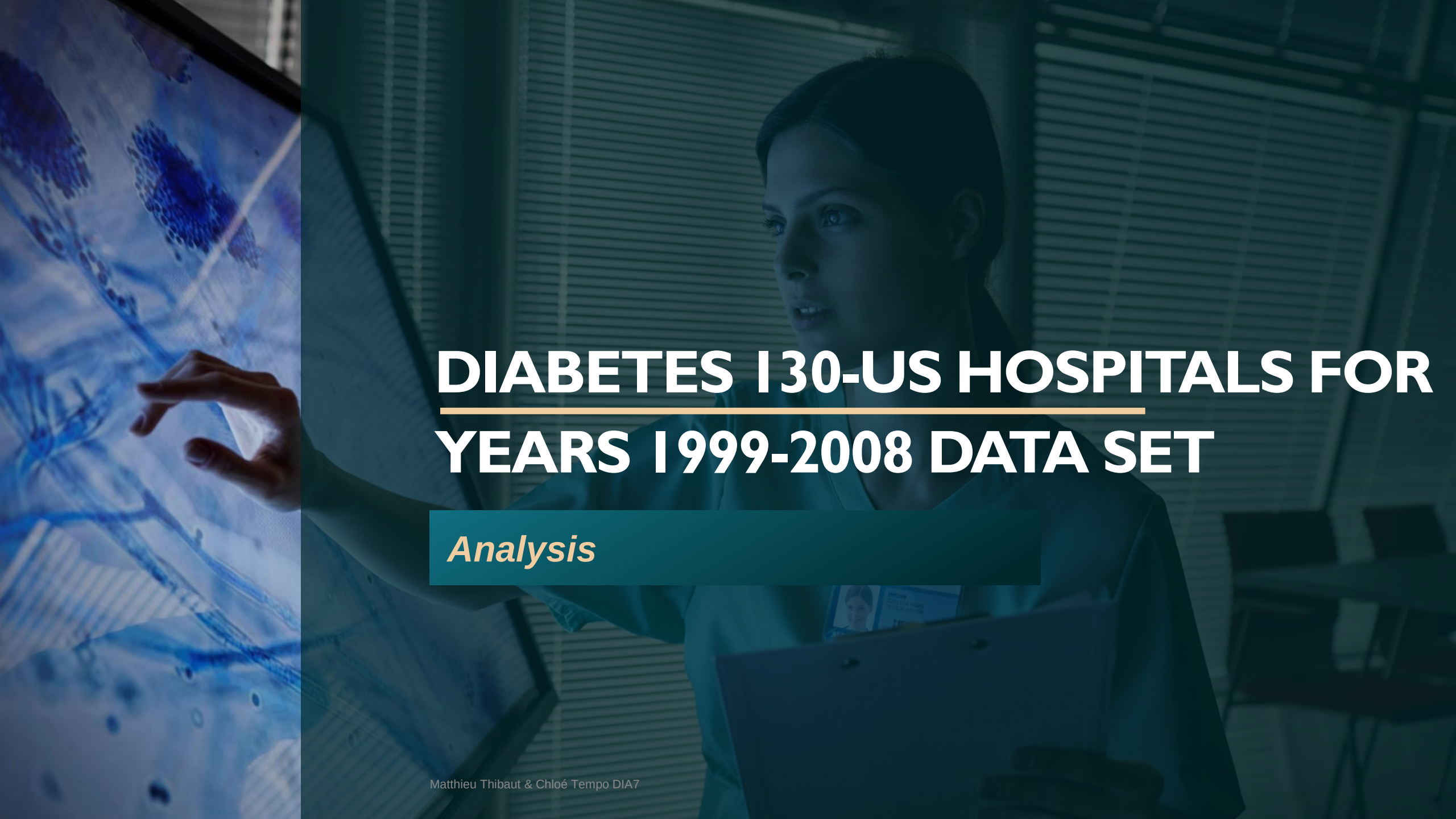


DIABETES 130-US HOSPITALS FOR YEARS 1999-2008 DATA SET

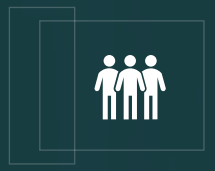
Analysis

Matthieu Thibaut & Chloé Tempo DIA7

DESCRIPTION OF THE DATASET

10 years (1999-2008) of clinical care at 140 US hospitals and integrated delivery networks.

It includes over 50 features representing patient and hospital outcomes. Information was extracted from the database for encounters that satisfied the following criteria :



INPATIENT

It is an inpatient encounter (a hospital admission).

DIABETIC

It is a diabetic encounter, that is, one during which any kind of diabetes was diagnosed.

1-14 DAYS

The length of the stay was at least 1 day and at most 14 days.

LABORATORY

Laboratory tests were performed during the encounter.

MEDICINE

Medication were administred during the encounter.

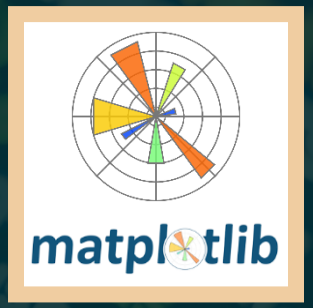
STEPS

1. LOAD AND CLEAN THE DATASET
2. ANALYSE , VISUALIAZE AND PREDICT
3. DISCUSS RESULTS

LIBRARIES

STRUCTURE

VISUALIZATION



MACHINE LEARNING

GOAL OF OUR ANALYSIS

- Attention focus on the 'readmitted' feature
--> 3 possibles types : 'No', '<30' and '>30'

- Goal : Study the variables and correlations and try to predict the 'readmitted' feature

FIRST VIEW OF THE DATA

0	encounter_id	101766 non-null	int64
1	patient_nbr	101766 non-null	int64
2	race	99493 non-null	object
3	gender	101766 non-null	object
4	age	101766 non-null	object
5	weight	3197 non-null	object
6	admission_type_id	101766 non-null	int64
7	discharge_disposition_id	101766 non-null	int64
8	admission_source_id	101766 non-null	int64
9	time_in_hospital	101766 non-null	int64
10	payer_code	61510 non-null	object
11	medical_specialty	51817 non-null	object
12	num_lab_procedures	101766 non-null	int64
13	num_procedures	101766 non-null	int64
14	num_medications	101766 non-null	int64
15	number_outpatient	101766 non-null	int64
16	number_emergency	101766 non-null	int64
17	number_inpatient	101766 non-null	int64
18	diag_1	101745 non-null	object
19	diag_2	101408 non-null	object
20	diag_3	100343 non-null	object
21	number_diagnoses	101766 non-null	int64
22	max_glu_serum	101766 non-null	object
23	A1Cresult	101766 non-null	object

24	metformin	101766 non-null	object
25	repaglinide	101766 non-null	object
26	nateglinide	101766 non-null	object
27	chlorpropamide	101766 non-null	object
28	glimepiride	101766 non-null	object
29	acetohexamide	101766 non-null	object
30	glipizide	101766 non-null	object
31	glyburide	101766 non-null	object
32	tolbutamide	101766 non-null	object
33	pioglitazone	101766 non-null	object
34	rosiglitazone	101766 non-null	object
35	acarbose	101766 non-null	object
36	miglitol	101766 non-null	object
37	troglitazone	101766 non-null	object
38	tolazamide	101766 non-null	object
39	examide	101766 non-null	object
40	citoglipton	101766 non-null	object
41	insulin	101766 non-null	object
42	glyburide-metformin	101766 non-null	object
43	glipizide-metformin	101766 non-null	object
44	glimepiride-pioglitazone	101766 non-null	object
45	metformin-rosiglitazone	101766 non-null	object
46	metformin-pioglitazone	101766 non-null	object
47	change	101766 non-null	object

48	diabetesMed	101766 non-null	object
49	readmitted	101766 non-null	object

dtypes: int64(13), object(37)
memory usage: 38.8+ MB

DATA REMOVING

Number of null values

	0
diag_1	0.020636
diag_2	0.351787
diag_3	1.398306
race	2.233555
payer_code	39.557416
medical_specialty	49.082208
weight	96.858479

Dropping columns with too much missing values

With the precedent results, we eliminate the columns that have too many missing values

```
diabetes_df.drop(columns = ['weight', 'medical_specialty', 'payer_code', 'encounter_id'], inplac
```

Some columns provide a constant value for every row, it is useless so we can eliminate them too.

```
diabetes_df.drop(columns = ['examide', 'citoglipton', 'glimepiride-pioglitazone', 'metformin-ros
```

Dropping redundant rows

When we analysed some data in this dataset, we observed that there was some redundant rows, some rows had the same number of patient value, and sometimes the linked variables were not coherent

We also drop these rows

```
diabetes_df.drop_duplicates(subset = "patient_nbr", keep = 'first', inplace = True)
```

MAPPING AND NEW COLUMNS

Group name	icd9 codes
Circulatory	390–459, 785
Respiratory	460–519, 786
Digestive	520–579, 787
Diabetes	250.xx
Injury	800–999
Musculoskeletal	710–739
Genitourinary	580–629, 788
Neoplasms	140–239
	780, 781, 784, 790–799
	240–279, without 250

```
map_admission_type_id = {1: 'Emergency',
                         2:'Urgent',
                         3:'Elective',
                         4:'Newborn',
                         5:'Not Available',
                         6:'NULL',
                         7:'Trauma Center',
                         8 :'Not Mapped'}

diabetes_df.admission_type_id = diabetes_df.admission_type_id.map(map_admission_type_id)
```

```
map_admission_source_id = {1:"Physician Referral",
                           2:"Clinic Referral",
                           3:"HMO Referral",
                           4:"Transfer from a hospital",
                           5:"Transfer from a Skilled Nursing Facility (SNF)",
                           6:"Transfer from another health care facility",
                           7:"Emergency Room",
                           8:"Court/Law Enforcement",
                           9:"Not Available",
                           10:"Transfer from critial access hospital",
                           11:"Normal Delivery",
                           12:"Premature Delivery",
                           13:"Sick Baby",
                           14:"Extramural Birth",
                           15:"Not Available",
                           17:"NULL",
                           18:"Transfer From Another Home Health Agency",
                           19:"Readmission to Same Home Health Agency",
                           20:"Not Mapped",
                           21:"Unknown/Invalid",
                           22:"Transfer from hospital inpt/same fac reslt in a sep claim",
                           23:"Born inside this hospital",
                           24:"Born outside this hospital",
                           25:"Transfer from Ambulatory Surgery Center",
                           26:"Transfer from Hospice"}

diabetes_df.admission_source_id = diabetes_df.admission_source_id.map(map_admission_source_id)
```

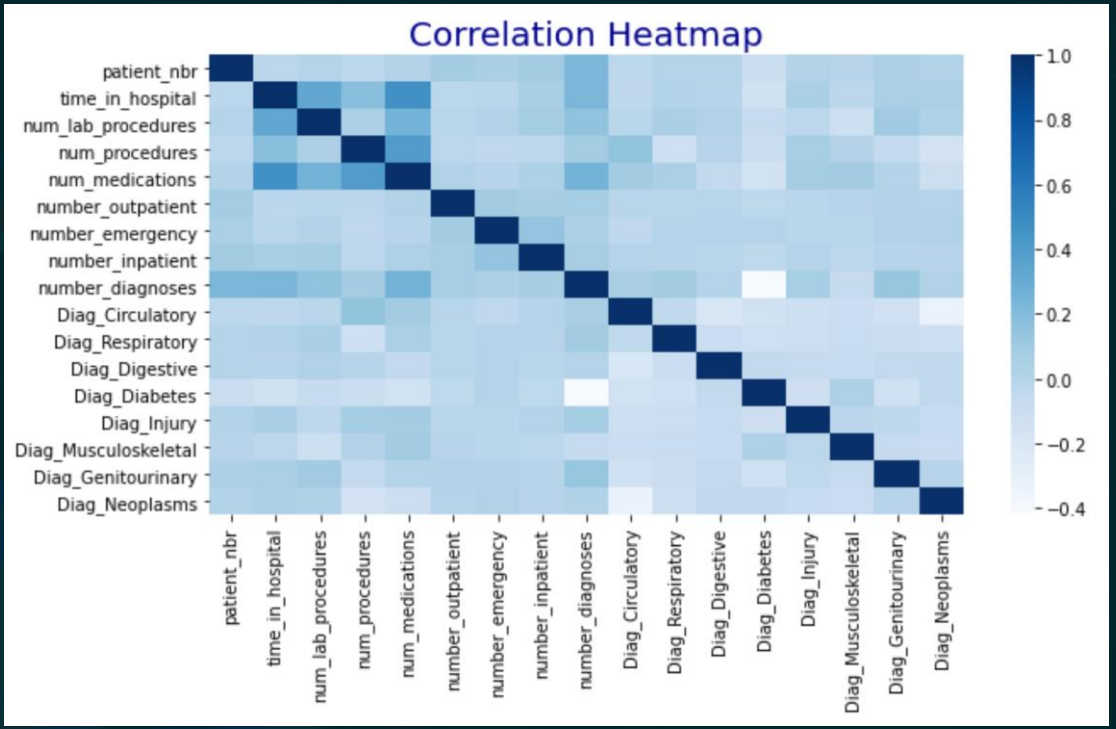
```
map_discharge_disposition_id = {1:"Discharged to home",
                                2:"Discharged/transferred to another short term hospital",
                                3:"Discharged/transferred to SNF",
                                4:"Discharged/transferred to ICF",
                                5:"Discharged/transferred to another type of inpatient care inst
                                6:"Discharged/transferred to home with home health service",
                                7:"Left AMA",
                                8:"Discharged/transferred to home under care of Home IV provider
                                9:"Admitted as an inpatient to this hospital",
                                10:"Neonate discharged to another hospital for neonatal aftercar
                                11:"Expired",
                                12:"Still patient or expected to return for outpatient services"
                                13:"Hospice / home",
                                14:"Hospice / medical facility",
                                15:"Discharged/transferred within this institution to Medicare a
                                16:"Discharged/transferred/referred another institution for outp
                                17:"Discharged/transferred/referred to this institution for outp
                                18:"NULL",
                                19:"Expired at home. Medicaid only, hospice.",
                                20:"Expired in a medical facility. Medicaid only, hospice.",
                                21:"Expired, place unknown. Medicaid only, hospice.",
                                22:"Discharged/transferred to another rehab fac including rehab
                                23:"Discharged/transferred to a long term care hospital.",
                                24:"Discharged/transferred to a nursing facility certified under
                                25:"Not Mapped",
                                26:"Unknown/Invalid",
                                30:"Discharged/transferred to another Type of Health Care Instit
                                27:"Discharged/transferred to a federal health care facility.",
                                28:"Discharged/transferred/referred to a psychiatric hospital of
                                29:"Discharged/transferred to a Critical Access Hospital (CAH)."

diabetes_df.discharge_disposition_id = diabetes_df.discharge_disposition_id.map(map_discharge_di
```

VISUALIZATION (I)

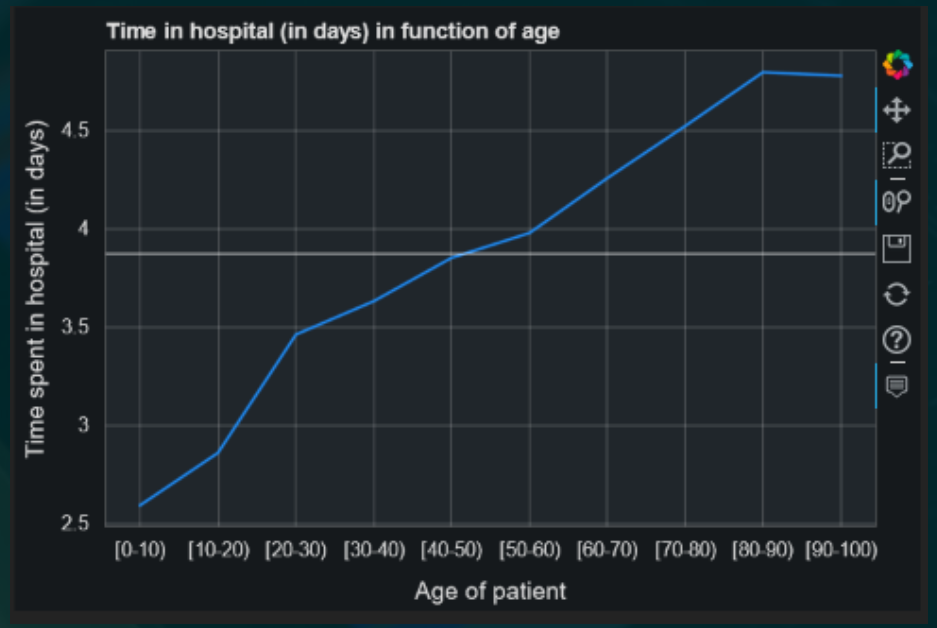
Secondly, visualization makes it possible to make the variations in the data visual and to establish certain opinions on behaviour between variables

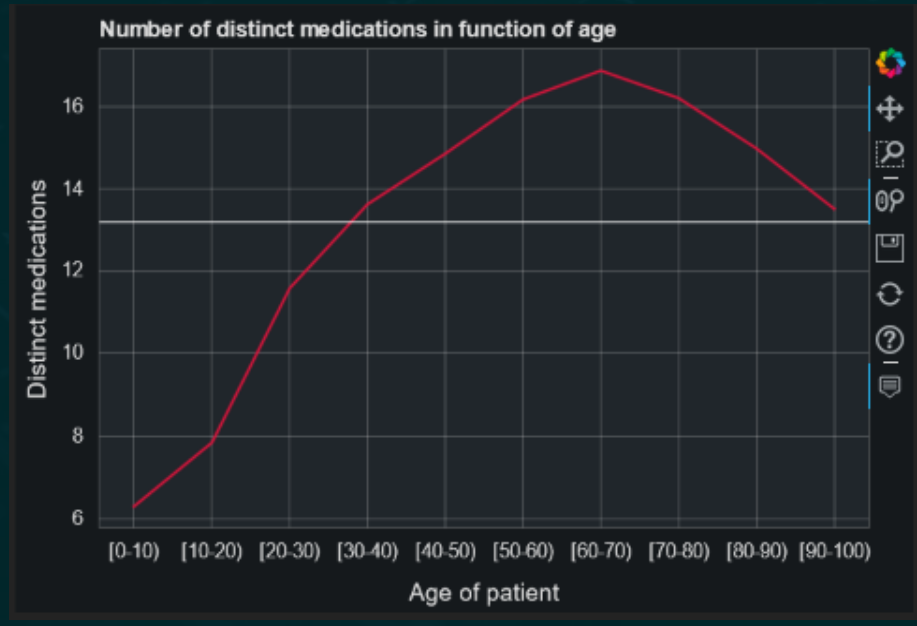
1. Correlation



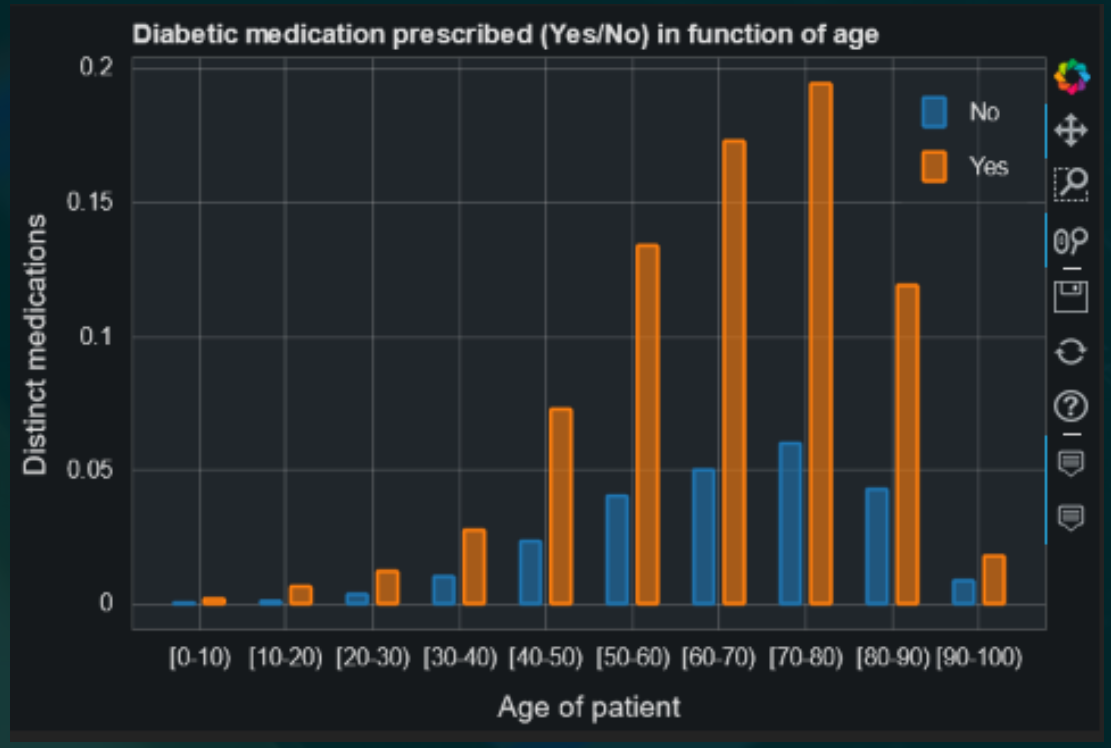
VISUALIZATION (II)

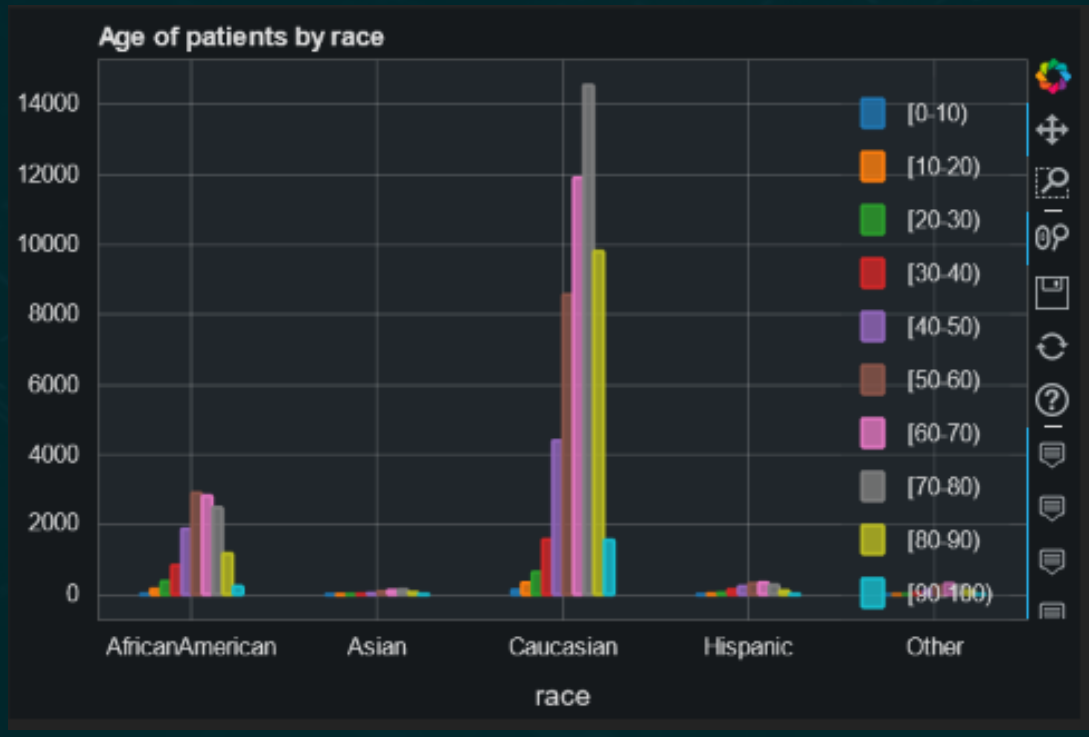
2. Observation of the variables with each others, some significant examples





VISUALIZATION (II)





MACHINE LEARNING

After preparing a machine learning oriented dataset from the original dataset, we decided to distinguish two cases according to the data:

Case 1: Predict patient’s readmission under 30 days
Case 2: Predict patient’s readmission under and above 30 days

MACHINE LEARNING

We obtained the following results :

- **Case 1: Predict patient's readmission under 30 days**

Model	Score	Accuracy
K-Nearest Neighbors	0.9067	0.9067
Logistic Regression	0.9128	0.9128
Linear SVC	0.9130	0.9130
Random Forest	0.9109	0.9121
Adaptive boosting	0.9128	0.9128
Decision Tree	0.8342	0.8327
Extra Trees	0.9112	0.9117
Naive Bayes	0.8783	0.8783

- **Case 2: Predict patient's readmission under or above 30 days**

Model	Score	Accuracy
K-Nearest Neighbors	0.5805	0.5805
Logistic Regression	0.6238	0.6238
Linear SVC	0.6231	0.6231
Random Forest	0.6093	0.6065
Adaptive boosting	0.6323	0.6323
Decision Tree	0.5600	0.5575
Extra Trees	0.6032	0.6032
Naive Bayes	0.6077	0.6077

MACHINE TUNING

Models can be configured with different settings. We can then conduct research to see which parameters best adapt to our dataset to obtain the best prediction results.

We obtained the following results :

- **Case 1: Predict patient's readmission under 30 days**

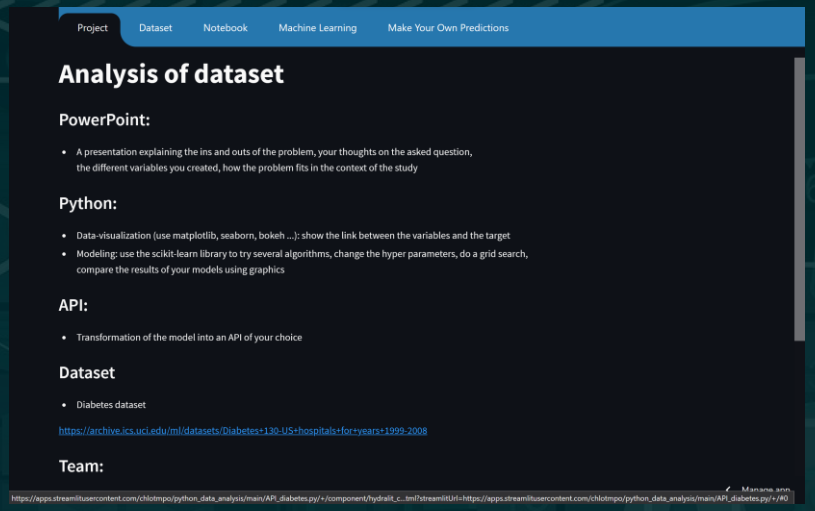
Model	Score	Accuracy
Linear SVC	0.9130	0.9067
Logistic Regression	0.9128	0.9128
Random Forest	0.9117	0.9130
Adaptive Boosting	0.9116	0.9121
Extra Trees	0.9115	0.9128
K-Nearest Neighbors	0.9067	0.8327
Naive Bayes	0.8865	0.9117
Decision Tree	0.8342	0.8783

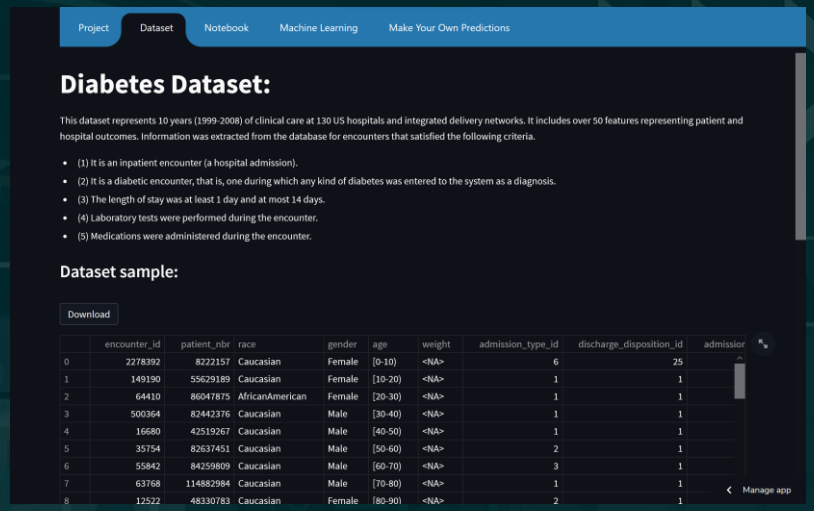
- **Case 2: Predict patient's readmission under or above 30 days**

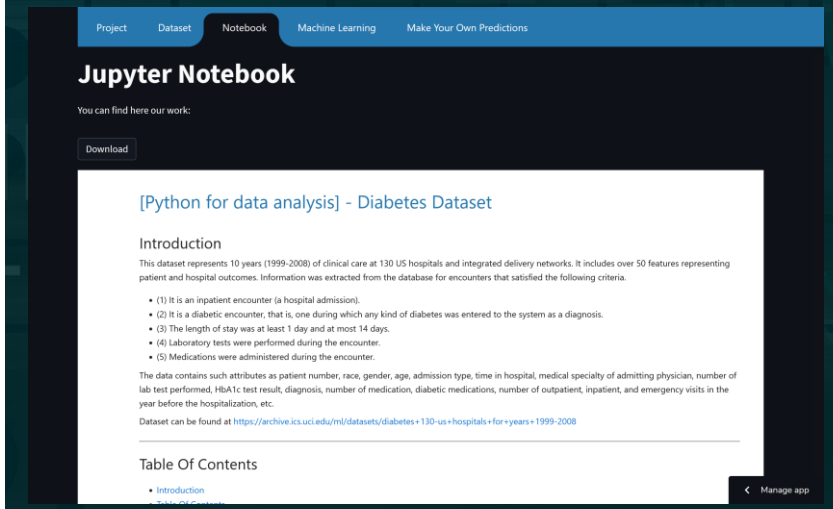
Model	Score	Accuracy
Adaptive Boosting	0.6366	0.6351
Random Forest	0.6359	0.6360
Naive Bayes	0.6359	0.6260
Logistic Regression	0.6238	0.6238
Linear SVC	0.6231	0.6231
Extra Trees	0.6230	0.6201
K-Nearest Neighbors	0.5805	0.5805
Decision Tree	0.5600	0.5575

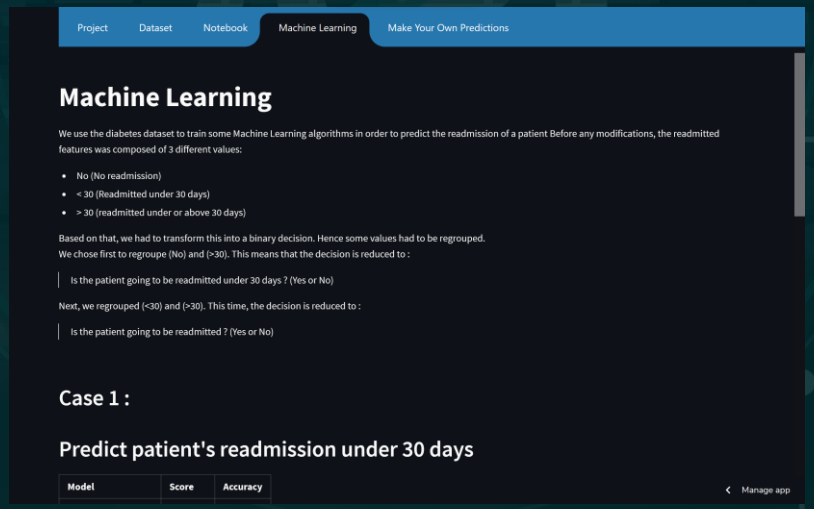
API

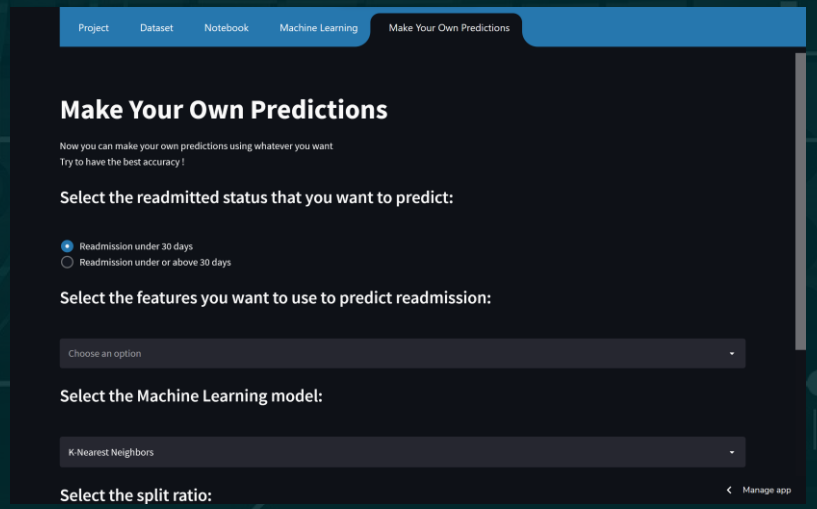
API











Matthieu Thibaut & Chloé Tempo DIA7

TEAM

CHLOÉ TEMPO

MATTHIEU THIBAUT

https://github.com/chlotmpo/python_data_analysis

https://share.streamlit.io/chlotmpo/python_data_analysis/master/API_diabetes(Streamlit)/API_diabetes.py