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

Analysis

Matthieu Thibaut & Chloé Tempo DIA7

DESCRIPTION OF THE DATASET

The dataset represents 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.

50 FEATURES (PART 1)

- **Encounter ID:** Unique identifier of an encounter
- **Patient number:** Unique identifier of a patient
- **Race:** Values: Caucasian, Asian, African American, Hispanic, and other
- **Gender:** Values: male, female, and unknown/invalid
- **Age:** Grouped in 10-year intervals: [0, 10), [10, 20), . . ., [90, 100)
- **Weight:** Weight in pounds
- **Admission type:** Integer identifier corresponding to 9 distinct values, for example, emergency, urgent, elective, newborn, and not available
- **Discharge disposition:** Integer identifier corresponding to 29 distinct values, for example, discharged to home, expired, and not available
- **Admission source:** Integer identifier corresponding to 21 distinct values, for example, physician referral, emergency room, and transfer from a hospital

50 FEATURES (PART 2)

- **Time in hospital:** Integer number of days between admission and discharge
- **Payer code:** Integer identifier corresponding to 23 distinct values, for example, Blue Cross\Blue, Shield, Medicare, and self-pay
- **Medical specialty:** Integer identifier of a specialty of the admitting physician, corresponding to 84 distinct values, for example, cardiology, internal medicine, family\general practice, and surgeon
- **Number of lab procedures:** Number of lab tests performed during the encount
- **Number of procedures:** Number of procedures (other than lab tests) performed during the encounter
- **Number of medications:** Number of distinct generic names administered during the encounter
- **Number of outpatient visits:** Number of outpatient visits of the patient in the year preceding the encounter
- **Number of emergency visits:** Number of emergency visits of the patient in the year preceding the encounter
- **Number of inpatient visits:** Number of inpatient visits of the patient in the year preceding the encounter
- **Diagnosis 1:** "The primary diagnosis (coded as first three digits of ICD9): 848 distinct values"
- **Diagnosis 2:** "Secondary diagnosis (coded as first three digits of ICD9): 923 distinct values"
- **Diagnosis 3:** "Additional secondary diagnosis (coded as first three digits of ICD9): 954 distinct values“
- **Number of diagnoses:** Number of diagnoses entered to the system

50 FEATURES (PART 3)

- **Glucose serum test result:** Indicates the range of the result or if the test was not taken. (“>200,” “>300, “normal,” and “none” if not measured)
- **A1c test result:** 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
- **Change of medications**: Indicates if there was a change in diabetic medications (either dosage or generic name). Values: “change” and “no change”
- **Diabetes medications:** Indicates if there was any diabetic medication prescribed. (YES or NO)
- **24 features for medications:** 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 whetherthe 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 prescri
- **Readmitted:** 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.

STEPS

1 LOAD AND CLEAN THE DATASET

2 ANALYSE , VISUALIAZE AND PREDICT

3 DISCUSS RESULTS

LIBRARIES

VISUALIZATION

matplotlib

seaborn

bokeh

STRUCTURE

NumPy

pandas

MACHINE LEARNING

scikit learn

GOAL OF OUR ANALYSIS

- After observing all the differents variables of this dataset, it seemed logical to focus our attention on the 'readmitted' feature. This column presents 3 possibles responses types : 'No', to indicate that the patient was not readmitted to hospital; '<30', to indicate that the patient was readmitted within 30 days, or '>30', to indicate that he was readmitted after 30 days.

- We will therefore study the other variables to find out if there is a link between them that could provide information on a possible correlationbetween patient-specific data and/or medical data with being or not being readmitted to hospital.

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
```

Looking at the structure of our data, we have observed that the variables are of two different types. 'Int' and 'object'. The type 'object' is very existent (we can see some in this extract and the variables from 24 to 49 are too.)

We can also see that a lot of data is missing in some columns and may hinder our analysis.

So we're going to retrieve what's not going to help us.

We will also have to process these data to make them viewable and usable, so that we can deduce a meaning.

DATA REMOVING

- The first step was to remove the columns that had too many missing values. There are :
  - **'weight'**, for 97% of missing values
  - **'medical_specialty'**, for 49% of missing values
  - **'payer_code'**, for 40% of missing values

- The columns that would not be useful in the analysis and could even interfere with it were then removed :
  - '**encounter_id**', because not useful

  These columns because they provide a constant value for every rows and it is useless
  - '**examide**'
  - **'citoglipton'**
  - '**glimepiride-pioglitazone**'
  - **'metformin-rosiglitazone'**
  - **'metformin-pioglitazone'**

  - We observed too that some rows were redundant, they had the same patient value and can interfere with the analysis too. We decided to remove these rows.

MAPPING AND NEW COLUMNS

By observing with more attention the different values present in the columns, we realized that some needeed to be adapted in order to be able to treat them.

In first case, the '**diag_1**', '**diag_2**' and '**diag_3**' columns had too disparate values. Thanks this image that mapped in value intervals, we were able to transform the values of this column into 8 categories. And to later be able to analyze them with computational algorithms, we created 8 columns corresponding to these categories, with binary values.

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

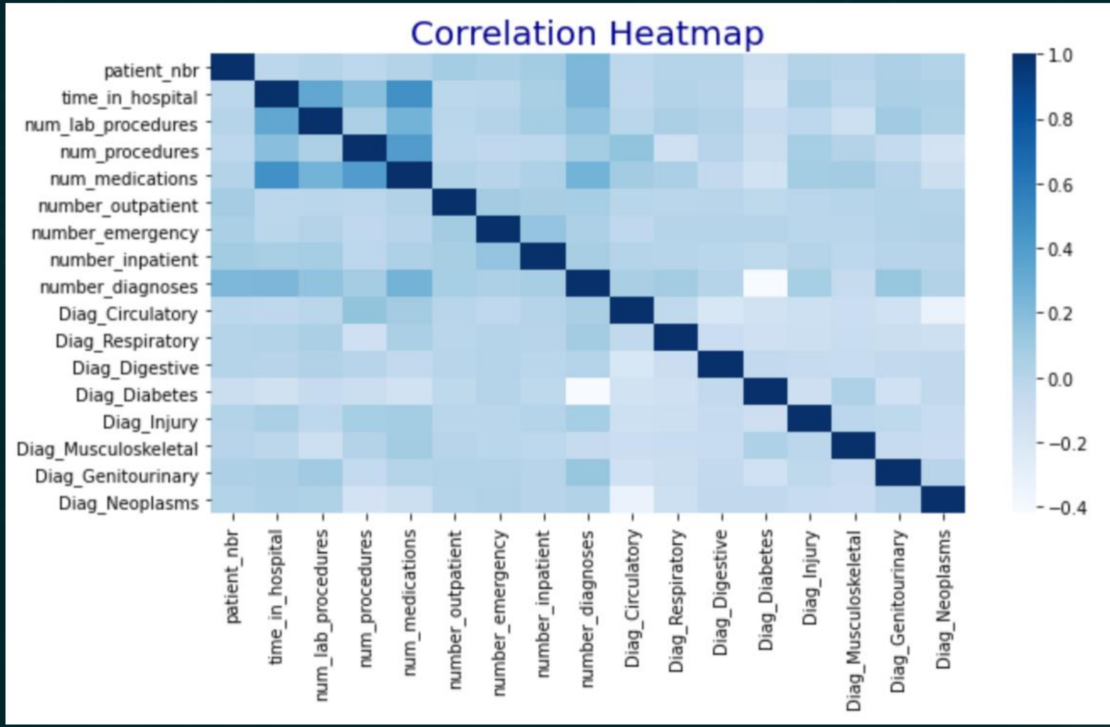
Other columns were mapped using a file provided with the data set. ('**admission_type_id**', '**discharge_disposition_id'** and '**admission_source_id**')

VISUALIZATION (I)

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

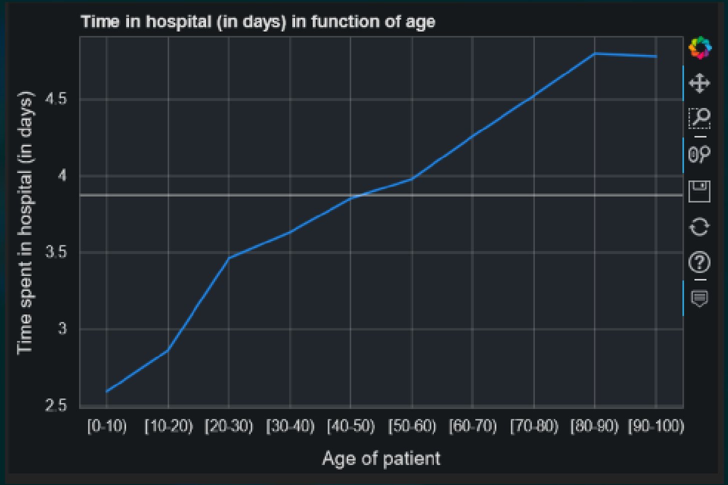
1. Correlation

This graph shows the existing correlations between the different variables in the dataset. A color scale represents the intensity of the correlation and makes it possible to visualize the hidden combinations of the data.



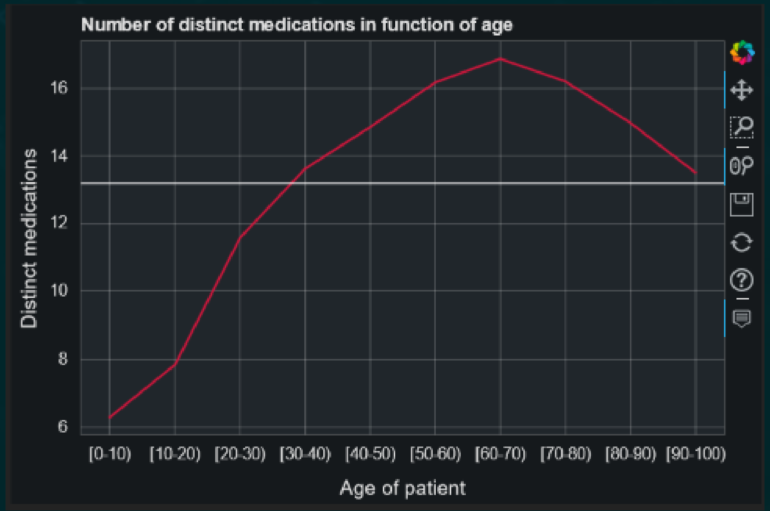
VISUALIZATION (II)

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



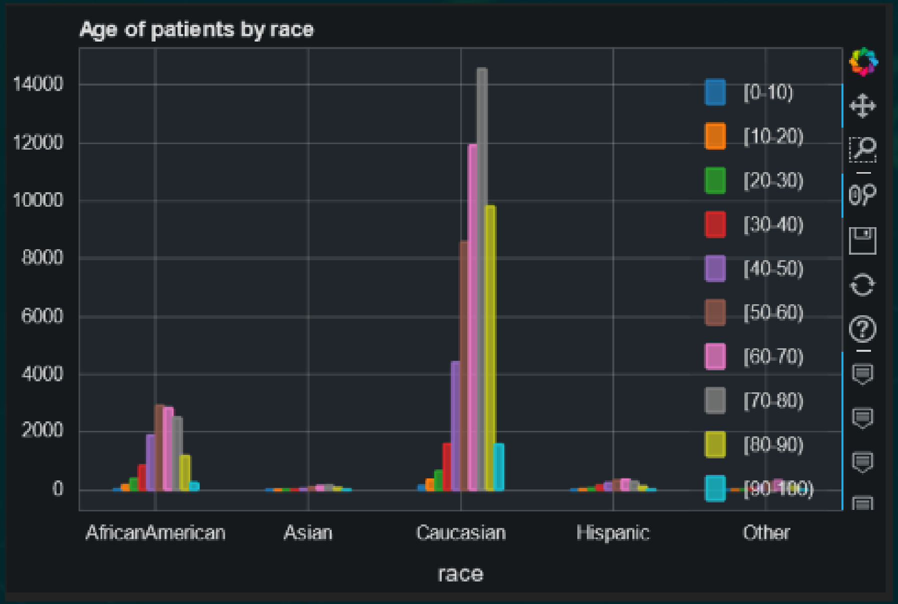
We can see here that there is a relationship between the age of the patient and the time he spent in hospital. On average, the older patients spend more time in hospital than the youngers.

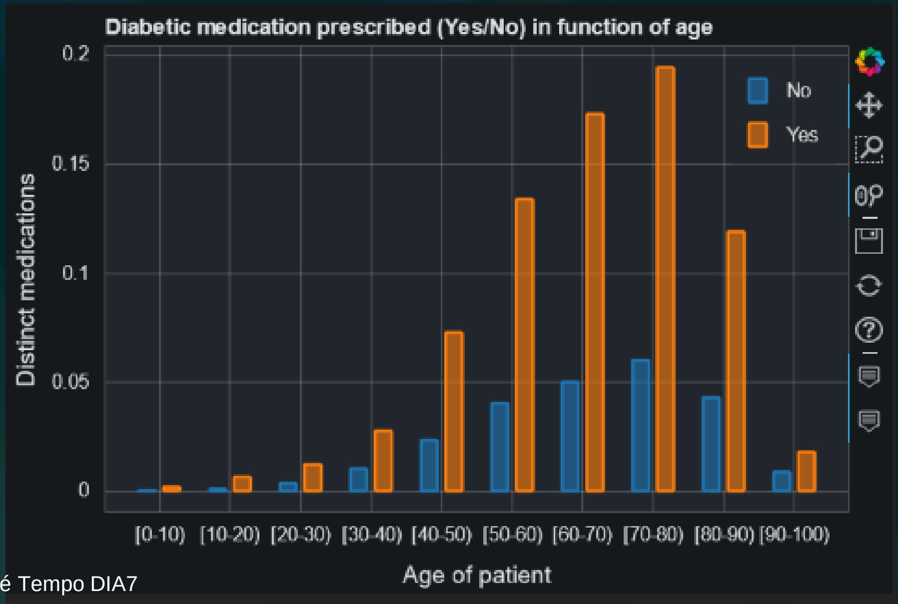
Here, the number of distinct medication seems to depend a lot on the patient's age with a peak for patients around 60-70 years old.
Once again, older patients seems to be more affected.



VISUALIZATION (II)

Race repartition by age category





We can see that the Yes and No values follow the same pattern with many values for patients between 50 and 90 years old but this can simply be explained by the fact that the pattern follows the distribution of patient of this dataset.
But here we can deduce that for a class of age, almost 2 patient over 3 were given diabetic medication

MACHINE LEARNING

The aim of this section is to use the diabetes dataset to train some Machine Learning models using the diabete dataset in order to **predict the readmission of a patient**

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

MODEL SELECTION

In this study, we try to predict a qualitative binary variable. We also have a fairly large set of data, which leads us to use some models more than others.

We therefore tested and implemented the following models to compare their prediction performance :

- K-Nearest Neighbors (KNN)
- Logistic Regression
- Linear SVC
- Random Forest
- Adaptive Boosting
- Decision Tree
- Extra Trees
- Naïve Bayes Classifier

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

DISCUSSION

After observing the results, the following conclusions can be drawn.

For case number 1, we can see that the predictions are made with a rather high score but we must qualify this result because the problem is unbalanced. You can't really rely on those results.

For case number 2, the prediction performance is lower, less accurate, but this result is closer to reality.

We can conclude from this that this dataset is rather little correlated, that the variables seems to have relations between them but without having very large and significant ones.

It is therefore difficult to predict whether or not a patient will be readmitted to hospital with the features available in this dataset. We can put out an idea but it will not be very reliable on the subject.

API

python™

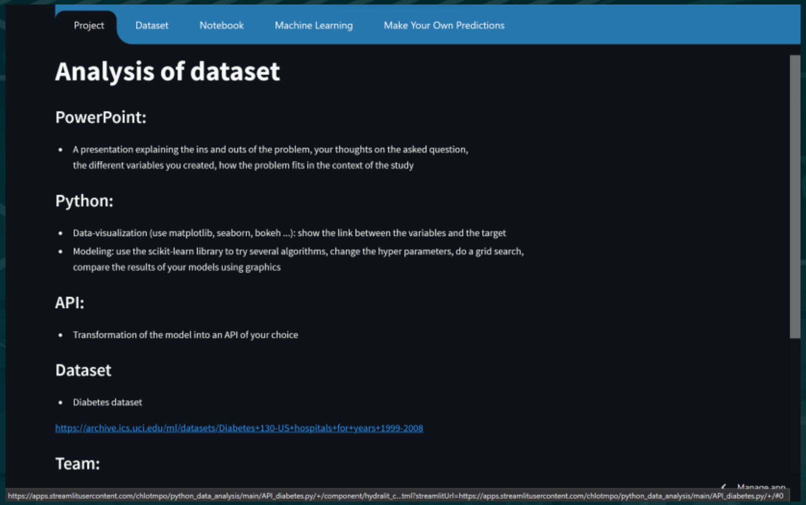
Streamlit

API

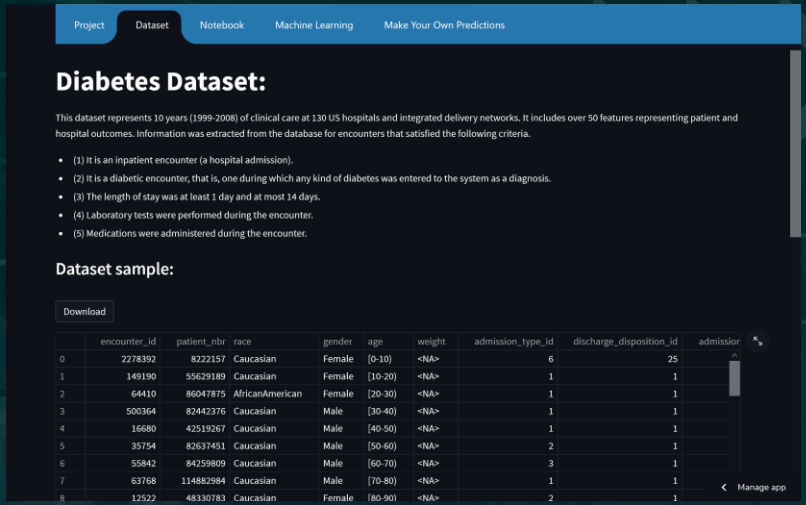
To make the project a little more visual and accessible, we created a streamlit project that we linked to our python code to create an API. It presents different tabs with the description of the dataset, the notebook, our results in machine learning and an interactive part.

In the next slide, you can have a quick view on the design of our api but for a better experience, we can’t help but recommend you to visit the website https://share.streamlit.io/chlotmpo/python_data_analysis/main/API_diabetes.py

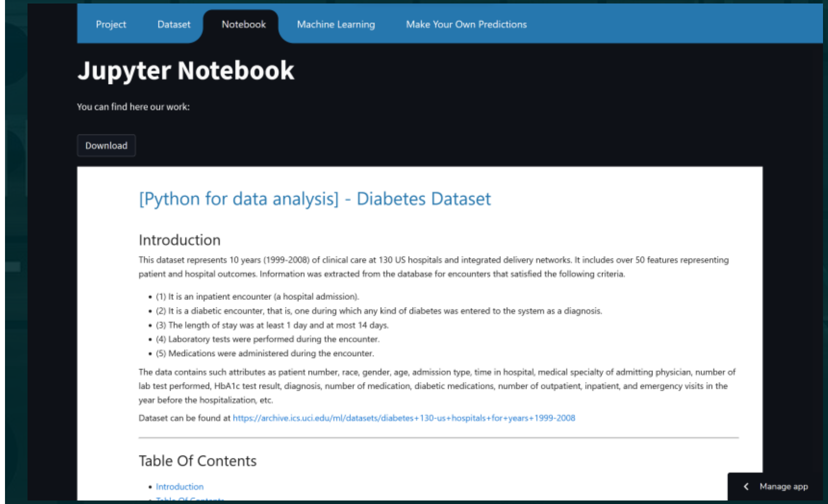
API



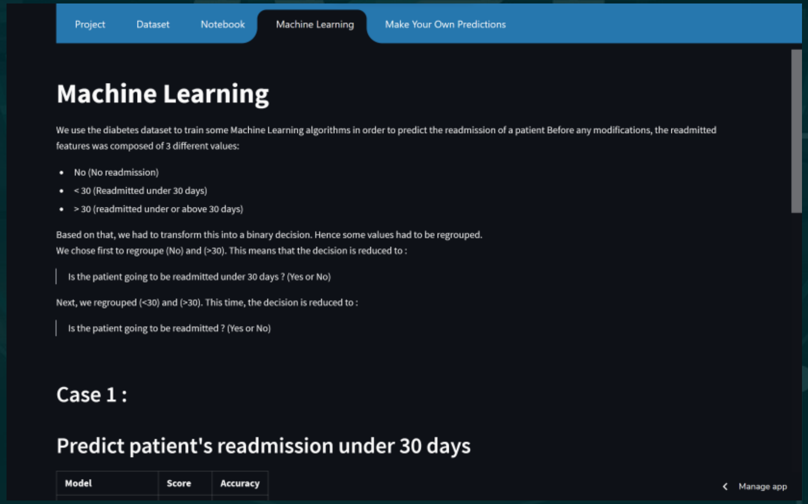
Project
Dataset
Notebook
Machine Learning
Make Your Own Predictions
Analysis of dataset
PowerPoint:
A presentation explaining the ins and outs of the problem, your thoughts on the asked question, the different variables you created, how the problem fits in the context of the study
Python:
Data-visualization (use matplotlib, seaborn, bokeh ...): show the link between the variables and the target
Modeling: use the scikit-learn library to try several algorithms, change the hyper parameters, do a grid search, compare the results of your models using graphics
API:
Transformation of the model into an API of your choice
Dataset
Diabetes dataset
https://archive.ics.uci.edu/ml/datasets/Diabetes+130-US+hospitals+for+years+1999-2008
Team:



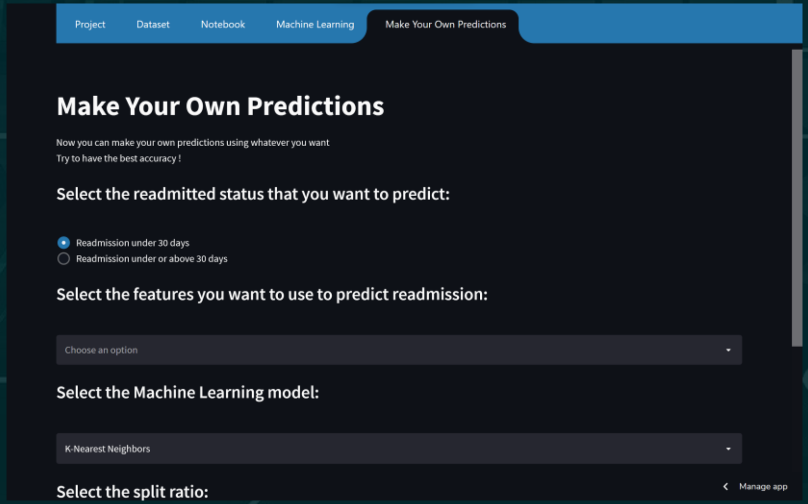
Project
Dataset
Notebook
Machine Learning
Make Your Own Predictions
Diabetes Dataset:
This dataset represents 10 years (1999-2008) of clinical care at 130 US hospitals and integrated delivery networks. It includes over 50 features representing patient and hospital outcomes. Information was extracted from the database for encounters that satisfied the following criteria.
(1) It is an inpatient encounter (a hospital admission).
(2) It is a diabetic encounter, that is, one during which any kind of diabetes was entered to the system as a diagnosis.
(3) The length of stay was at least 1 day and at most 14 days.
(4) Laboratory tests were performed during the encounter.
(5) Medications were administered during the encounter.
Dataset sample:
Download
encounter_id patient_nbr race gender age weight admission_type_id discharge_disposition_id admissior
0 2278392 8222157 Caucasian Female [0-10) <NA> 6 25
1 149190 55629189 Caucasian Female [10-20) <NA> 1 1
2 64410 86047875 AfricanAmerican Female [20-30) <NA> 1 1
3 500364 82442376 Caucasian Male [30-40) <NA> 1 1
4 16680 42519267 Caucasian Male [40-50) <NA> 1 1
5 35754 82637451 Caucasian Male [50-60) <NA> 2 1
6 55842 84259809 Caucasian Male [60-70) <NA> 3 1
7 63768 114882984 Caucasian Male [70-80) <NA> 1 1
8 12522 48330783 Caucasian Female [80-90) <NA> 2 1
Manage app



Project
Dataset
Notebook
Machine Learning
Make Your Own Predictions
Jupyter Notebook
You can find here our work:
Download
[Python for data analysis] - Diabetes Dataset
Introduction
This dataset represents 10 years (1999-2008) of clinical care at 130 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.
(1) It is an inpatient encounter (a hospital admission).
(2) It is a diabetic encounter, that is, one during which any kind of diabetes was entered to the system as a diagnosis.
(3) The length of stay was at least 1 day and at most 14 days.
(4) Laboratory tests were performed during the encounter.
(5) Medications were administered during the encounter.
The data contains such attributes as patient number, race, gender, age, admission type, time in hospital, medical specialty of admitting physician, number of lab test performed, HbA1c test result, diagnosis, number of medication, diabetic medications, number of outpatient, inpatient, and emergency visits in the year before the hospitalization, etc.
Dataset can be found at https://archive.ics.uci.edu/ml/datasets/diabetes+130-us+hospitals+for+years+1999-2008
Table Of Contents
Introduction
Manage app



Project
Dataset
Notebook
Machine Learning
Make Your Own Predictions
Machine Learning
We use the diabetes dataset to train some Machine Learning algorithms in order to predict the readmission of a patient Before any modifications, the readmitted features was composed of 3 different values:
No (No readmission)
< 30 (Readmitted under 30 days)
> 30 (readmitted under or above 30 days)
Based on that, we had to transform this into a binary decision. Hence some values had to be regrouped.
We chose first to regroupe (No) and (>30). This means that the decision is reduced to :
Is the patient going to be readmitted under 30 days ? (Yes or No)
Next, we regrouped (<30) and (>30). This time, the decision is reduced to :
Is the patient going to be readmitted ? (Yes or No)
Case 1 :
Predict patient's readmission under 30 days
Model
Score
Accuracy
Manage app



Project
Dataset
Notebook
Machine Learning
Make Your Own Predictions
Make Your Own Predictions
Now you can make your own predictions using whatever you want
Try to have the best accuracy !
Select the readmitted status that you want to predict:
Readmission under 30 days
Readmission under or above 30 days
Select the features you want to use to predict readmission:
Choose an option
Select the Machine Learning model:
K-Nearest Neighbors
Select the split ratio:
Manage app

TEAM

CHLOÉ TEMPO

MATTHIEU THIBAUT

GitHub

https://github.com/chlotmpo/python_data_analysis

WEBSITE

https://share.streamlit.io/chlotmpo/python_data_analysis/main/API_diabetes.py