Prédiction de réadmission des patients diabétiques à l'aide de DataWarehouse

Préparé par : AABBOU Zakaria
FILALI Amine

Encadré par : Guénaël Cabanes

Année universitaire: 2020/2021

SOMMAIRE

TABLE DE FIGURES

I. INTRODUCTION

Dans le cadre du cours (DataWarehouse) enseigné par notre cher prof **Guénaël Cabanes**, et dans le cadre du projet de fin de formation, nous sommes amenés à concevoir et développer un système décisionnel (DataWarehouse) sur le diabète afin de prédire les réadmissions des patients diabétiques et appuyer la prise de décisions dans le secteur de la santé.

Le diabète sucré (DS) est une maladie chronique où le sang a un taux de sucre élevé. Cela peut survenir lorsque le pancréas ne produit pas suffisamment d'insuline ou lorsque le corps ne peut pas utiliser efficacement l'insuline qu'il produit (OMS). Le diabète est une maladie évolutive qui peut entraîner un nombre important de complications de santé et réduire profondément la qualité de vie. Alors que de nombreux patients diabétiques gèrent la complication de santé avec un régime et de l'exercice, certains nécessitent des médicaments pour contrôler leur glycémie. Tel que publié par un article de recherche intitulé «La relation entre le diabète sucré et les taux de réadmission à 30 jours», on estime que 9,3% de la population aux États-Unis souffre de diabète sucré (DM), dont 28% ne sont pas diagnostiqués. Ces dernières années, les agences gouvernementales et les systèmes de santé se sont de plus en plus concentrés sur les taux de réadmission à 30 jours pour déterminer la complexité de leurs populations de patients et améliorer la qualité. Les taux de réadmission à 30 jours pour les patients hospitalisés atteints de DM se situent entre 14,4 et 22,7%, bien plus élevé que le taux de tous les patients hospitalisés (8,5 à 13,5%).

La réadmission à l'hôpital est un indicateur de la qualité des soins et un facteur de l'augmentation du coût des soins de santé. Comme d'autres maladies chroniques, le diabète est associé à un risque plus élevé de réadmission à l'hôpital. Dans cette recherche, nous allons construire un DataWarehouse à partir des données transactionnelles fournis par « Center for Clinical and Translational Research ». Après, nous évaluons plusieurs approches d'apprentissage automatique pour prédire la probabilité de réadmission des patients diabétiques à l'hôpital. L'ensemble de données utilisé pour cette étude contient plus de 100 000 données sur les patients diabétiques et 55 variables, y compris la durée du séjour, l'insuline et les visites des patients hospitalisés dans les hôpitaux des États-Unis. Nous exploitons plusieurs techniques de pré-traitement et étudions les performances des différents modèles. Les variables importantes qui contribuent à l'analyse sont le nombre de patients hospitalisés, la durée du séjour, le nombre de médicaments, le nombre de diagnostics et l'âge. Les résultats démontrent la viabilité des techniques permettant de mieux comprendre les facteurs influençant la réadmission à l'hôpital.

- Outils et technologies utilisés:

DataWarhouse: Oracle, PL/SQL

IDE: SQL developer, Jupyter Notebook

Visualisation: Python (Matplotlib and Seaborn)

Modèles: Logistic Regression, Decision Tree, Random Forest, SVM

Biblios: Numpy, Pandas, Seaborn, Matplotlib, Scikit-learn

1. Problématique

Identifier les facteurs qui conduisent au taux élevé de réadmission des patients diabétiques dans les 30 jours suivant leur sortie et prédire en conséquence les patients diabétiques à haut risque qui sont les plus susceptibles d'être réadmis dans les 30 jours afin d'améliorer la qualité des soins, l'expérience des patients, la santé de la population et de réduire les coûts en réduisant les taux de réadmission. Aussi, pour identifier les médicaments qui sont les plus efficaces dans le traitement du diabète.

2. Impact sur le business

La réadmission à l'hôpital est un facteur important des dépenses médicales totales et est un indicateur émergent de la qualité des soins. Le diabète, comme d'autres problèmes de santé chroniques, est associé à un risque accru de réadmission à l'hôpital. Comme mentionné dans l'article « Correction to: Hospital Readmission of Patients with Diabetes», la réadmission à l'hôpital est une mesure de qualité des soins de santé hautement prioritaire et un objectif de réduction des coûts, en particulier dans les 30 jours suivant la sortie. Le fardeau du diabète chez les patients hospitalisés est considérable, croissant et coûteux, et les réadmissions contribuent pour une part importante à ce fardeau. La réduction des taux de réadmission chez les patients diabétiques a un potentiel de réduire considérablement les coûts des soins de santé tout en améliorant simultanément les soins.

Notre objectif est de fournir des informations sur les facteurs de risque de réadmission et d'identifier les médicaments les plus efficaces dans le traitement du diabète.

3. Ensemble de données et domaine

Le sous-ensemble de données utilisé pour l'analyse couvre 10 ans de données sur les rencontres de patients diabétiques (1999 - 2008) parmi 130 hôpitaux américains comptant plus de 100 000 patients diabétiques. De plus, toutes les rencontres utilisées pour l'analyse répondent à cinq critères clés:

- C'est une hospitalisation.
- Le patient hospitalisé a été classé comme diabétique (au moins un des trois diagnostics initiaux incluait le diabète).
- La durée du séjour était comprise entre 1 et 14 jours.
- Le patient hospitalisé a subi des tests de laboratoire.
- Le patient hospitalisé a reçu des médicaments pendant son séjour.

Data Source: UCI Dataset Link

4. Dictionnaire de données

Voici une description des données utilisées dans ce projet telles que décrites dans la source:

Nom du variable	Type	Description et valeurs	% manquant
Encounter ID	Numeric	Unique identifier of an encounter	0%
Patient number	Numeric	Unique identifier of a patient	0%

Race	Nominal	Values: Caucasian, Asian, African American, Hispanic, and other	2%
Gender	Nominal	Values: male, female, and unknown/invalid	0%
Age	Nominal	Grouped in 10-year intervals: 0, 10), 10, 20), …, 90, 100)	0%
Weight	Numeric	Weight in pounds.	97%
Admission type	Nominal	Integer identifier corresponding to 9 distinct values, for example, emergency, urgent, elective, newborn, and not available	0%
Discharge disposition	Nominal	Integer identifier corresponding to 29 distinct values, for example, discharged to home, expired, and not available	0%
Admission source	Nominal	Integer identifier corresponding to 21 distinct values, for example, physician referral, emergency room, and transfer from a hospital	0%
Time in hospital	Numeric	Integer number of days between admission and discharge	0%
Payer code	Nominal	Integer identifier corresponding to 23 distinct values, for example, Blue Cross/Blue Shield, Medicare, and self-pay	52%
Medical specialty	Nominal	Integer identifier of a specialty of the admitting physician, corresponding to 84 distinct values, for example, cardiology, internal medicine, family/general practice, and surgeon	53%
Number of lab procedures	Numeric	Number of lab tests performed during the encounter	0%
Number of procedures	Numeric	Number of procedures (other than lab tests) performed during the encounter	0%
Number of medications	Numeric	Number of distinct generic names administered during the encounter	0%
Number of outpatient visits	Numeric	Number of outpatient visits of the patient in the year preceding the encounter	0%
Number of emergency visits	Numeric	Number of emergency visits of the patient in the year preceding the encounter	0%
Number of inpatient visits	Numeric	Number of inpatient visits of the patient in the year preceding the encounter	0%
Diagnosis 1	Nominal	The primary diagnosis (coded as first three digits of ICD9); 848 distinct values	0%
Diagnosis 2	Nominal	Secondary diagnosis (coded as first three digits of ICD9); 923 distinct values	0%
Diagnosis 3	Nominal	Additional secondary diagnosis (coded as first three digits of ICD9); 954 distinct values	1%
Number of diagnoses	Numeric	Number of diagnoses entered to the system	0%

Glucose serum test result	Nominal	Indicates the range of the result or if the test was not taken. Values: “>200,” “>300,” “normal,” and “none” if not measured	0%
A1c test result	Nominal	Indicates the range of the result or if the test was not taken. Values: “>8” if the result was greater than 8%, “>7” if the result was greater than 7% but less than 8%, “normal” if the result was less than 7%, and “none” if not measured.	0%
Change of medications	Nominal	Indicates if there was a change in diabetic medications (either dosage or generic name). Values: “change” and “no change”	0%
Diabetes medications	Nominal	Indicates if there was any diabetic medication prescribed. Values: “yes” and “no”	0%
24 features for medications	Nominal	For the generic names: metformin, repaglinide, nateglinide, chlorpropamide, glimepiride, acetohexamide, glipizide, glyburide, tolbutamide, pioglitazone, rosiglitazone, acarbose, miglitol, troglitazone, tolazamide, examide, sitagliptin, insulin, glyburide-metformin, glipizide-metformin, glimepiride-pioglitazone, metformin-rosiglitazone, and metformin-pioglitazone, the feature indicates whether the drug was prescribed or there was a change in the dosage. Values: “up” if the dosage was increased during the encounter, “down” if the dosage was decreased, “steady” if the dosage did not change, and “no” if the drug was not prescribed	0%
Readmitted	Nominal	Days to inpatient readmission. Values: “<30” if the patient was readmitted in less than 30 days, “>30” if the patient was readmitted in more than 30 days, and “No” for no record of readmission.	0%

5. Travail à effectuer

Voici un récapitulatif du travail effectuer dans ce projet

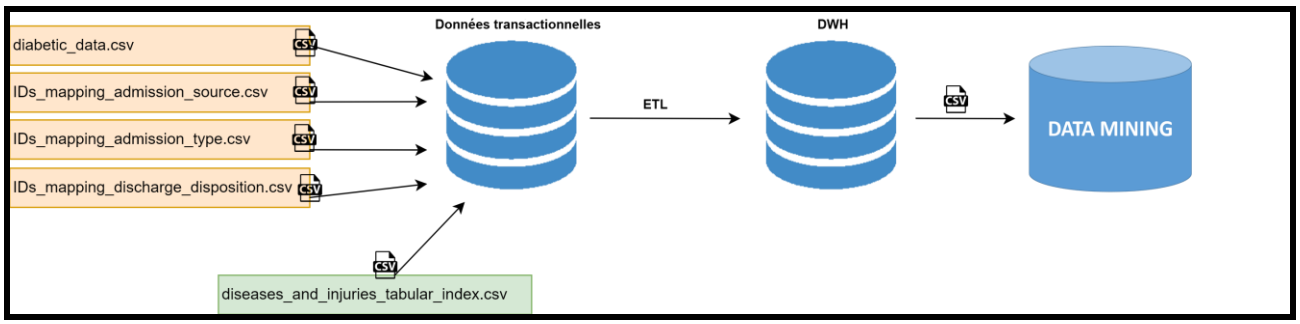


FIGURE 1 : RECAPITULATIF DU TRAVAIL

II. MODELISATION

L'ensemble de données a été téléchargé à partir de la bibliothèque d'apprentissage machine UCI et les fichiers csv ont été extraits. Une base de données a été créée sur une instance Oracle locale pour stocker les données dans des tables et créer des sauvegardes lors du nettoyage des données. Les fichier IDs_mapping_XX.csv ont été chargé et transformé en format relationnel. Les fichiers de mappage et de détail ont été importés dans la base de données précédemment crée à l'aide des tables externes.

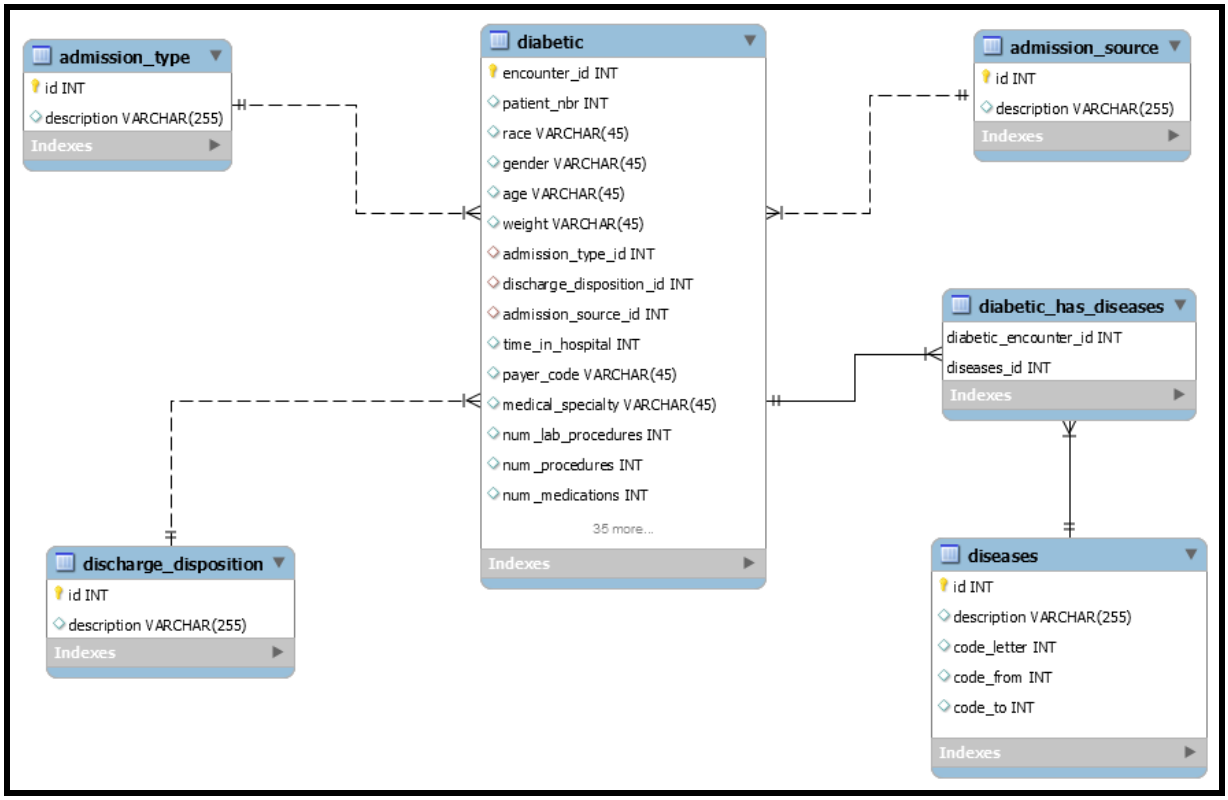


FIGURE 2 : MODELE E/A NORMALISE DES DONNEES

1. Étude des besoins

Exemple de requêtes brutes :

Quels sont les patients diabétiques qui ont des visites l'année précédente ?

```
patient
        / week , weekday
        / Month , Quarter
        / year
```

Quels sont les patients renvoyés dans un autre établissement médical ou renvoyé à domicile avec des services de santé ?

```
patient
        / readmitted
        / payer_code , discharge_disposition
```

Quels sont les patients qui ont effectués un test A1C en fonction du diagnostic secondaire.

```
patient
        / a1c_test_results
        / diag2
```

Requêtes organisées par dimension :

```
patient
        / Time(week , weekday)
        / Time(Month , Quarter)
        / Time(year)
```

```
patient
        / discharge(readmitted)
        / discharge(payer_code, discharge_disposition)
```

```
patient
        / Test(a1c_test_results)
        / diagnosis(diag2)
```

2. Étude des données

Le modèle relationnel sous-jacent aux données présentes :

```
discharge(id ,discharge_disposition,readmitted,payer_code)
patient(id,num, race , gender , age)
admission(id, admission_type , admission_source , medical_speciality)
Test(id, glucose_serum_test_result , a1c_test_results)
diagnosis (id, diag1, diag2 , diag3)
```

3. Modèle dimensionnel

Pour chaque contexte d'usage on identifie :

- La table des faits
- Les dimensions en intégrant les vues exprimées

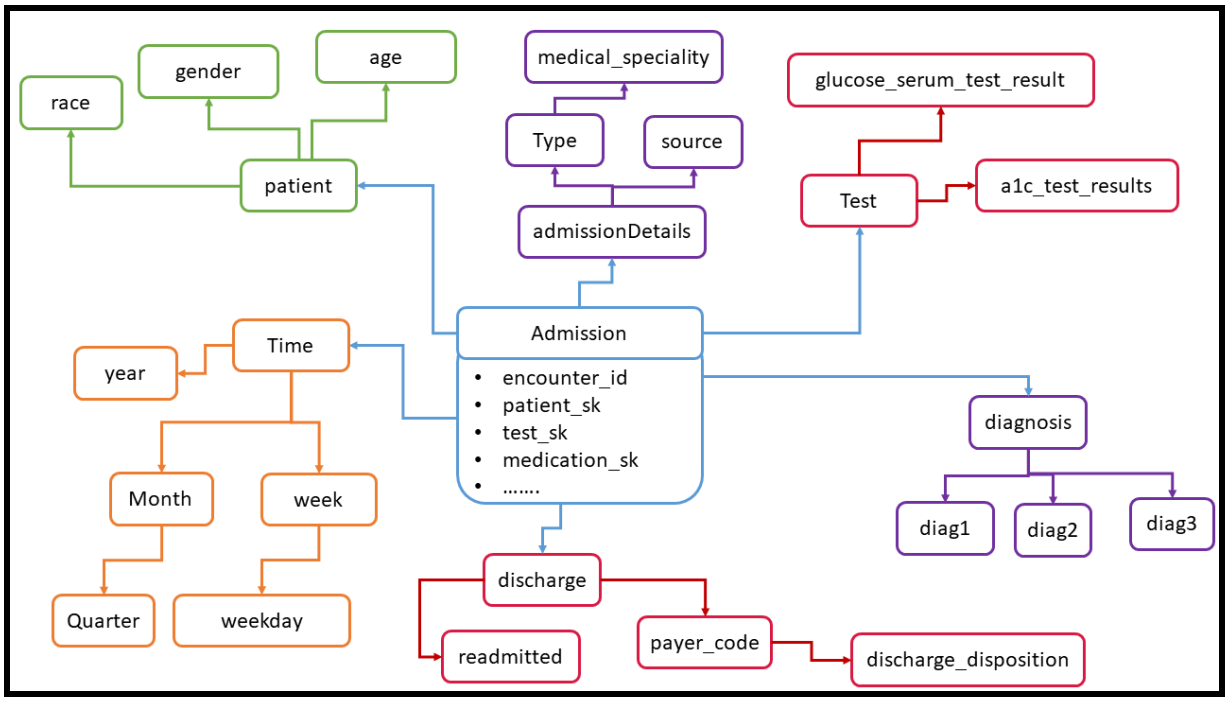


FIGURE 3 : VUE INTEGREE DU MODELE DIMENSIONNEL

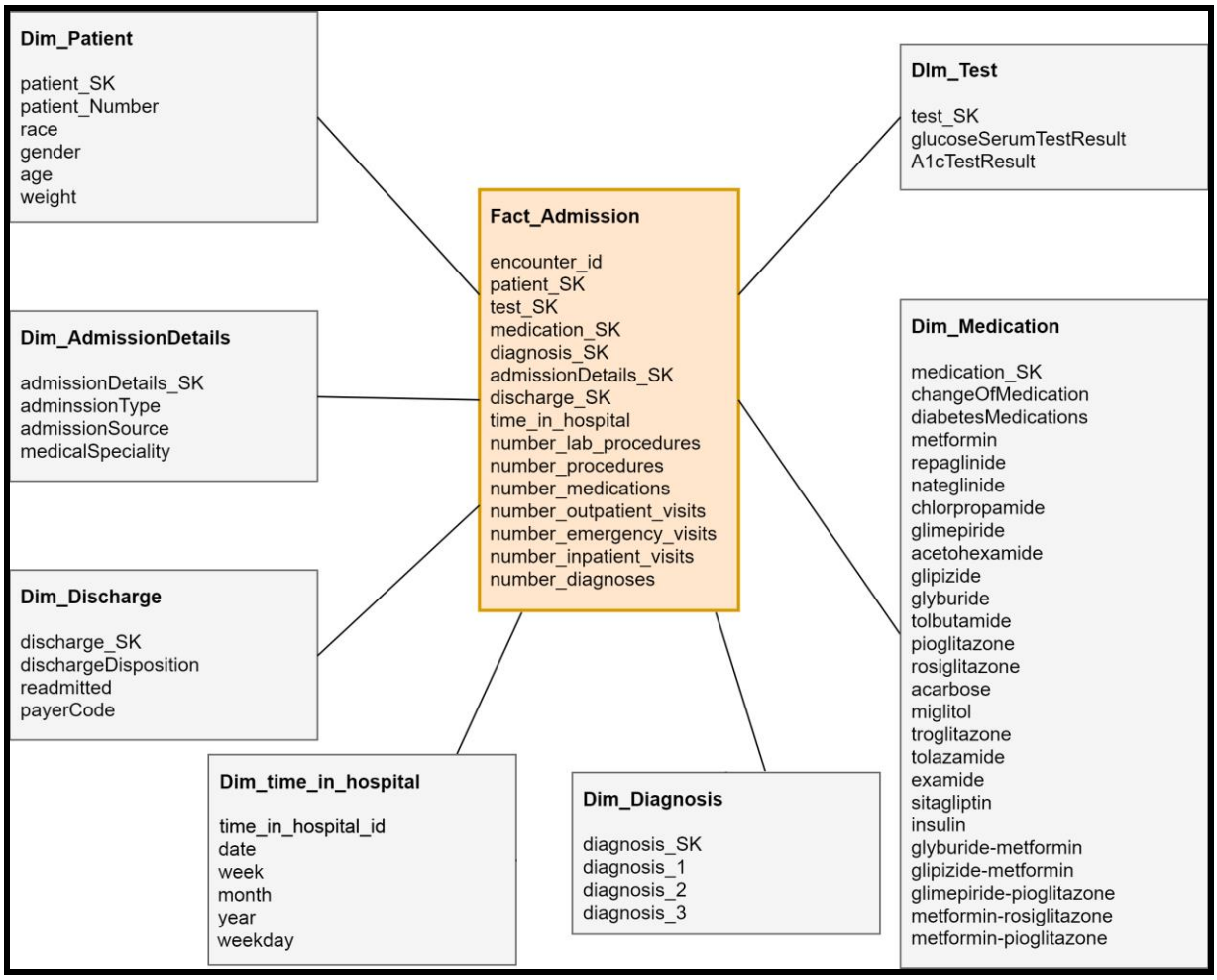


FIGURE 4 : MODELE DIMENSIONNEL DU DATA WAREHOUSE

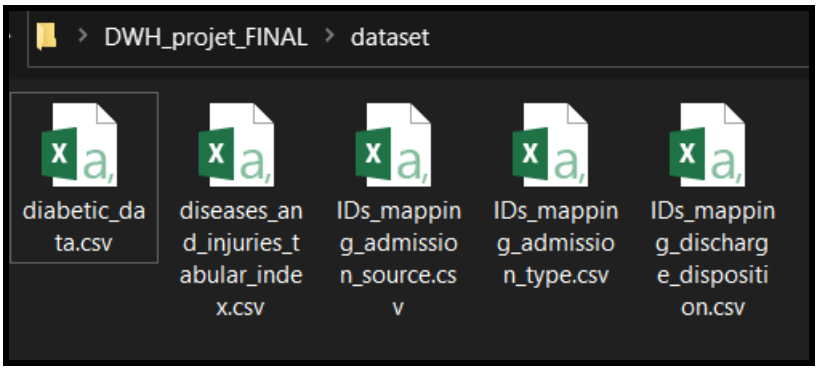
III. ETL

Nous allons Créer un ETL permettant de remplir notre DataWarehouse

1. Extraction

La première étape de l'ETL est l'extraction. Au cours de l'extraction, les données sont spécifiquement identifiées et ensuite prélevées à de nombreux endroits différents. Ces données peuvent provenir d'une variété de choses, comme des fichiers, des feuilles de calcul, des systèmes de bases de données et des applications, etc. Il n'est généralement pas possible d'identifier le sous-ensemble exact d'intérêt, de sorte que l'on extrait plus de données que nécessaire pour s'assurer qu'il couvre tous les besoins.

L'objectif est d'abord de créer la Base de Donnée Externe, Donc on crée des tables externes pour chacun des fichiers :



Voici un exemple de création de table externe à partir du de ficher CSV 'diabetic_data.csv' :

```
BEGIN
  EXECUTE IMMEDIATE 'DROP TABLE zak_DWH.dataset cascade constraints';
EXCEPTION
  WHEN OTHERS THEN
    IF sqlcode != -0942 THEN RAISE; END IF;
END;
/
CREATE OR REPLACE DIRECTORY diabeticDataSrc
AS 'C:/Users/Zakaria/Desktop/DWH_projet_FINAL/dataset';
CREATE OR REPLACE DIRECTORY diabeticDataLog
AS 'C:/Users/Zakaria/Desktop/DWH_projet_FINAL/Log';
/
CREATE TABLE zak_DWH.dataset (
  encounter_id NUMBER(10) NULL ,
  patient_nbr NUMBER(10) NULL ,
  race VARCHAR2(45) NULL ,
  gender VARCHAR2(45) NULL ,
  age VARCHAR2(45) NULL ,
  weight VARCHAR2(45) NULL ,
  admission_type_id NUMBER(10) NULL ,
  discharge_disposition_id NUMBER(10) NULL ,
  admission_source_id NUMBER(10) NULL ,
  time_in_hospital NUMBER(10) NULL ,
  payer_code VARCHAR2(45) NULL ,
  medical_specialty VARCHAR2(45) NULL ,
  num_lab_procedures NUMBER(10) NULL ,
  num_procedures NUMBER(10) NULL ,
```

```
  num_medications NUMBER(10) NULL ,
  number_outpatient NUMBER(10) NULL ,
  number_emergency NUMBER(10) NULL ,
  number_inpatient NUMBER(10) NULL ,
  diag_1 VARCHAR2(200) NULL ,
  diag_2 VARCHAR2(200) NULL ,
  diag_3 VARCHAR2(200) NULL ,
  number_diagnoses NUMBER(10) NULL ,
  max_glu_serum VARCHAR2(45) NULL ,
  A1Cresult VARCHAR2(45) NULL ,
  metformin VARCHAR2(45) NULL ,
  repaglinide VARCHAR2(45) NULL ,
  nateglinide VARCHAR2(45) NULL ,
  chlorpropamide VARCHAR2(45) NULL ,
  glimepiride VARCHAR2(45) NULL ,
  acetohexamide VARCHAR2(45) NULL ,
  glipizide VARCHAR2(45) NULL ,
  glyburide VARCHAR2(45) NULL ,
  tolbutamide VARCHAR2(45) NULL ,
  pioglitazone VARCHAR2(45) NULL ,
  rosiglitazone VARCHAR2(45) NULL ,
  acarbose VARCHAR2(45) NULL ,
  miglitol VARCHAR2(45) NULL ,
  troglitazone VARCHAR2(45) NULL ,
  tolazamide VARCHAR2(45) NULL ,
  examide VARCHAR2(45) NULL ,
  citoglipton VARCHAR2(45) NULL ,
  insulin VARCHAR2(45) NULL ,
  glyburide_metformin VARCHAR2(45) NULL ,
  glipizide_metformin VARCHAR2(45) NULL ,
  glimepiride_pioglitazone VARCHAR2(45) NULL ,
  metformin_rosiglitazone VARCHAR2(45) NULL ,
  metformin_pioglitazone VARCHAR2(45) NULL ,
  change VARCHAR2(45) NULL ,
  diabetesMed VARCHAR2(45) NULL ,
  readmitted VARCHAR2(45) NULL
)
ORGANIZATION EXTERNAL
(TYPE ORACLE_LOADER
DEFAULT DIRECTORY diabeticDataSrc
ACCESS PARAMETERS
(
RECORDS DELIMITED BY '\r\n'
SKIP 1
CHARACTERSET UTF8
BADFILE diabeticDataLog:'diabeticDataBad.bad'
LOGFILE diabeticDataLog:'diabeticDataLog.log'
FIELDS TERMINATED BY ','
OPTIONALLY ENCLOSED BY '"'
)
LOCATION ('diabetic_data.csv'))
REJECT LIMIT UNLIMITED;

-- Vérifiez si tous les 101766 enregistrements sont importés en exécutant
la requête ci-dessous.
SELECT COUNT(*) FROM dataset;
```

```
Procédure PL/SQL terminée.

Directory DIABETICDATASRC créé(e).

Directory DIABETICDATALOG créé(e).

Table ZAK_DWH.DATASET créé(e).

  COUNT(*)
----------
    101766
```

2. Transformation

La prochaine étape du processus ETL est la transformation. Une fois les données extraites, elles doivent être physiquement transportées vers la destination cible et converties dans le format approprié. Cette transformation de données peut inclure des opérations telles que le nettoyage, l'assemblage et la validation des données.

- **Filtrage horizontal :**

Certains attributs importants qui devraient être pris en compte sont absents de l'ensemble de données. Il faut les supprimer :

```sql
DELETE FROM zak_DWH.dataset_modified
WHERE payer_code = '?';

DELETE FROM zak_DWH.dataset_modified
WHERE medical_specialty = '?';

DELETE FROM zak_DWH.dataset_modified
WHERE race = '?';

DELETE FROM zak_DWH.dataset_modified
WHERE diag_1 = '?';

DELETE FROM zak_DWH.dataset_modified
WHERE diag_2 = '?';

DELETE FROM zak_DWH.dataset_modified
WHERE diag_3 = '?';

SELECT COUNT(*) FROM zak_DWH.dataset_modified;
```

```
775 lignes supprimé.


5 lignes supprimé.


107 lignes supprimé.


273 lignes supprimé.


  COUNT(*)
----------
     26755
```

Sélectionner des données sales en fonction du sexe du patient et les nettoyés en sélectionnant les données les plus fréquentes et les plus récentes:

```sql
SELECT DISTINCT gender
FROM zak_DWH.dataset_modified;

CREATE OR REPLACE VIEW zak_DWH.dirty_patient_gender AS
SELECT *
FROM zak_DWH.dataset_modified
WHERE patient_nbr in (
        SELECT patient_nbr
        FROM zak_DWH.dataset_modified
        WHERE gender = 'Female'
) AND patient_nbr in (
        SELECT patient_nbr
        FROM zak_DWH.dataset_modified
        WHERE gender = 'Male'
);

SELECT encounter_id, patient_nbr, race, gender
FROM zak_DWH.dirty_patient_gender;

UPDATE zak_DWH.dataset_modified
SET gender = 'Male'
WHERE patient_nbr = 109210482;
```

```
GENDER
------------------------------------------
Male
Female


View ZAK_DWH.DIRTY_PATIENT_GENDER créé(e).


ENCOUNTER_ID PATIENT_NBR RACE                                            GENDER
------------ ----------- ------------------------------------------------ ----------
   186533256   109210482 Caucasian                                        Male
   183845022   109210482 Caucasian                                        Female


2 lignes mis à jour.
```

Sélectionner des données sales en fonction de la race du patient et les nettoyés en sélectionnant les données les plus fréquentes et les plus récentes:

```sql
SELECT distinct race
FROM zak_DWH.dataset_modified;
--6 races distinctes sont trouvées. (Caucasian, AfricanAmerican, ?, Other,
Asian, Hispanic)

-- Vues pour identifier les données sales
CREATE OR REPLACE VIEW zak_DWH.dirty_patient_race AS
SELECT patient_nbr, count(distinct race) as race_count
FROM zak_DWH.dataset_modified
group by patient_nbr
having count(distinct race) > 1;

SELECT count(patient_nbr)
FROM zak_DWH.dirty_patient_race;


SELECT encounter_id, patient_nbr, race
FROM zak_DWH.dataset_modified
WHERE patient_nbr in (
        SELECT patient_nbr
    FROM zak_DWH.dirty_patient_race
)
ORDER BY patient_nbr, encounter_id;

-- Définir race comme Caucasian
UPDATE zak_DWH.dataset_modified
SET race = 'Caucasian'
WHERE patient_nbr IN (1553220, 23724792, 38893887, 42246738, 52316388,
112367349);

-- Définir race comme AfricanAmerican
UPDATE zak_DWH.dataset_modified
SET race = 'AfricanAmerican'
WHERE patient_nbr IN (6919587, 10980891, 40090752, 54643194, 101753730,
107849052);

-- Définir race comme Other
UPDATE zak_DWH.dataset_modified
SET race = 'Other'
WHERE patient_nbr IN (28532295, 30689766, 32314608, 33247647, 36967347,
37547937, 37638306, 38774187, 39160719, 42096384, 90817893, 93105117,
93662784, 94027644, 98584524, 100322946, 103228398, 103690161, 105125598,
106425234);

-- Définir race comme Asian
UPDATE zak_DWH.dataset_modified
SET race = 'Asian'
WHERE patient_nbr IN (24332220, 31812075, 34248078, 94539465, 97024806,
103305528, 104622570, 110657970, 111534210);

-- Définir race comme Hispanic
UPDATE zak_DWH.dataset_modified
SET race = 'Hispanic'
WHERE patient_nbr IN (37572957, 44744166, 45113778, 90035874, 91107549,
93809358, 94088088, 98934615, 106895331, 109448541);
```

```
RACE
-----------------------------------------
Asian
Other
Caucasian
AfricanAmerican
Hispanic


View ZAK_DWH.DIRTY_PATIENT_RACE créé(e).


COUNT(PATIENT_NBR)
------------------
                50
```

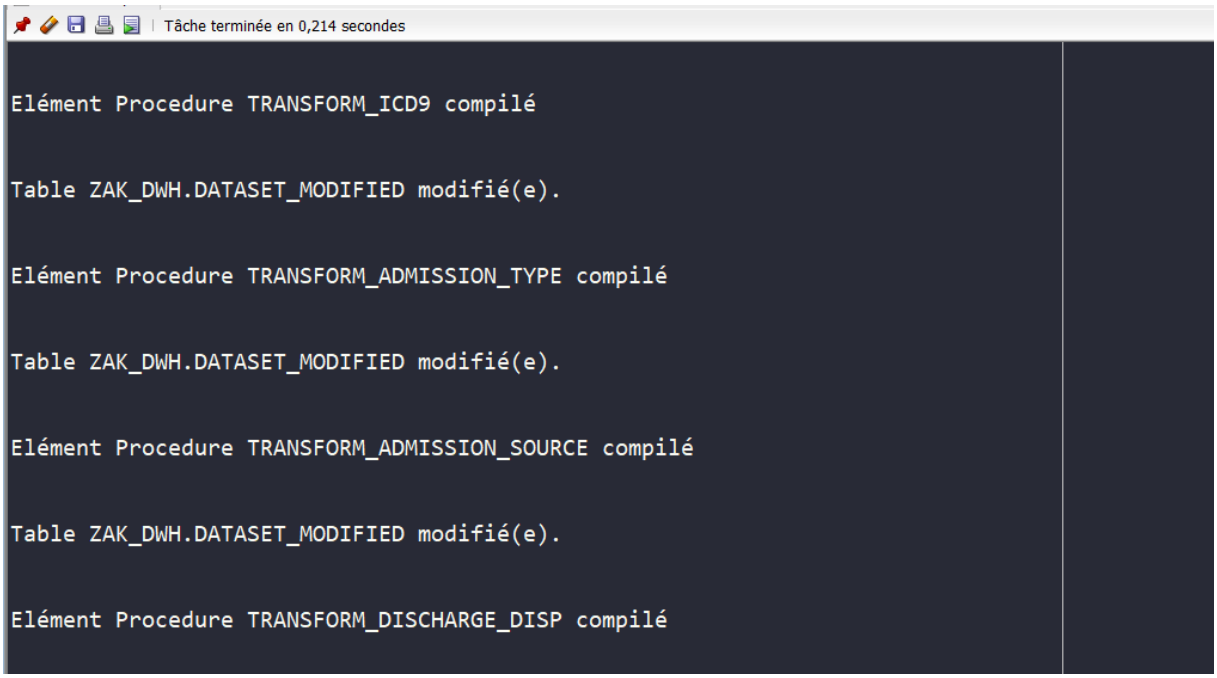
```
ENCOUNTER_ID PATIENT_NBR RACE
------------ ----------- ------------------------------------------
    94182252     1553220 AfricanAmerican
   108685740     1553220 Caucasian
    90247512     6919587 Caucasian
    95240682     6919587 AfricanAmerican
   103499478    10980891 Caucasian
   111776052    10980891 AfricanAmerican
   124794924    23724792 Other
   169202754    23724792 Caucasian
   107885574    24332220 Caucasian
   181908144    24332220 Asian
   115370466    28532295 Hispanic

ENCOUNTER_ID PATIENT_NBR RACE
------------ ----------- ------------------------------------------
   143876196    28532295 Hispanic
   162090672    28532295 Hispanic
```

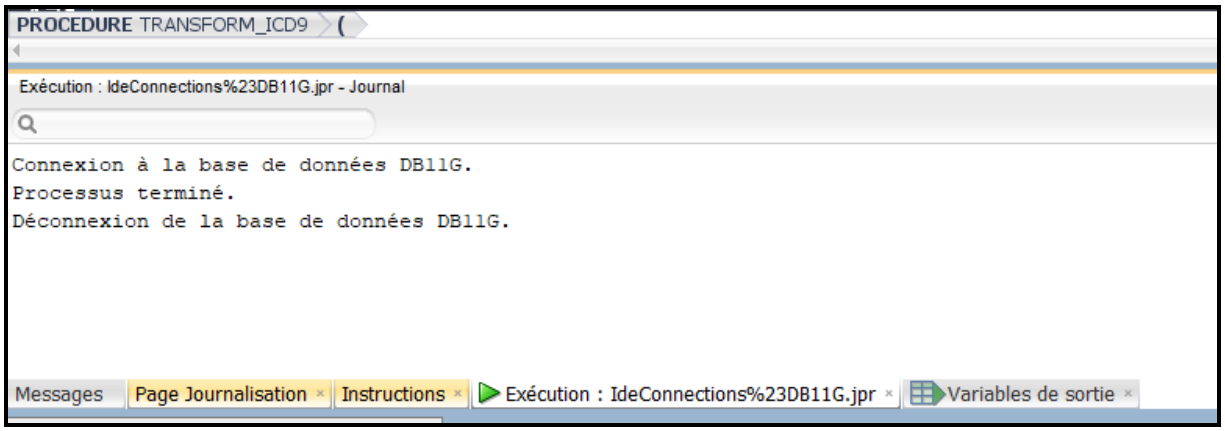
Nous allons maintenant transformer le diagnostic primaire, secondaire et supplémentaire basé sur la "International Statistical Classification of Diseases and Related Health Problems"

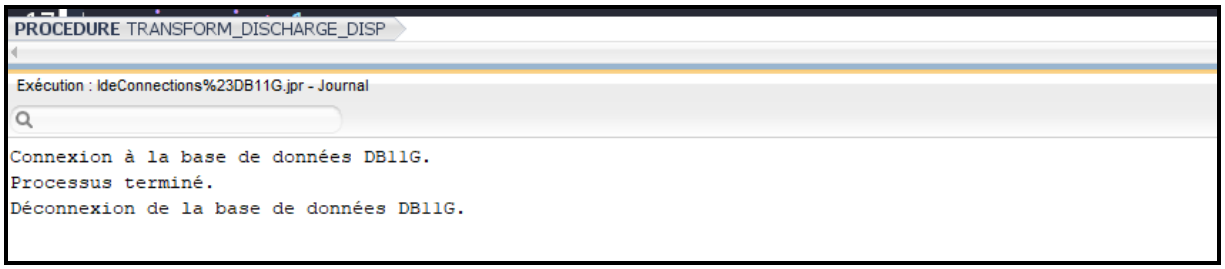
NB : Ce siteWeb contient l'index tabulaire des maladies et des blessures.

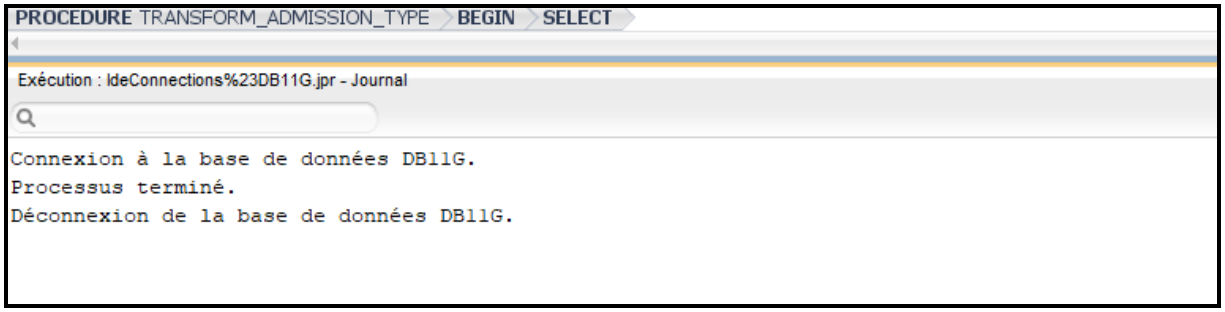
Après qu'on a chargé le fichier csv 'diseases_and_injuries_tabular_index.csv' dans une table externe, on peut maintenant lancer les procédures suivantes afin de Transformer les détails de l'ICD9 ainsi que ADMISSION_TYPE, ADMISSION_SOURCE et DISCHARGE_DISPOSITION :

```
BEGIN
    zak_DWH.TRANSFORM_ICD9();
    zak_DWH.TRANSFORM_ADMISSION_TYPE();
    zak_DWH.TRANSFORM_ADMISSION_SOURCE();
    zak_DWH.TRANSFORM_DISCHARGE_DISP();
--rollback;
END;
```


Tâche terminée en 0,214 secondes
Elément Procedure TRANSFORM_ICD9 compilé
Table ZAK_DWH.DATASET_MODIFIED modifié(e).
Elément Procedure TRANSFORM_ADMISSION_TYPE compilé
Table ZAK_DWH.DATASET_MODIFIED modifié(e).
Elément Procedure TRANSFORM_ADMISSION_SOURCE compilé
Table ZAK_DWH.DATASET_MODIFIED modifié(e).
Elément Procedure TRANSFORM_DISCHARGE_DISP compilé


PROCEDURE TRANSFORM_ICD9
Exécution : IdeConnections%23DB11G.jpr - Journal
Connexion à la base de données DB11G.
Processus terminé.
Déconnexion de la base de données DB11G.
Messages
Page Journalisation
Instructions
Exécution : IdeConnections%23DB11G.jpr
Variables de sortie


PROCEDURE TRANSFORM_DISCHARGE_DISP
Exécution : IdeConnections%23DB11G.jpr - Journal
Connexion à la base de données DB11G.
Processus terminé.
Déconnexion de la base de données DB11G.


PROCEDURE TRANSFORM_ADMISSION_TYPE
BEGIN
SELECT
Exécution : IdeConnections%23DB11G.jpr - Journal
Connexion à la base de données DB11G.
Processus terminé.
Déconnexion de la base de données DB11G.

3. Chargement des données (Loading)

La dernière étape du processus ETL consiste à charger les données transformées dans les tables de dimensions et des faits. Cette cible peut être une base de données ou un DataWarehouse.

Voici un exemple de création de la table de dimension dim_patient :

```
CREATE TABLE dim_patient (
  patient_sk NUMBER(10) ,
  patient_number VARCHAR(45),
  race VARCHAR(45)  ,
  gender VARCHAR(45)  ,
  age VARCHAR(45)

);

CREATE UNIQUE INDEX patient_sk_UNIQU ON dim_patient (patient_sk);

ALTER TABLE dim_patient ADD (
  CONSTRAINT patient_sk_pk PRIMARY KEY (patient_sk)
  CONSTRAINT gender_type CHECK (gender in ('male','female'))
  );

-- SEQUENCE
CREATE SEQUENCE patient_seq START WITH 1;

-- TRIGGER
CREATE OR REPLACE TRIGGER patient_bir
BEFORE INSERT ON dim_patient
FOR EACH ROW

BEGIN
  SELECT patient_seq.NEXTVAL
  INTO   :new.patient_sk
  FROM   dual;
END;
/
```

```
Table DIM_PATIENT créé(e).


INDEX PATIENT_SK_UNIQU créé(e).


Table DIM_PATIENT modifié(e).


Sequence PATIENT_SEQ créé(e).


Elément Trigger PATIENT_BIR compilé
```

Remplissage de la table de dimensions dim_patient :

```
INSERT INTO dim_patient (patient_number, race, gender, age)
SELECT DISTINCT patient_nbr, race, gender, age
FROM zak_DWH.dataset_modified
```

```sql
ORDER BY patient_nbr, age;

SELECT COUNT(*) FROM dim_patient;

SELECT *
FROM dim_patient
WHERE patient_sk = 62;
```

```
19 808 lignes inséré.


  COUNT(*)
----------
     19808


PATIENT_SK PATIENT_NUMBER                           RACE                                               GENDER
---------- ---------------------------------------- -------------------------------------------------- ------------
        62 179973                                   Caucasian                                          Male
```

Nous pouvons traiter la variable age pour qu'elle soit au format entier et non au format d'une chaîne de caractères :

```sql
ALTER TABLE zak_DWH.dataset_modified
ADD age_int NUMBER(10);


CREATE OR REPLACE PROCEDURE zak_DWH.transform_for_datamining
IS

 i NUMBER(10) DEFAULT 0;
 age_str VARCHAR2(10);
 age_str_int NUMBER(10);
BEGIN

WHILE i < 10 LOOP
        age_str := '[' || i * 10 || '-' || (i+1) * 10 || ')';
    age_str_int := i * 10 + 5;

    UPDATE zak_DWH.dataset_modified
    SET age_int = age_str_int
    WHERE age = age_str;

    i := i+1;
END LOOP;

END;
/

--BEGIN
--zak_DWH.transform_for_datamining();
--rollback;
--END;


SELECT DISTINCT age, age_int
FROM zak_DWH.dataset_modified;
```

Sortie de script × | Résultat de requête ×

SQL | Toutes les lignes extraites : 10 en 0,023 secondes

	AGE	AGE_INT
1	[90-100)	95
2	[50-60)	55
3	[30-40)	35
4	[10-20)	15
5	[70-80)	75
6	[60-70)	65
7	[20-30)	25
8	[0-10)	5
9	[80-90)	85
10	[40-50)	45

Après la création des tables du modèle en étoile, on peut maintenant générer un fichier csv pour effectuer les taches de Data Mining :

```
SET HEADING OFF
SET FEEDBACK OFF
SET ECHO OFF
SET PAGESIZE 0
SPOOL C:/Users/Zakaria/Desktop/DWH_projet_FINAL/out.csv
SELECT '"' || race || '";"' || gender || '";"' || age_int || '";"' ||
admission_type || '";"' ||
        discharge_disposition || '";"' || admission_source || '";"' ||
time_in_hospital || '";"' || payer_code || '";"' ||
    medical_specialty || '";"' || num_lab_procedures || '";"' ||
num_procedures || '";"' || num_medications || '";"' ||
    number_outpatient || '";"' || number_emergency || '";"' ||
number_inpatient || '";"' || diag_1 || '";"' || diag_2 || '";"' ||
    diag_3 || '";"' || number_diagnoses || '";"' || max_glu_serum || '";"'
|| A1Cresult || '";"' || metformin || '";"' || repaglinide || '";"' ||
    nateglinide || '";"' || chlorpropamide || '";"' || glimepiride || '";"'
|| acetohexamide || '";"' || glipizide || '";"' ||
    glyburide || '";"' || tolbutamide || '";"' || pioglitazone || '";"' ||
rosiglitazone || '";"' || acarbose || '";"' || miglitol || '";"' ||
    troglitazone || '";"' || tolazamide || '";"' || examide || '";"' ||
citoglipton || '";"' || insulin || '";"' || glyburide_metformin || '";"' ||
    glipizide_metformin || '";"' || glimepiride_pioglitazone || '";"' ||
metformin_rosiglitazone || '";"' ||
    metformin_pioglitazone || '";"' || change || '";"' || diabetesMed ||
'";"' || readmitted || '"'
FROM dataset_modified;
SPOOL OFF
```

Après, on peut exécuter cette commande pour générer le ficher CSV :

```
SQL> @"C:/Users/Zakaria/Desktop/DWH_projet_FINAL/22_Export_CSV_file_for_Data_Mining.sql";
```

IV. EXPLORATION ET DATAMINING

Maintenant, on va Explorer nos données afin de répondre aux questions que nous avons posées au début de ce rapport.

L'analyse exploratoire des données est le processus de visualisation et d'analyse des données pour en extraire des informations. En d'autres termes, c'est le processus de synthèse des caractéristiques importantes des données afin de mieux comprendre l'ensemble de données.

1. Analyse univariée

L'analyse de données univariées (le type de données ne comprend qu'une seule variable) est donc la forme d'analyse la plus simple puisque l'information ne concerne qu'une

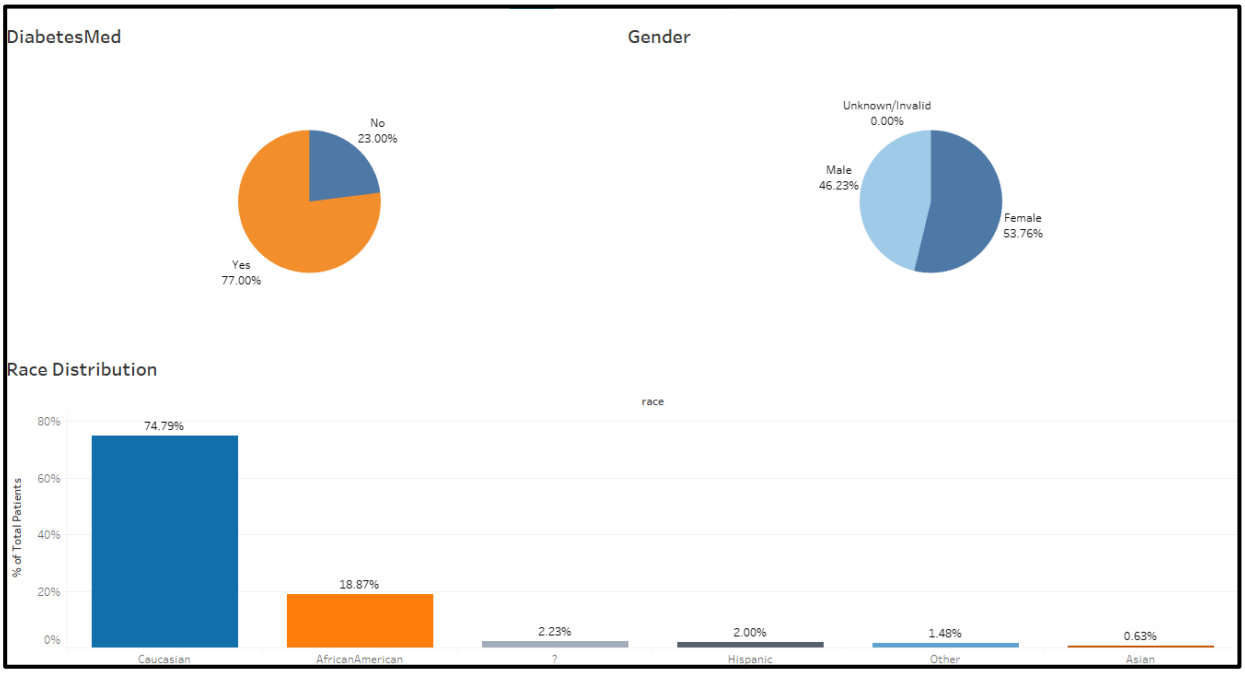


FIGURE 5 : DIAGRAMMES UNIVARIES: RACE, SEXE, DIABETE

seule grandeur qui change. Il ne traite pas des causes ou des relations et le but principal de l'analyse est de décrire les données et de trouver les modèles qui y existent.

- **DiabetesMed**: Les médicaments contre le diabète sont administrés à la majorité (77%) des patients admis à l'hôpital.
- **Gender**: il y a une répartition presque égale des hommes et des femmes admis à l'hôpital, environ 54% de femmes et 46% d'hommes. Pas une grande différence en ce qui concerne le sexe.

- **Race**: Il y a 5 catégories différentes dans la fonction Race, le maximum (plus de 70%) d'entre elles appartenant à la race caucasienne.

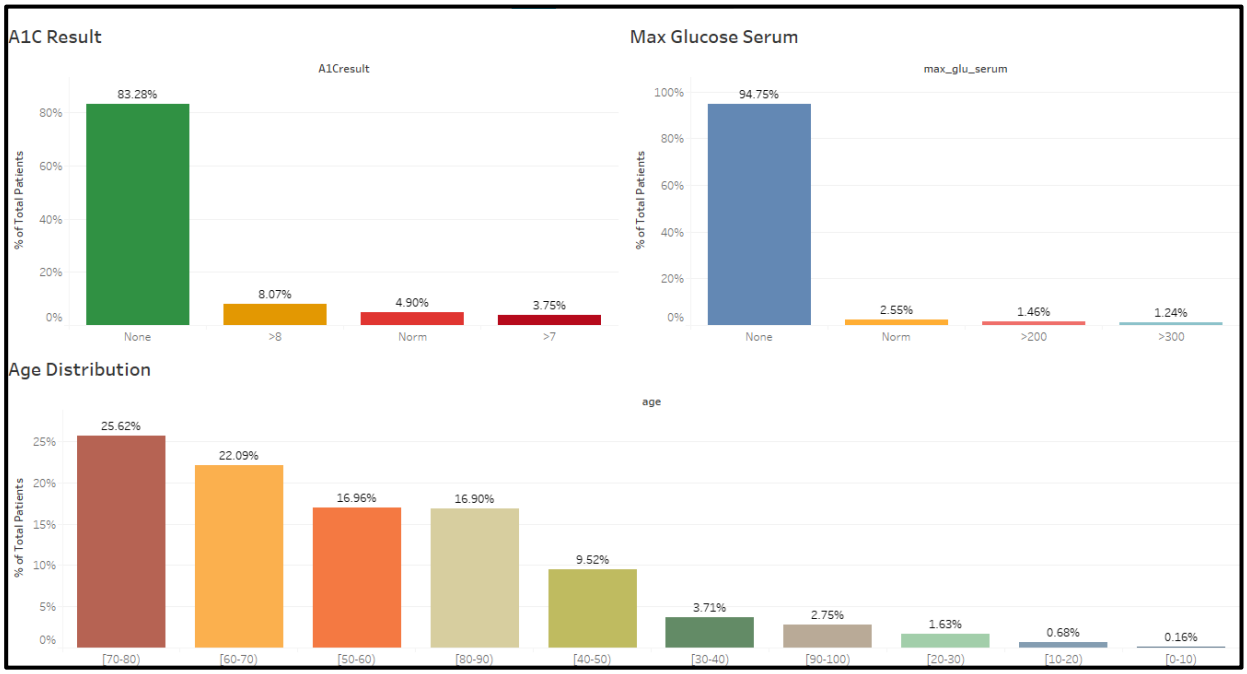


FIGURE 6 : GRAPHIQUES UNIVARIES: AGE, MAX_GLU_SERUM, A1CRESULT

- **A1Cresult**: Il y a 4 catégories dans A1Cresult qui indiquent le pourcentage maximum de patients qui n'ont pas été testés pour le test A1C.
- **Max_glu_serum**: Il y a 4 catégories dans max_glu_serum qui indiquaient le pourcentage maximum de patients n'ayant pas été testés pour le max_glu_serum.
- **Age**: le nombre maximum de patients admis se situe entre 50 et 80 ans.

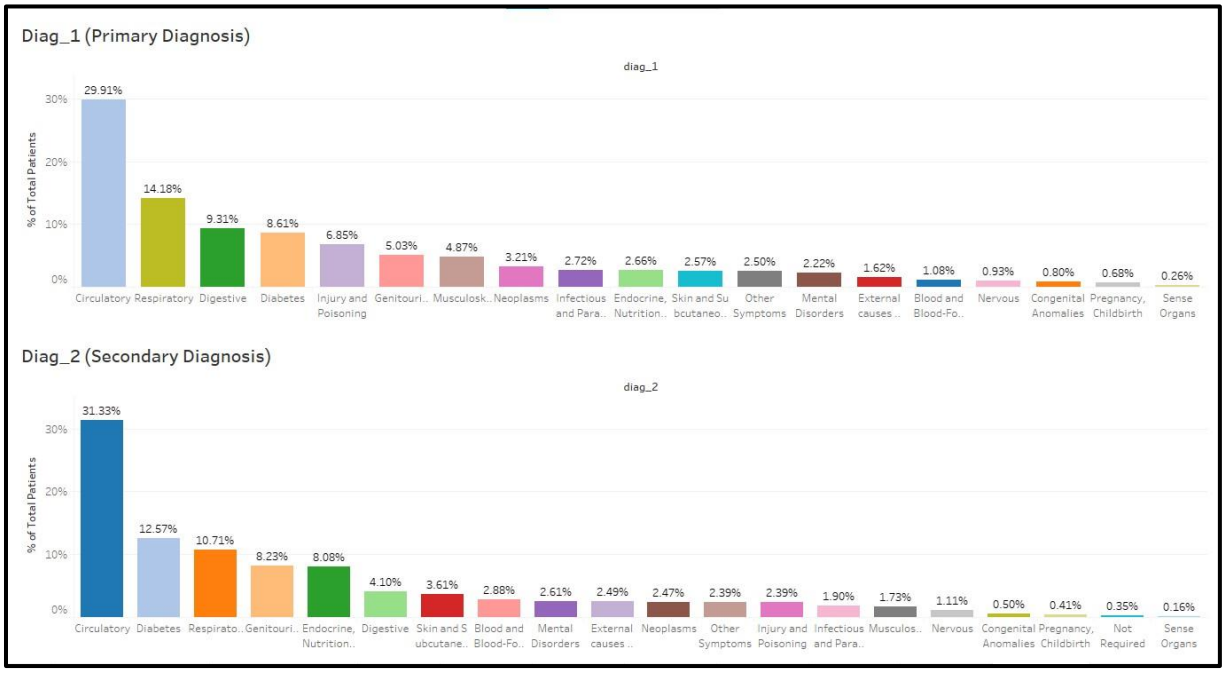


FIGURE 7 : DIAGNOSTIC PRIMAIRE ET SECONDAIRE

- Les maladies circulatoires sont celles qui sont le plus souvent diagnostiquées à la fois comme diagnostic primaire (~ 30%) et secondaire (~ 31%), indiquant que les États-Unis ont beaucoup de patients souffrant de problèmes de maladie circulatoire.
- 14% des patients ont été principalement diagnostiqués avec des maladies respiratoires suivies de maladies digestives (~ 9%).
- Environ 9% des patients ont été diagnostiqués principalement comme diabétiques et environ 13% des patients ont reçu un diagnostic de diabète lors d'un diagnostic secondaire.

2. Analyse bivariée

L'analyse des données bivariées (le type de données implique deux variables différentes) traite des causes et des relations et l'analyse est effectuée pour découvrir la relation entre les deux variables.

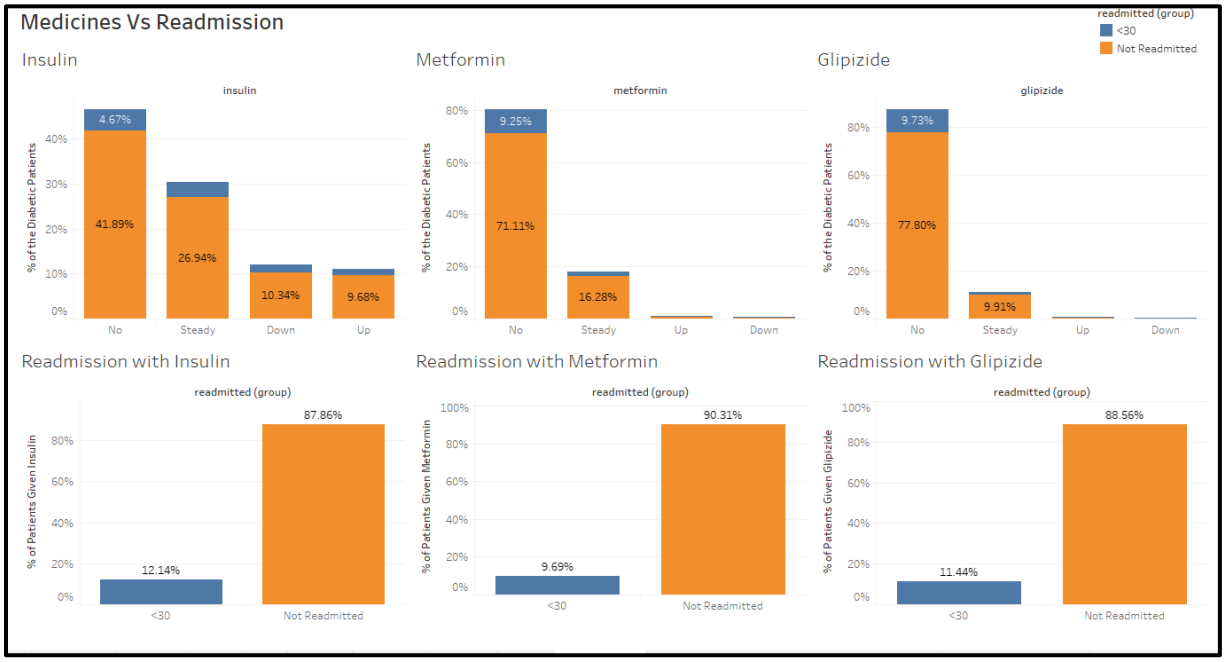


FIGURE 8 : MEDICAMENTS IMPORTANTS CONTRE READMISSION

- Au-dessus, 3 médicaments sont les plus significatifs selon le test du chi 2 d'indépendance par rapport à la variable dépendante «readmitted» que nous avons tracée avec la variable readmitted pour voir la relation entre eux. En outre, parmi les médicaments, il y a ceux qui sont administrés au maximum de patients par rapport aux autres médicaments.
- L'insuline est le médicament le plus important administré à près de 55% des patients admis à l'hôpital. Environ 88% des patients ayant reçu de l'insuline n'ont pas été réadmis.
- La metformine semble être le 2e médicament le plus important après l'insuline. Il est donné à env. 20% des patients. Environ 90% des patients qui ont reçu de la metformine n'ont pas été réadmis.

- Le glipizide est un autre médicament important administré à environ 13% des patients. Environ 89% des patients ayant reçu du glipizide n'ont pas été réadmis.

3. Construction de modèles

a) Divisez les données (Train/Test)

La fonction train_test_split sert à diviser un seul ensemble de données en deux parties différentes: Apprentissages et Tests. Le sous-ensemble d'Apprentissage sert à créer notre modèle et le sous-ensemble de test sert à utiliser le modèle sur des données inconnues pour évaluer les performances du modèle.

On va diviser les données en 70% en tant que données de train, 30% en tant que données de test et état aléatoire = 0.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 0, stratify = y)
```

b) Appliquer les modèles de machine Learning

Notre choix de modèles est principalement régi par notre objectif de comprendre les facteurs les plus importants, ainsi que leurs effets relatifs sur le changement de médicament et la réadmission. Par conséquent, nous n'avons pas implémenté de modèles qui ont peu ou pas d'interprétabilité (réseaux de neurones, machines vectorielles de support, voisins les plus proches, etc.). Les modèles que nous avons mis en œuvre sont les suivants :

- **Logistic Regression**: La régression logistique est le pendant de la classification de la régression linéaire. Les prédictions sont mappées entre 0 et 1 via la fonction logistique, ce qui signifie que les prédictions peuvent être interprétées comme des probabilités de classe. Les modèles eux-mêmes sont toujours «linéaires», ils fonctionnent donc bien lorsque vos classes sont linéairement séparables (c'est-à-dire qu'ils peuvent être séparés par une seule surface de décision.

- **Decision Trees :** Un arbre de décision est une représentation simple des classifications et des régressions. Il s'agit d'un Machine Learning supervisé où les données sont continuellement divisées en fonction d'un certain paramètre. Il décompose un ensemble de données en sous-ensembles de plus en plus petits tout en développant en même temps un arbre de décision associé. Le résultat final est un arbre avec des nœuds de décision et des nœuds feuilles. Un nœud de décision a deux branches ou plus. Le nœud feuille représente une classification ou une décision. Le nœud de décision le plus élevé dans un arbre qui correspond au meilleur prédicteur appelé nœud racine. Dans l'arbre de décision, comme nous n'avons pas de modèle probabiliste, mais juste une division binaire, nous n'avons pas besoin de faire d'hypothèse du tout.

- **Random Forests**: Random Forest est un algorithme d'apprentissage supervisé. La «forêt» qu'elle construit est un ensemble d'arbres de décision, généralement formés avec la méthode du «bagging». En considérant plus d'un arbre de décision puis en effectuant un vote à la majorité, les forêts aléatoires ont contribué à être des représentations prédictives plus robustes que les arbres comme dans le cas

précédent. Il n'a pas de modèle en dessous et la seule hypothèse sur laquelle il repose est que l'échantillonnage est représentatif. Mais c'est généralement une hypothèse courante.

- **Support Vector Machines** : SVM construit un hyperplan ou un ensemble d'hyperplans dans un espace de dimension élevée ou infinie, qui peut être utilisé pour la classification, la régression ou d'autres tâches telles que la détection des valeurs aberrantes. Intuitivement, une bonne séparation est obtenue par l'hyperplan qui a la plus grande distance au point de données d'apprentissage le plus proche de toute classe (soi-disant marge fonctionnelle), car en général plus la marge est grande, plus l'erreur de généralisation du classificateur est faible.

NB : La mise en œuvre de ces modèles ainsi que leurs évaluations correspondantes sont décrites dans les fichiers « .ipynb » associés à ce projet.

Résultats trouvées :

Si le patient présente les caractéristiques suivantes, il a une forte probabilité d'être réadmis:

• Nombre élevé de visites l'année précédente.

• Si le patient est renvoyé dans un autre établissement médical ou renvoyé à domicile avec des services de santé.

• Nombre élevé de diagnostics.

• Si le patient reçoit des médicaments contre le diabète.

• Si la principale maladie diagnostiquée était du système circulatoire.

• Si la metformine et / ou l'insuline ne sont pas administrées ou si la posologie est faible.

• Si le diagnostic secondaire venait à être le diabète.

• Si le test A1C n'a pas été effectué.

V. CONCLUSION

Les réadmissions sont des admissions aiguës non planifiées à l'hôpital dans un délai défini à partir d'une admission initiale. Les taux de réadmission sont une mesure de la qualité de la santé bien établie à l'échelle internationale, car certaines réadmissions qui se produisent sont évitables et si les données sont correctement modélisées, les groupes de patients à haut risque de réadmission sont identifiables. La réadmission à l'hôpital est un facteur important des dépenses médicales totales et est un indicateur émergent de la qualité des soins. Il est perturbateur pour les patients et coûteux pour les systèmes de santé. L'objectif de ce projet était de développer un modèle de risque prédictif à partir d'un DataWarehouse pour identifier les patients diabétiques présentant un risque élevé de réadmission à l'hôpital.

Ce projet a été conçu pour aider les hôpitaux à réduire les taux de réadmission des patients diabétiques. Avec le modèle proposé, les hôpitaux peuvent cibler les patients dans les percentiles à haut risque. Non seulement ces patients présentent un risque plus élevé d'être réadmis, mais la précision du modèle est considérablement meilleure pour les percentiles supérieurs, ce qui signifie que les hôpitaux peuvent utiliser efficacement leurs ressources pour réduire les taux de réadmission en administrant des médicaments très efficaces dans le traitement du diabète comme l'insuline, la metformine et le glipizide. et glyburide. Les établissements médicaux peuvent prendre des mesures de précaution avec ces patients lors de leur admission initiale en rendant obligatoire le test A1C et le test de glucose sérique maximal et en fournissant le traitement en conséquence. Des visites de suivi pour vérifier leurs progrès devraient également être planifiées au moment du congé.

Cela permet aux hôpitaux de fournir une meilleure qualité de soins à leurs patients et de réduire les taux de réadmission. Cette réduction peut aider les hôpitaux à éviter les pénalités encourues pour des taux de réadmission élevés, entraînant une réduction des dépenses de santé pour des centaines, voire des milliers de dollars par patient diabétique, tout en améliorant simultanément l'état de santé et en sauvant des vies.

Pour résumer, voici les recommandations qui peuvent être proposées :

• Les établissements médicaux peuvent prendre des mesures de précaution avec les patients lors de leur admission initiale en rendant obligatoire l'A1C et le test de glucose sérique maximal et en fournissant le traitement en conséquence, car 80 à 90% des patients réadmis n'ont pas subi ces tests.

• Un suivi avec les patients sortis devrait être un suivi pour garder une trace de leur santé et pour les conseiller de temps en temps.

• Le régime médicamenteux actuel des patients à haut risque doit être réévalué et les médicaments les plus efficaces doivent être envisagés.

• Les médicaments les plus efficaces selon les résultats sont l'insuline, la metformine et le glipizide. Ces médicaments se révèlent statistiquement significatifs, très importants par rapport aux différents modèles d'apprentissage automatique utilisés, sont les plus largement prescrits et sont associés à un faible risque de réadmission s'ils sont administrés au patient.

• Les plans annuels, les données financières et l'infrastructure / l'inventaire de l'hôpital doivent être planifiés en conséquence en tenant compte des réadmissions prévues.

• Les hôpitaux doivent accorder une attention et des soins supplémentaires aux patients à haut risque.

Vivre avec le diabète est difficile et pénible. L'état du patient diabétique ne peut pas être compris uniquement à partir de ses dossiers médicaux. Il est nécessaire de collecter et d'analyser des informations à la fois subjectives et objectives sur les patients afin de bien comprendre la survenue de réadmission de patients diabétiques. Les données subjectives peuvent être saisies en interrogeant les patients ou en menant des enquêtes qui enrichiront la profondeur des informations sur les patients. La conversation entre le médecin et le patient peut également être recueillie et analysée, ce qui pourrait aider à extraire des caractéristiques importantes correspondant à la volonté et à l'attitude du patient grâce à des techniques d'exploration de texte. Ces informations pourraient aider à construire des DataWarehouse performant et améliorer les modèles intelligents pour identifier les patients à haut risque de réadmission.

À la fin de ce rapport, nous tenons à remercier notre cher professeur Mr. **Guénaël Cabanes** pour la passion qu'il nous a communiqué pour ce travail et nous espérons être à la hauteur de ses attentes.