Analyse préparatoire pour PDI : MIMIC IV

Informations générales

Sommaire

liste des colonnes

- `subject_id` : int4 (FK)
- `hadm_id` : int4 (FK)
- `pharmacy_id` : int4 (PK)
- `poe_id` : varchar(25)
- `poe_seq` : int4
- `order_provider_id` : varchar(10)
- `starttime` : timestamp
- `stoptime` : timestamp
- `drug_type` : varchar(20) (PK)
- `drug` : varchar(255) (PK)
- `formulary_drug_cd` : varchar(50)
- `gsn` : varchar(255) (En pratique : entier de 6 chiffres commençant par un 0)
- `ndc` : varchar(25) (En pratique : entier de 11 chiffres)
- `prod_strength` : varchar(255)
- `dose_val_rx` : varchar(100) (En pratique : un float)
- `dose_unit_rx` : varchar(50)
- `form_val_disp` : varchar(50) (En pratique : un float)
- `form_unit_disp` : varchar(50)
- `doses_per_24_hrs` : float4
- `route` : varchar(50)

→15,416,708 lignes de prescriptions identifiées par patient_id et numéro d'hospitalisation

Colonnes `poe_id`, `poe_seq` et `order_provider_id`

Données contenant les traces du donneur d'ordre

Identification des produits

Colonne `drug_type`

- `ADDITIVE` : 23,891
- `BASE`: 2,642,933
- `MAIN`: 12,749,884

Colonne `drug`

> Contient les noms des médications (langage humain, anglosaxon)

- 9,614 valeurs uniques non nulles
- clé mais valeurs impropres (caractère vide en position 0, différences à la casse, fautes, etc...)

Colonne `formulary_drug_cd`

Contient les codes des medications (alphanumérique, souvent non intelligible)

> → 3,804 valeurs uniques → 18,957 valeurs nulles

Plusieurs noms de médications par code, parfois jusqu'à 25/30 Plusieurs codes par medication, aussi jusqu'à 25/30

> - il n'est pas possible de servir d'un des champs pour généraliser l'autre.
> - compliqué pour se débarasser de la variabilité des champs textuels

---

Quantités et dispensation

Quantités de médicament

Colonne `prod_strength`

Contient le dosage du médicament prescrit pour l'unité d'encapsulation (format et quantité de principe actif)

- ex : `30 mg Tab`

Colonnes `dose_val_rx` et `dose_unit_rx`

Contiennent la quantité de médicament requise par le patient et son unité

- ex : `60`, `mg`

Colonnes `form_val_disp` et `form_unit_disp`

Contiennent le nombre et la nature des unités à être dispensées, calculées grâce au format sous lequel se présente le médicament(`prod_strength`) et aux quantités prescrites au patient (`dose_val_rx` et `dose_unit_rx`)

- ex : `2`, `TAB` pour "deux comprimés"

Fréquence de dispensation

Colonne `doses_per_24_hrs`

Donne le nombre de répétitions par jour de la dose calculée précédemment

> A noter : la colonne `form_unit_disp` peut aussi déjà contenir des unités incluant une dimension temporelle (ex : `UNIT/HR`, `mg/day`)

- ex : `4`

Colonne `route`

Contient la route d'administration du médicament
(99 valeurs distinctes)

- ex : `ORAL`

Positionnement

> Je souhaite travailler sur les erreurs de prescription liées à l'interaction entre les médicaments prescrits.

Piste :

- Preprocessing
  - travailler la colonne `prod_strength` pour déterminer leq quantités et formats de dispensation
  - normaliser les valeurs des champs textuels qui induisent trop de variabilité
- Modélisation
  - reformater les données pour couvrir toutes les substances prescrites lors d'un séjour à l'hopital sur une ligne et non plusieurs
  - Utiliser en prmier lieu les modèles prouvés efficaces pour le OCC (RandomForest, XGBoost), et explorer d'autres si besoin