/\*\*\*

Title: step4b\_hospnames

Description: Add the name of the hospital providing service (at this time, this is the 1st provider in a case; ignores info on transfer provider)

TK: Ask client what info they want to see on the transfer facility

Version Control: https://dsghe.lacare.org/nblume/Readmissions/tree/master/Code/Data\_Acquisition\_and\_Understanding/Cloudera%20DSW/Iteration2/

Data Source: nathalie.prjrea\_step4a\_demog

plandata.provider

Output: nathalie.prjrea\_step4b\_hospitals is the main data set, not aggregated

nathalie.prjrea\_tblo\_index\_hospitals: rates are aggregated by index hospital

nathalie.prjrea\_tblo\_readmit\_hospitals: rates are aggregated by readmitting hospital

Issues: In QNXT's provider table, the provid field matches many but not all cases.

A greater number of matches to QNXT.provider's fedid field exist, but there are duplicate fedid values that point simultaneously to

hospitals and to individual providers. The provtype field may help specify hospitals, but the values are specific to smaller subsets

that need to be unified into a single 'hospital' group. For provtype, 88=snf, 15=Community Hospital - Outpatient, 46=Rehab Clinic,

70=Acute Psychiatric Hospital - Institution For Mental Disease , 16=Community Hospital - Inpatient

The fact that there are SNFs indicates that I have a capture problem further upstream.

\*\*\*/

drop table if exists nathalie.prjrea\_step4b\_hospitals

;

set max\_row\_size = 7mb

;

create table nathalie.prjrea\_step4b\_hospitals

as

select A.\*, PROVNAME\_REF.hospname

from nathalie.prjrea\_step4a\_demog as A

left join

(

select case\_id, fullname as hospname

from

(

select case\_id, provider, fullname, source, row\_number() over(partition by case\_id order by source asc) as rn

from

(

--A provid match is prefered and is taken to be unique.

select case\_id, A.provider, B.fullname, 1 as source

from nathalie.prjrea\_step4a\_demog as A

left join plandata.provider as B

on A.provider = B.provid

union

--In the absence of a provid match, backup matches to fedid are used. However each fedid may be associated with several names in

--the reference file. Below provtype is ranked so that a SNF name is attached to the data set where several names may have been

--associated to the same prov code in the reference file.

--Recall that: For provtype, 88=snf, 15=Community Hospital - Outpatient, 46=Rehab Clinic, 70=Acute Psychiatric Hospital.

select case\_id, A.provider, B.fullname, 2 as source

from nathalie.prjrea\_step4a\_demog as A

left join plandata.provider as B

on A.provider = B.fedid

where B.provtype in ('88') --this is assigned a higher source value because you want to preserve SNF info as much as possible

union

select case\_id, A.provider, B.fullname, 3 as source

from nathalie.prjrea\_step4a\_demog as A

left join plandata.provider as B

on A.provider = B.fedid

where B.provtype in ('16', '70') --this is assigned the next highest source value so that potential inpatient hosp. that are not rehab are preserved

union

select case\_id, A.provider, B.fullname, 4 as source

from nathalie.prjrea\_step4a\_demog as A

left join plandata.provider as B

on A.provider = B.fedid

where B.provtype in ('15', '46')

union

select case\_id, A.provider, B.fullname, 5 as source -- there is some kind of match to a fullname

from nathalie.prjrea\_step4a\_demog as A

left join plandata.provider as B

on A.provider = B.fedid

where B.provtype not in ('88', '70', '16', '15', '46')

and B.fullname is not null

union

select case\_id, A.provider, A.provider as fullname, 6 as source -- there is no match to a fullname, but A.provider may not be null

from nathalie.prjrea\_step4a\_demog as A

left join plandata.provider as B

on A.provider = B.fedid

where B.provtype not in ('88', '70', '16', '15', '46')

and B.fullname is null

) S

where fullname is not null --unnecessary since a name is guaranteed in the inner query

) S2

where rn = 1

) PROVNAME\_REF

on A.case\_id = PROVNAME\_REF.case\_id

;

set max\_row\_size = 1mb

;

/\*

EXCLUDE CASES

--Exclude CCI --> for more accurate report of readmission rates. Counts need to come from unfiltered data set.

\*/

drop table if exists nathalie.tmp\_no\_cci

;

create table nathalie.tmp\_no\_cci

as

select \*

from nathalie.prjrea\_step4b\_hospitals

where segment != 'CCI'

;

/\*

SUMMARIZE: INDEX HOSPITALS

-- Counts and rates by hospital where the index admit took place (before the readmit)

\*/

drop table if exists nathalie.prjrea\_tblo\_index\_hospitals

;

create table nathalie.prjrea\_tblo\_index\_hospitals

as

select A.hospname as index\_acute\_inpatient\_facility, admit\_count, readmission\_rate as no\_cci\_readmission\_rate, admit\_count \* readmission\_rate as calculated\_readmit\_count

from

(

select

hospname

, count(\*) as admit\_count

from prjrea\_step4b\_hospitals

group by hospname

) as A

left join

( -- rates are derived without cci for accuracy. cci subpopulation may change admit outside LACare, so their admit count is under-reported.

select

hospname

, sum(is\_followed\_by\_a\_30d\_readmit) as number\_readmits

, count(\*) as number\_all\_admits

, round(sum(is\_followed\_by\_a\_30d\_readmit) / count(\*), 2) as readmission\_rate

from nathalie.tmp\_no\_cci

group by hospname

) as B

on A.hospname = B.hospname

;

/\*

SUMMARIZE: READMITTING HOSPITALS

-- Counts and rates by hospital that receive members who were admitted elsewhere in the last 30 days

\*/

drop table if exists nathalie.prjrea\_tblo\_readmitting\_hospitals

;

create table nathalie.prjrea\_tblo\_readmitting\_hospitals

as

select A.hospname as readmitting\_acute\_inpatient\_facility, admit\_count, readmission\_rate as no\_cci\_readmission\_rate, admit\_count \* readmission\_rate as calculated\_readmit\_count

from

(

select

hospname

, count(\*) as admit\_count

from prjrea\_step4b\_hospitals

group by hospname

) as A

left join

( -- rates are derived without cci for accuracy. cci subpopulation may change admit outside LACare, so their admit count is under-reported.

select

hospname

, sum(is\_a\_30d\_readmit) as number\_readmits

, count(\*) as number\_all\_admits

, round(sum(is\_a\_30d\_readmit) / count(\*), 2) as readmission\_rate

from nathalie.tmp\_no\_cci

group by hospname

) as B

on A.hospname = B.hospname

;