NCDC Data Harmonization

The NCDC project takes leverage of multiple sources of data by using a federated infrastructure. Being able to use these sources of data requires a data harmonization process to make the data interoperable.

OMOP CDM

A clinical model defines a structure and relationships that allow representing different types of clinical data. In combination with standard vocabularies, it becomes possible to achieve a higher degree of interoperability, metadata description, and sustainability.

The OMOP (Observational Medical Outcomes Partnership) CDM (Common Data Model) emerges in this context, presenting a clinical model that consistently grew to accommodate more types of clinical data. Its structure also includes standard vocabularies obtained from known sources, such as SNOMED.

The NCDC project takes leverage of the OMOP CDM model to represent the data from each source and mantain the source data. Although OMOP represents a more complex data structure, the NCDC project mainly uses the following tables to represent the data (https://ohdsi.github.io/CommonDataModel/cdm60.html):

- PERSON: "central identity management for all Persons in the database ... uniquely identify each person or patient, and some demographic information."
- OBSERVATION: "clinical facts about a Person obtained in the context of examination, questioning or a procedure."
- MEASUREMENT: "structured values (numerical or categorical) obtained through systematic and standardized examination or testing of a Person or Person's sample"
- CONDITION_OCCURRENCE: "suggesting the presence of a disease or medical condition stated as a diagnosis, a sign, or a symptom"

```
# Creating the database client using the "DBI" library
# Change the parameters accordingly
library(DBI)
library(RPostgres)
con <- dbConnect(
  RPostgres::Postgres(),
  dbname = "CDM",
  host = "localhost",
  port = 5432,
  password = password,
  user = "postgres"
)
```

Data Extraction and Querying

OMOP CDM

Some examples on how to extract data and query the OMOP CDM are given below. To write a new query it may be useful to use both the NCDC mapping information and the OMOP CDM v6.0 definition (https://ohdsi.github.io/CommonDataModel/cdm60.html).

```
# Extracting data to a dataframe
#
# Reading the table 'Person' to a dataframe and getting the average
```

```
# year of birth
df <- dbGetQuery(con, 'SELECT * FROM PERSON')
# summary(df)
mean(df$year_of_birth)
```

```
## [1] 1931.75
```

```
# Another use case would be to already use the SQL query to make a sub-selection
# of the data to retrieve. For example, extracting all observation for persons
# with more than 75 years and a negative dementia diagnosis
df <- dbGetQuery(con, "SELECT * FROM OBSERVATION AS o WHERE o.person_id IN
    (SELECT DISTINCT p.person_id FROM PERSON AS p JOIN OBSERVATION as o ON
    p.person_id = o.person_id WHERE o.observation_concept_id = '4182210' AND
    (date_part('year', now()) - p.year_of_birth) > 75);")
```

```
# Select the max, min, and average year of birth for all persons in the database by gender
#
# Gender concept id:
#   - 8532: Female
#   - 8551: Unknown
#   - 8507: Male
sql_statement <- "SELECT gender_concept_id, COUNT(person_id),
MAX(year_of_birth), MIN(year_of_birth), AVG(year_of_birth) FROM
PERSON GROUP BY gender_concept_id"
query <- dbSendQuery(con, sql_statement)
dbFetch(query)
```

```
##   gender_concept_id count  max  min  avg
## 1              8507     1 1914 1914 1914
## 2              8551     1 1943 1943 1943
## 3              8532     2 1941 1929 1935
```

```
dbClearResult(query)
```

```
# Selecting the average age for all persons with a dementia diagnosis
#
# Condition with concept id 4182210 from SNOMED "Dementia"
sql_statement <- "SELECT AVG(date_part('year', now()) - p.year_of_birth)
FROM PERSON AS p INNER JOIN CONDITION_OCCURRENCE AS c ON p.person_id = c.person_id
WHERE c.condition_concept_id = '4182210'"
query <- dbSendQuery(con, sql_statement)
dbFetch(query)
```

```
##   avg
## 1  78
```

```
dbClearResult(query)
```

Simplified Table

One of the drawbacks of using a clinical model can be the higher complexity in its model definition. This is the case with the OMOP CDM, it requires more knowledge about its schema and the querying can be more difficult, especially when taking the first steps. Although we recommend using the OMOP CDM, it's also possible to use a simplified table that mimics most of the representations used from the source data, a plane table with an entry by visit.

```r
# Extracting data to a pandas dataframe
#
# Reading the table 'NCDC' to a pandas dataframe and getting the average
# year of birth
df <- dbGetQuery(con, 'SELECT * FROM NCDC')
mean(df[!duplicated(df['id']),]$birth_year)
```

```
## [1] 1931.75
```

```r
# Extracting all observation for persons with more than 75 years and a
# negative dementia diagnosis
df <- dbGetQuery(con, "SELECT * FROM NCDC2 WHERE dementia_diagnosis IS FALSE AND
    (date_part('year', now()) - birth_year) > 75")
```

```r
# Select the max, min, and average year of birth for all persons in the database by gender
#
# NCDC coding:
#  - 0: Male
#  - 1: Female
#  - NULL: Unknown
sql_statement <- "SELECT t.sex, COUNT(t.id),
MAX(t.birth_year), MIN(t.birth_year), AVG(t.birth_year) FROM
(SELECT DISTINCT id, sex, birth_year FROM ncdc) AS t
GROUP BY t.sex"
query <- dbSendQuery(con, sql_statement)
dbFetch(query)
```

```
##   sex count  max  min  avg
## 1  NA     1 1943 1943 1943
## 2   0     1 1914 1914 1914
## 3   1     2 1941 1929 1935
```

```r
dbClearResult(query)
```

```r
# Selecting the average age for all persons with a dementia diagnosis
#
# NCDC variable for dementia: "dementia_diagnosis"
# NCDC coding: TRUE ('1'), FALSE ('0')
sql_statement <- "SELECT AVG(date_part('year', now()) - birth_year) FROM ncdc
  WHERE id IN (SELECT DISTINCT id FROM ncdc
  WHERE dementia_diagnosis IS TRUE)"
# Alternative: dementia_diagnosis = '1'
query <- dbSendQuery(con, sql_statement)
dbFetch(query)
```

```
##   avg
## 1  78
```

```r
dbClearResult(query)
```