databaseName <-
  "Medicare Claims Synthetic Public Use Files (SynPUFs)"

databaseDescription <-
  "Medicare Claims Synthetic Public Use Files (SynPUFs) were created to allow interested parties to gai

CohortDiagnostics::executeDiagnostics(
  cohortDefinitionSet = cohortDefinitionSet,
  exportFolder = outputFolder,
  databaseId = databaseId,
  databaseName = databaseName,
  databaseDescription = databaseDescription,
  cohortDatabaseSchema = cohortDatabaseSchema,
  cdmDatabaseSchema = cdmDatabaseSchema,
  connectionDetails = connectionDetails,
  cohortTableNames = cohortTableNames
)
```