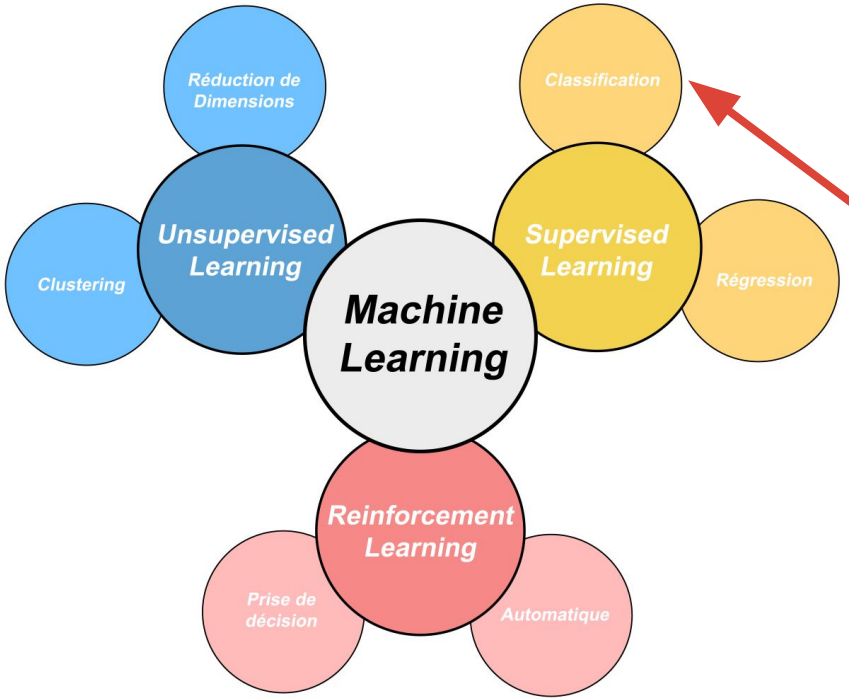
TD: Introduction to Machine-Learning

Briefing & Correction

Corentin Meyer
2nd Year PhD Student - CSTB iCube
corentin.meyer@etu.unistra.fr

Presentation on 04/10/2021
@ ESBS 3A Biotech
UE: Traitement et Flux de données

What's ML and today's goal



Algorithms created to:
Automatically learn from data and make predictions

Today's work:
Supervised-Learning -> Classification -> **Binary Classification**

Classification: predict a category
Regression: predict a value

Your Coding Environnement

Easy Way

Google Collaboratory

Or

Conventional Hard Way

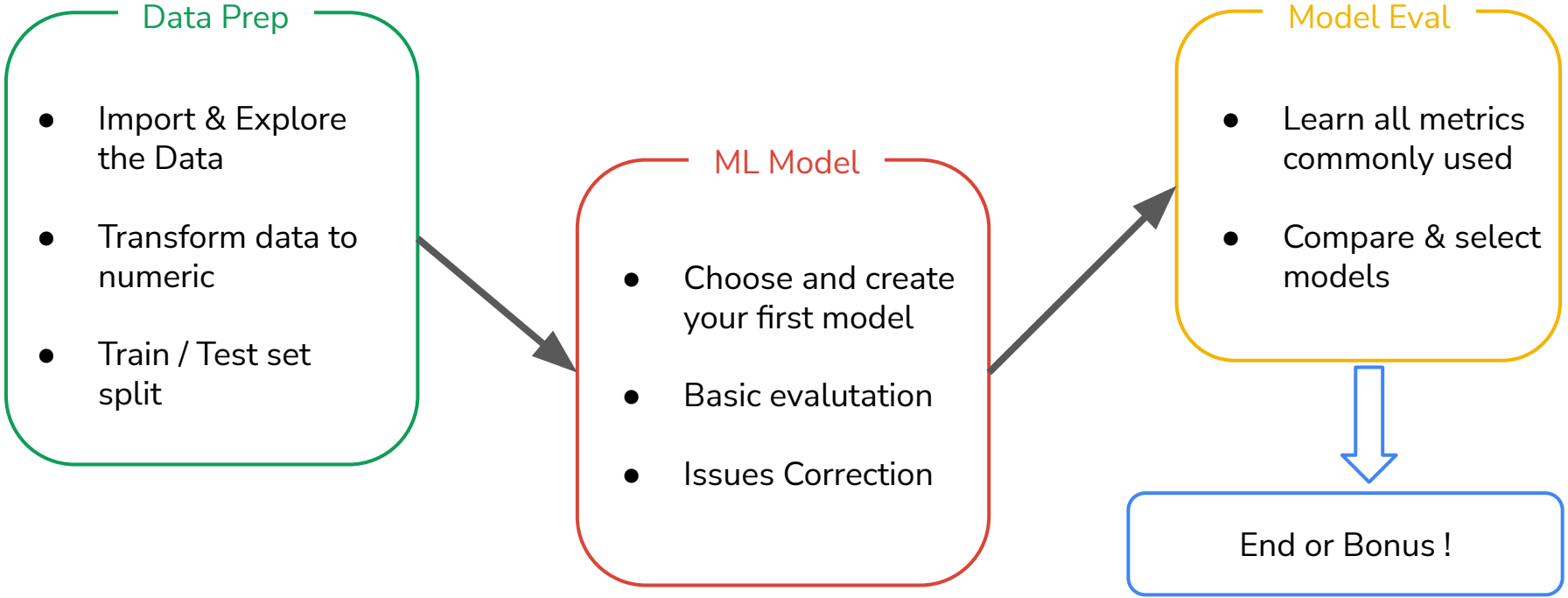
The Data you will use

Stroke Prediction Dataset

-> **Predict** whether a patient is likely to get **stroke**

Using 11 parameters: age, BMI, Glucose level, lifestyle, smoke status...

TD Workflow


Data Prep
Import & Explore the Data
Transform data to numeric
Train / Test set split
ML Model
Choose and create your first model
Basic evalutation
Issues Correction
Model Eval
Learn all metrics commonly used
Compare & select models
End or Bonus !

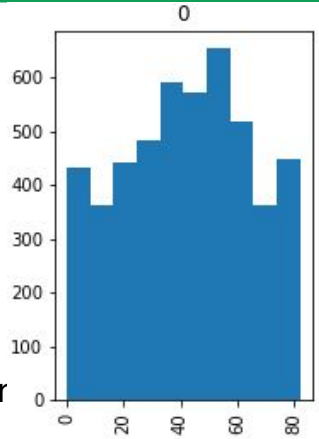
Explore the data

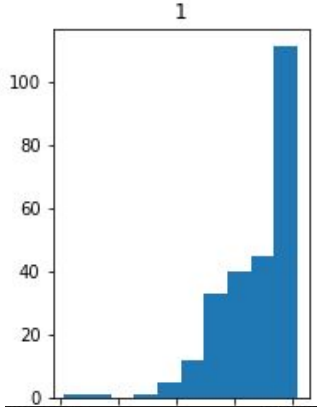
Questions:

1. How many entries (patients) are in the dataset ?
5110 rows in the dataset

2. How many columns (features) ?
11 Features

3. Plot the histogram of the age feature. Do separate histogram for stroke vs non-stroke patients





4. What is the percentage of the patients that had a stroke ?
95%/5%

5. Show the type of data in each columns. What type of processing will we have to do for each type ?
Float -> Scaling ; Objects (text) -> One-Hot Encoding
Int (0/1) -> Nothing to do

```
gender              object
age                float64
hypertension         int64
heart_disease        int64
ever_married        object
work_type           object
Residence_type      object
avg_glucose_level  float64
bmi                float64
smoking_status      object
stroke               int64
dtype: object
```

Format the data to numeric

Questions

1. What columns are categorical data, what columns are numeric.

Numeric: age, avg_glucose_level, bmi

Categorical: gender, ever_married, work_type, Residence_type, smoking_status

2. What columns are already ready to be used and needs no change.

Ready: hypertension, heart_disease, stroke

3. What type of processing do you need to do on categorical data and why

Text to numeric -> Vectors -> One-Hot Encoding

4. What type of processing do you need to do on numeric data and why

Numeric between any values -> Numeric between 0 and 1 or -1/+1 or Z-score !

5. What columns contains missing data ? What type of processing do you need to do in this case.

BMI contains missing data, we need either to removes rows or impute them (predict)

Train / Test data spliting

Questions

1. What train/test ratio should you use.

Something between 20% and 40% is fine

2. How many entries are in your train dataset and in your test dataset.

For 40% I get 3066 entries in training, 2044 in testing set

3. Verify that you have the same stroke / no-stroke ratio between train and test dataset.

4.8% of stroke in training set
5.5% of stroke in testing set, so it's fine !

Model Choice & Evalution

Questions:

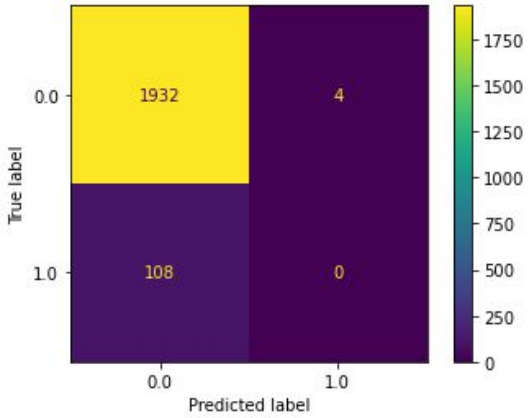
1. Which model did you choose and why ? Have you set any particular (hyper)parameters ?

I choose a RandomForest, because it is widely used in biology and works well.
I used class_weight=”balanced” to try to go against the imbalanced dataset.

2. What accuracy-score do you get and what conclusion can you take ?

94% Accuracy ! Looks great !

3. What do you observe on the confusion matrix and what conclusion can you take ?
Actually, almost all testing data has been classified as “no-stroke”, not so good...



Model Choice & Evalution

Questions:

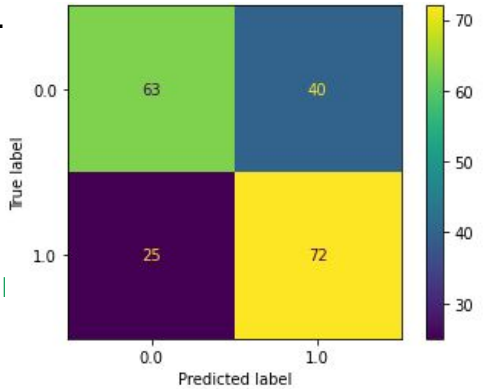
1. What accuracy-score do you get with the new model and what conclusion can you take.

67.5% Accuracy, looks worst than before !

2. What do you observe on the confusion matrix and what conclusion can you take.

At least now we actually predicted 72 patients strokes !

3. What's the shape of the prediction probability output ? Is there a high variance between the different test entries in probability ?
Shape: n entries x 2 columns. Some data points are almost 50% / 50% proba, other are 95% / 5%. We need to set a threshold of confidence for each prediction !



All commonly used metrics

Questions:

1. Which model have the best accuracy ?

Basic one for the basic accuracy, downsampled one for the balanced accuracy !

2. What's ROC-Curve ? Which model have the best area under the curve (AUC) for the ROC-curve ?

ROC is the TPR (true positive rate) vs FDR (false discovery rate) for all prediction proba threshold. Basic model has a better AUC for the roc-curve (only by a very little margin)

3. What's F1-Score ? Which model have the best F1-Score and sensitivity ?

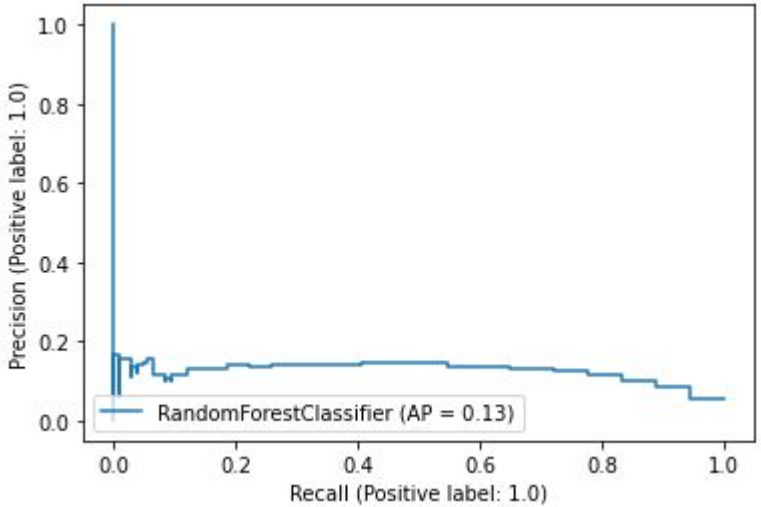
F1-Score is the harmonic mean between precision and recall ! Down-sampled models has a way, way, better F1-Score

4. Eventually, which model is better according to you based on the metrics ?

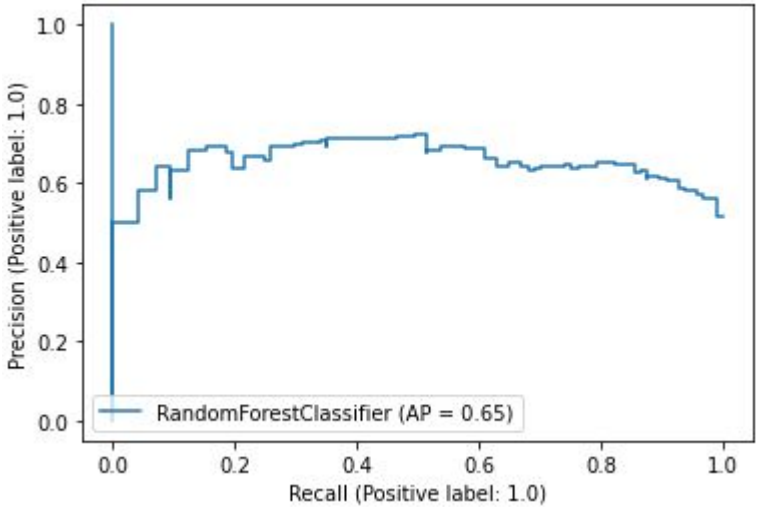
Downsampled in the end, looks better due to huge gap in F1-Score

Models Comparison

	Balanced-Accuracy	Accuracy	F1-Score	Sensitivity (Recall)	Specificity	Precision	TP	TN	FP	FN
CLF	0.498967	0.945205	0.000000	0.000000	0.997934	0.000000	0	1932	4	108
CLF DownSampled	0.676959	0.675000	0.688995	0.742268	0.611650	0.642857	72	63	40	25



Initial Model Precision Recall Curve



Downsampled Model Pression Recall Curve

Bonus Answers !

Questions:

1. What is the point of cross-validation ? Did it increase performance ? If not, what is it useful for ?

It does not increase performance. It just calculate X models on X differents test/train splits ! Actually it is used to give a **confidence interval** on accuracy and other performances as you have now X accuracy values for your model on you dataset ! Makes your models results more robusts !

Questions:

1. What are the best parameters detected ? Are your best parameters different from the one of other students ? Why ?

Yes they are different because it is a stochastic exploration with a limited number of trials !

2. Did the metrics improved ? Was the optimisation useful ?

We wanted to maximize F1-Score and it improved a little bit ! (As well as other metrics)

```
Best trial:
  Score:  0.8306905059398083
  Params:
    n_estimators: 187
    criterion: entropy
    max_depth: 12
    min_samples_split: 29
    min_samples_leaf: 6
    max_features: None
    bootstrap: True
    oob_score: False
    n_jobs: -1
    class_weight: None
```