SQL_reperage_cond_med

1 Description

Repérage d'une condition médicale

2 Usage

```
NomTable <- SQL_reperage_cond_med(
  conn = SQL_connexion(),
  debut, fin,
  Dx_table,
  CIM = c("CIM9", "CIM10"),
  by_Dx = FALSE,
  date_dx_var = "admis",
  n1 = 30, n2 = 730
)
```

3 Arguments

3.1 conn

Variable contenant la connexion entre R et Teradata. Voir la documentation de la fonction `SQL_connexion()`.

```
conn = SQL_connexion("msXXX")
```

3.2 debut & fin

Date de début et de fin de la période d'étude au format `AAAA-MM-JJ`.

```
debut = "2021-01-01", fin = "2021-12-31"
```

3.3 Dx_table

`list` contenant les codes de diagnostics à l'étude.

```
Dx_table = list(
  diabete = list(CIM9 = c("2504%", "2505%", "2506%", "2507%"),
                 CIM10 = c("E102%", "E103%", "E104%", "E105%", "E107%")),
  diabete_complique = list(CIM9 = c("250%"),
                           CIM10 = c("E100%", "E101%", "E106%", "E108%", "E109%"))
)
```

3.4 CIM

Sélection entre `CIM9`, `CIM10` ou les deux.

```
CIM = "CIM9"  # CIM9 seulement
CIM = "CIM10"  # CIM10 seulement
CIM = c("CIM9", "CIM10")  # CIM9 et CIM10
```

3.5 by_Dx

Permet des études de sensibilité sans avoir besoin de réécrire la liste des codes. Voir ci-dessous en exemple les équivalences en utilisant `by_Dx` au lieu de tout réécrire le code.

```
Dx_table = list(
  diabete = list(CIM9 = c("2504%", "2505%", "2506%", "2507%"),
                 CIM10 = c("E102%", "E103%", "E104%", "E105%", "E107%")),
  diabete_complique = list(CIM9 = c("250%"),
                           CIM10 = c("E100%", "E101%", "E106%", "E108%", "E109%"))
),
by_Dx = FALSE

# Revient à écrire :
Dx_table = list(
  diabete_tout = list(
    CIM9 = c("2504%", "2505%", "2506%", "2507%",
             "250%"),
    CIM10 = c("E102%", "E103%", "E104%", "E105%", "E107%",
              "E100%", "E101%", "E106%", "E108%", "E109%"))
),
by_Dx = TRUE  # ou FALSE, pas d'importance, car un seul type de condition médicale
```

3.6 date_dx_var

Indique si on utilise la date d'admission (`admis`) ou la date de départ (`depar`) comme date de diagnostic lors de l'extraction des données. Voir la section *Processus*, *Étape 1* et *Étape 2* pour voir l'impact sur le code SQL généré.

```
date_dx_var = "admis"  # date d'amission
date_dx_var = "depar"  # date de départ
```

3.7 n1 & n2

Nombre entier. Permet de construire l'intervalle `[n1, n2]`. Pour qu'un code de diagnostic soit confirmé, il faut que *DIAGN{i}* soit suivi de *DIAGN{j}* (où `i` < `j`) et que le nombre de jours entre `DIAGN{j} - DIAGN{i}` soit dans cet intervalle.

```
n1 = 30, n2 = 730
```

4 Résultat

Variable	Descriptif
ID	Numéro d'identification de l'individu.
DIAGN	Nom du diagnostic inscrit dans l'argument Dx_table. Selon l'exemple de la section *Arguments*, on y verrait les valeurs diabete et diabete_complique. **Présente seulement si by_Dx=TRUE**.
DI_Finale	Date d'incidence retenue.
DI_Hospit	Date d'incidence des hospitalisations
DI_Acte	Date d'incidence des actes
DC_Acte	Date qui confirme la date d'incidence de l'acte `DI_Acte` selon l'intervalle [n1, n2].
D_Recent	Date du diagnostic le plus récent dans l'intervalle `[debut, fin]` dans les trois banques (BDCU, MED-ÉCHO et SMOD) sans tenir compte de l'algorithme.

5 Processus

Supposons les arguments suivants :

```
debut = "2021-01-01"
fin = "2021-12-31"
Dx_table = list(
  code1 = list(CIM9 = c("2504%", "2505%", "2506%"),
               CIM10 = c("E102%", "E103%", "E104%"))
)
n1 = 30
n2 = 730
```

5.1 Étape 1 — MED-ÉCHO

Pour chaque personne, repérer la 1^{re} admission parmi toutes ses hospitalisations qui contient un diagnostic de la condition médicale. Rechercher dans les diagnostics hsopitaliers suivants : Principal (P), Secondaires (S) et Services et décès (D).

5.1.1 V_DIAGN_SEJ_HOSP_CM

5.1.1.1 date_dx_var = "admis" & CIM = c("CIM9", "CIM10")

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_ADMIS_SEJ_HOSP as DATE_DX
from RES_SSS.V_DIAGN_SEJ_HOSP_CM
where SHOP_DAT_ADMIS_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and  (
            (SHOP_COD_DIAGN_MDCAL_CLINQ like any ('E102%', 'E103%', 'E104%')
              and SHOP_NO_SEQ_SYS_CLA = 1)
          or
            (SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%')
              and SHOP_NO_SEQ_SYS_CLA = 4)
          )
    and SHOP_TYP_DIAGN_SEJ_HOSP in ('P', 'S', 'D');
```

5.1.1.2 date_dx_var = "depar" & CIM = "CIM9"

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_DEPAR_SEJ_HOSP as DATE_DX
from RES_SSS.V_DIAGN_SEJ_HOSP_CM
where SHOP_DAT_DEPAR_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%')
    and SHOP_NO_SEQ_SYS_CLA = 4
    and SHOP_TYP_DIAGN_SEJ_HOSP in ('P', 'S', 'D');
```

5.1.1.3 date_dx_var = "admis" & CIM = "CIM10"

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_ADMIS_SEJ_HOSP as DATE_DX
from RES_SSS.V_DIAGN_SEJ_HOSP_CM
where SHOP_DAT_ADMIS_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('E102%', 'E103%', 'E104%')
    and SHOP_NO_SEQ_SYS_CLA = 1
    and SHOP_TYP_DIAGN_SEJ_HOSP in ('P', 'S', 'D');
```

5.1.2 V_SEJ_SERV_HOSP_CM

5.1.2.1 date_dx_var = "admis"

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_ADMIS_SEJ_HOSP as DATE_DX
from RES_SSS.V_SEJ_SERV_HOSP_CM
where SHOP_DAT_ADMIS_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%', 'E102%', 'E103%', 'E104%');
```

5.1.2.2 date_dx_var = "depar"

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_DEPAR_SEJ_HOSP as DATE_DX
from RES_SSS.V_SEJ_SERV_HOSP_CM
where SHOP_DAT_DEPAR_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%', 'E102%', 'E103%', 'E104%');
```

5.2 Étape 2 — BDCU, SMOD et MED-ÉCHO

Pour chaque personne, repérer le 1^{er} diagnostic de la condition médicale inscrit à un des trois fichiers si celui-ci est suivi d'un autre diagnostic de la condition médicale inscrit à un de ces mêmes trois fichiers dans un intervalle d'au moins `n1` jours et de moins de `n2` jours.
Pour MED-ÉCHO, rechercher dans tous les diagnostics hospitaliers : Admission (A), Principal (P), Secondaires (S) et Services et décès (D).

5.2.1 V_DIAGN_SEJ_HOSP_CM — MED-ÉCHO

5.2.1.1 date_dx_var = "admis" & CIM = c("CIM9", "CIM10")

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_ADMIS_SEJ_HOSP as DATE_DX
from RES_SSS.V_DIAGN_SEJ_HOSP_CM
where SHOP_DAT_ADMIS_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and  (
            (SHOP_COD_DIAGN_MDCAL_CLINQ like any ('E102%', 'E103%', 'E104%')
              and SHOP_NO_SEQ_SYS_CLA = 1)
          or
            (SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%')
              and SHOP_NO_SEQ_SYS_CLA = 4)
          )
    and SHOP_TYP_DIAGN_SEJ_HOSP in ('A', 'P', 'S', 'D');
```

5.2.1.2 date_dx_var = "depar" & CIM = "CIM9"

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_DEPAR_SEJ_HOSP as DATE_DX
from RES_SSS.V_DIAGN_SEJ_HOSP_CM
where SHOP_DAT_DEPAR_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%')
    and SHOP_NO_SEQ_SYS_CLA = 4
    and SHOP_TYP_DIAGN_SEJ_HOSP in ('A', 'P', 'S', 'D');
```

5.2.1.3 date_dx_var = "admis" & CIM = "CIM10"

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_ADMIS_SEJ_HOSP as DATE_DX
from RES_SSS.V_DIAGN_SEJ_HOSP_CM
where SHOP_DAT_ADMIS_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('E102%', 'E103%', 'E104%')
    and SHOP_NO_SEQ_SYS_CLA = 1
    and SHOP_TYP_DIAGN_SEJ_HOSP in ('A', 'P', 'S', 'D');
```

5.2.2 V_SEJ_SERV_HOSP_CM — MED-ÉCHO

5.2.2.1 date_dx_var = "admis"

```
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_ADMIS_SEJ_HOSP as DATE_DX
from RES_SSS.V_SEJ_SERV_HOSP_CM
where SHOP_DAT_ADMIS_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%', 'E102%', 'E103%', 'E104%');
```

5.2.2.2 date_dx_var = "depar"

```sql
select SHOP_NO_INDIV_BEN_BANLS as ID,
       SHOP_DAT_DEPAR_SEJ_HOSP as DATE_DX
from RES_SSS.V_SEJ_SERV_HOSP_CM
where SHOP_DAT_DEPAR_SEJ_HOSP between '2021-01-01' and '2021-12-31'
    and SHOP_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%', 'E102%', 'E103%', 'E104%');
```

5.2.3 V_EPISO_SOIN_DURG_CM — BDCU

5.2.3.1 date_dx_var = "admis"

```sql
select SURG_NO_INDIV_BEN_BANLS as ID,
       SURG_DHD_EPISO_SOIN_DURG as DATE_DX
from RES_SSS.V_EPISO_SOIN_DURG_CM
where SURG_DHD_EPISO_SOIN_DURG between to_date('2021-01-01') and to_date('2021-12-31')
    and SURG_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%', 'E102%', 'E103%', 'E104%');
```

5.2.3.2 date_dx_var = "depar"

```sql
select SURG_NO_INDIV_BEN_BANLS as ID,
       SURG_DH_DEPAR_USAG_DURG as DATE_DX
from RES_SSS.V_EPISO_SOIN_DURG_CM
where SURG_DH_DEPAR_USAG_DURG between to_date('2021-01-01') and to_date('2021-12-31')
    and SURG_COD_DIAGN_MDCAL_CLINQ like any ('2504%', '2505%', '2506%', 'E102%', 'E103%', 'E104%');
```

5.2.4 I_SMOD_SERV_MD_CM — SMOD

```sql
select SMOD_NO_INDIV_BEN_BANLS as ID,
       SMOD_DAT_SERV as DATE_DX
from PROD.I_SMOD_SERV_MD_CM
where SMOD_DAT_SERV between '2021-01-01' and '2021-12-31'
    and SMOD_COD_DIAGN_PRIMR like any ('2504%', '2505%', '2506%', 'E102%', 'E103%', 'E104%')
    and SMOD_COD_STA_DECIS = 'PAY';
```

5.2.5 Confirmation des diagnostics

Exemples où `n1=30` et `n2=730`.

5.2.5.1 Exemple 1

Avant

ID	DATE_DX
1	2020-01-01
1	2020-01-20
1	2020-02-05

Après

ID	DATE_REP	DATE_CONF1
1	2020-01-01	2020-02-05

5.2.5.2 Exemple 2

Avant

ID	DATE_DX
1	2018-01-01
1	2018-01-20
1	2020-06-15
1	2020-07-25
1	2020-09-05

Après

ID	DATE_REP	DATE_CONF1
1	2020-06-15	2020-07-25
1	2020-07-25	2020-09-05