

Description

Confirmation d'un diagnostic par d'autres diagnostics lorsque ceux-ci se retrouvent dans un intervalle précis.

Usage

```
confirm_2Dx(  
  dt,  
  ID,  
  DATE,  
  DIAGN = NULL,  
  study_start = NULL,  
  study_end = NULL,  
  n1 = 30,  
  n2 = 730,  
  keep_first = FALSE,  
  reverse = FALSE  
)
```

```
confirm_3Dx(  
  dt,  
  ID,  
  DATE,  
  DIAGN = NULL,  
  study_start = NULL,  
  study_end = NULL,  
  n1 = 30,  
  n2 = 730,  
  keep_first = FALSE,  
  reverse = FALSE  
)
```

Arguments

<code>dt</code>	Table contenant les dates de diagnostics des individus.
<code>ID</code>	Nom de la colonne contenant le numéro d'identification unique des individus.
<code>DATE</code>	Nom de la colonne contenant la date du diagnostic.
<code>DIAGN</code>	Facultatif. Nom de la colonne indiquant les codes de diagnostics.
<code>study_start</code>	Date de début de la période d'étude contenant les dates de repérage . Si NULL, aura pour valeur la première date de <code>dt</code> , la plus ancienne.

<code>study_end</code>	Date de fin de la période d'étude contenant les dates de repérage . Si NULL, aura pour valeur la dernière date de <code>dt</code> , la plus récente.
<code>n1, n2</code>	Nombre de jours permettant de construire l'intervalle <code>[n1; n2]</code> où un code de diagnostic peut en confirmer un autre.
<code>keep_first</code>	TRUE ou FALSE. Permet d'arrêter le processus si on veut conserver la première date qui est confirmée par une autre dans l'intervalle <code>[n1; n2]</code> . Accélère le processus en évitant de confirmer d'autres dates inutilement.
<code>reverse</code>	TRUE ou FALSE. Si on doit faire la vérification en prenant la date la plus récente et en reculant dans le temps.

Examples

```
dt_ex <- data.frame(
  id = 1L,
  dates = c('2020-01-01', '2020-01-09', '2020-01-10', '2020-01-15', '2020-01-16',
            '2020-01-20', '2020-01-26', '2020-01-31')
)
ex_2dx <- confirm_2Dx(dt = dt_ex, ID = 'id', DATE = 'dates', DIAGN = NULL,
  n1 = 10, n2 = 20, reverse = FALSE)
ex_2dx_reverse <- confirm_2Dx(dt = dt_ex, ID = 'id', DATE = 'dates', DIAGN = NULL,
  n1 = 10, n2 = 20, reverse = TRUE)
ex_3dx <- confirm_3Dx(dt = dt_ex, ID = 'id', DATE = 'dates',
  n1 = 10, n2 = 20, reverse = FALSE)
ex_3dx_reverse <- confirm_3Dx(dt = dt_ex, ID = 'id', DATE = 'dates', DIAGN = NULL,
  n1 = 10, n2 = 20, reverse = TRUE)

### Avec argument DIAGN
dt_ex_dx <- data.frame(
  id = 1L,
  dates = c('2020-01-01', '2020-01-09', '2020-01-10', '2020-01-15', '2020-01-16',
            '2020-01-20', '2020-01-26', '2020-01-31'),
  dx = c(rep('diab', 4), rep('canc', 4))
)
ex_2dx_diagn <- confirm_2Dx(dt = dt_ex_dx, ID = 'id', DATE = 'dates', DIAGN = 'dx',
  n1 = 10, n2 = 20, reverse = FALSE)

### study_start & study_end
ex_studydates <- confirm_2Dx(dt = dt_ex, ID = 'id', DATE = 'dates', DIAGN = NULL,
  study_start = '2020-01-10', study_end = '2020-01-20',
  n1 = 10, n2 = 20, reverse = FALSE)
ex_studydates_rev <- confirm_2Dx(dt = dt_ex, ID = 'id', DATE = 'dates', DIAGN = NULL,
  study_start = '2020-01-10', study_end = '2020-01-20',
  n1 = 10, n2 = 20, reverse = TRUE)
```