Théo Matas
Maximilien Frat Leprince

Project Report :

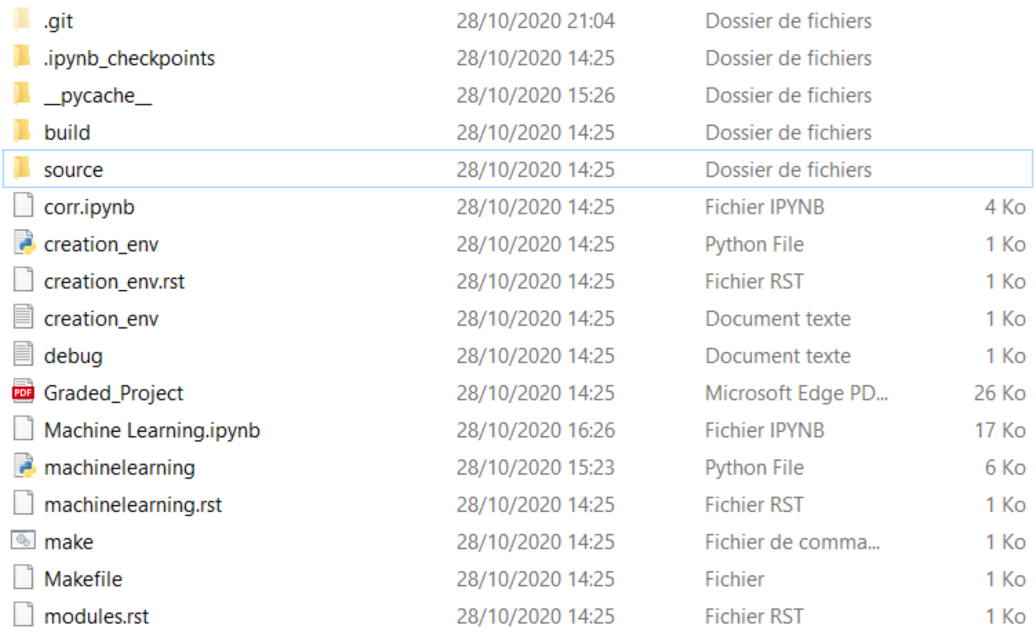
Objective :

The project objective was to construct an architecture of data science project to qualify a dataset of contracts using algorithms trained on a public dataset.

Project structure :

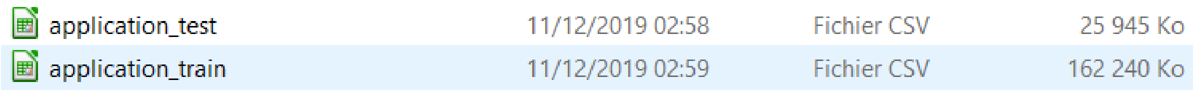
The project was composed of three parts : un git project containing the code, un folder for data and a folder of models saved using mlrun.

Git :



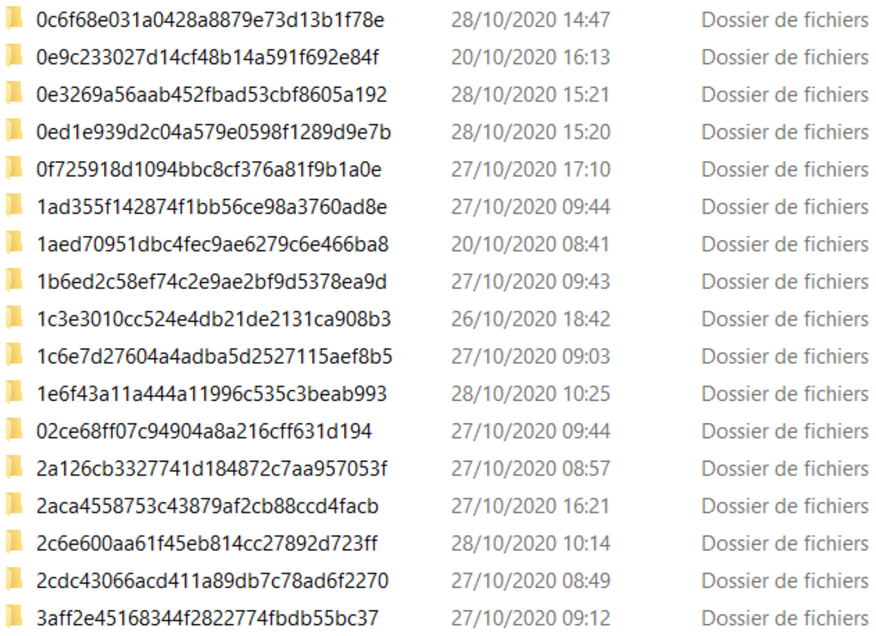
.git	28/10/2020 21:04	Dossier de fichiers	
.ipynb_checkpoints	28/10/2020 14:25	Dossier de fichiers	
__pycache__	28/10/2020 15:26	Dossier de fichiers	
build	28/10/2020 14:25	Dossier de fichiers	
source	28/10/2020 14:25	Dossier de fichiers	
corr.ipynb	28/10/2020 14:25	Fichier IPYNB	4 Ko
creation_env	28/10/2020 14:25	Python File	1 Ko
creation_env.rst	28/10/2020 14:25	Fichier RST	1 Ko
creation_env	28/10/2020 14:25	Document texte	1 Ko
debug	28/10/2020 14:25	Document texte	1 Ko
Graded_Project	28/10/2020 14:25	Microsoft Edge PD...	26 Ko
Machine Learning.ipynb	28/10/2020 16:26	Fichier IPYNB	17 Ko
machinelearning	28/10/2020 15:23	Python File	6 Ko
machinelearning.rst	28/10/2020 14:25	Fichier RST	1 Ko
make	28/10/2020 14:25	Fichier de comma...	1 Ko
Makefile	28/10/2020 14:25	Fichier	1 Ko
modules.rst	28/10/2020 14:25	Fichier RST	1 Ko

Data :



application_test	11/12/2019 02:58	Fichier CSV	25 945 Ko
application_train	11/12/2019 02:59	Fichier CSV	162 240 Ko

Models :

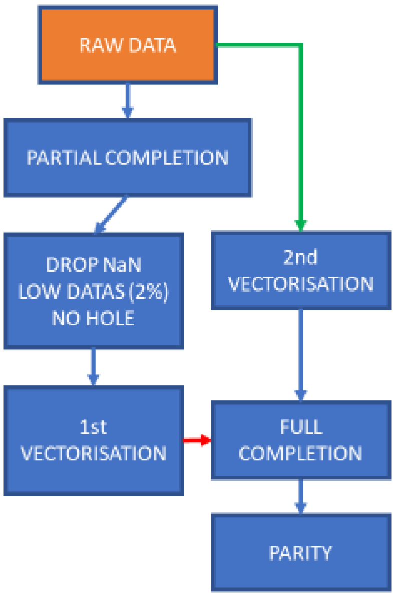


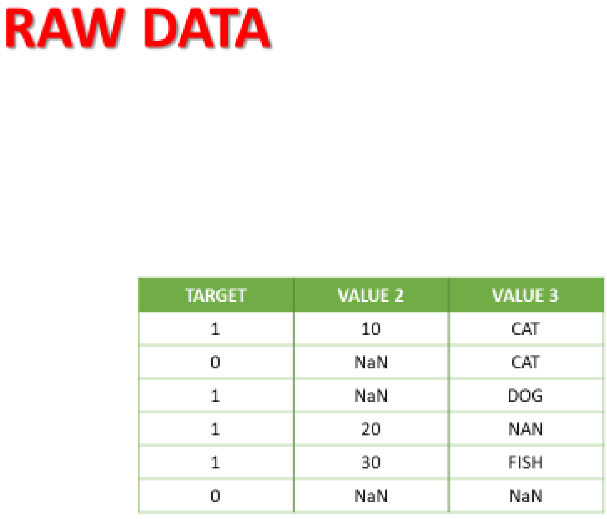
0c6f68e031a0428a8879e73d13b1f78e	28/10/2020 14:47	Dossier de fichiers
0e9c233027d14cf48b14a591f692e84f	20/10/2020 16:13	Dossier de fichiers
0e3269a56aab452fbad53cbf8605a192	28/10/2020 15:21	Dossier de fichiers
0ed1e939d2c04a579e0598f1289d9e7b	28/10/2020 15:20	Dossier de fichiers
0f725918d1094bbc8cf376a81f9b1a0e	27/10/2020 17:10	Dossier de fichiers
1ad355f142874f1bb56ce98a3760ad8e	27/10/2020 09:44	Dossier de fichiers
1aed70951dbc4fec9ae6279c6e466ba8	20/10/2020 08:41	Dossier de fichiers
1b6ed2c58ef74c2e9ae2bf9d5378ea9d	27/10/2020 09:43	Dossier de fichiers
1c3e3010cc524e4db21de2131ca908b3	26/10/2020 18:42	Dossier de fichiers
1c6e7d27604a4adba5d2527115aef8b5	27/10/2020 09:03	Dossier de fichiers
1e6f43a11a444a11996c535c3beab993	28/10/2020 10:25	Dossier de fichiers
02ce68ff07c94904a8a216cff631d194	27/10/2020 09:44	Dossier de fichiers
2a126cb3327741d184872c7aa957053f	27/10/2020 08:57	Dossier de fichiers
2aca4558753c43879af2cb88ccd4facb	27/10/2020 16:21	Dossier de fichiers
2c6e600aa61f45eb814cc27892d723ff	28/10/2020 10:14	Dossier de fichiers
2cdc43066acd411a89db7c78ad6f2270	27/10/2020 08:49	Dossier de fichiers
3aff2e45168344f2822774fbdb55bc37	27/10/2020 09:12	Dossier de fichiers

The code :

We devided the code in three parts : data processing, machine learning and data viz.

Data processing

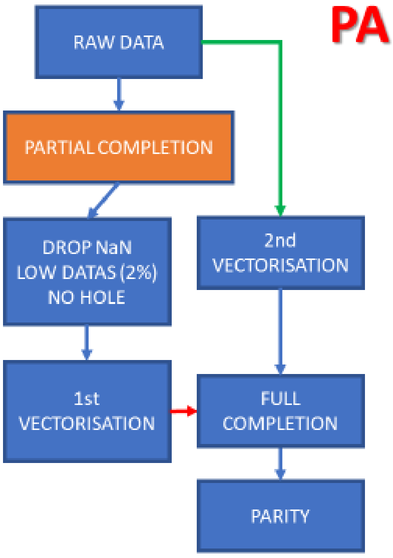




RAW DATA

TARGET	VALUE 2	VALUE 3
1	10	CAT
0	NaN	CAT
1	NaN	DOG
1	20	NAN
1	30	FISH
0	NaN	NaN

PARTIAL COMPLETION



RAW DATA

PARTIAL COMPLETION

DROP NaN LOW DATAS (2%) NO HOLE

2nd VECTORISATION

1st VECTORISATION

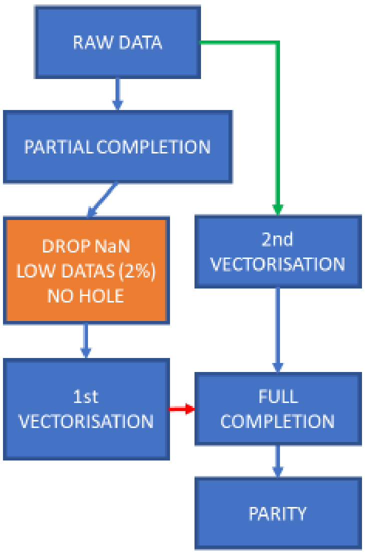
FULL COMPLETION

PARITY

TARGET	VALUE 2	VALUE 3
1	10	CAT
0	NaN	CAT
1	NaN	DOG
1	20	NAN
1	30	FISH
0	NaN	NaN

TARGET	VALUE 2	VALUE 3
1	10	CAT
0	20	CAT
1	20	DOG
1	20	NAN
1	30	FISH
0	20	NaN

DROP NaN



RAW DATA

PARTIAL COMPLETION

DROP NaN LOW DATAS (2%) NO HOLE

2nd VECTORISATION

1st VECTORISATION

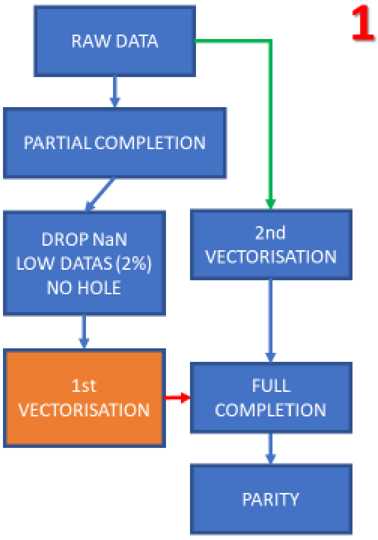
FULL COMPLETION

PARITY

TARGET	VALUE 2	VALUE 3
1	10	CAT
0	20	CAT
1	20	DOG
1	20	NAN
1	30	FISH
0	20	NaN

TARGET	VALUE 2	VALUE 3
1	10	CAT
0	20	CAT
1	20	DOG
1	30	FISH

1st VECTORISATION



RAW DATA

PARTIAL COMPLETION

DROP NaN LOW DATAS (2%) NO HOLE

2nd VECTORISATION

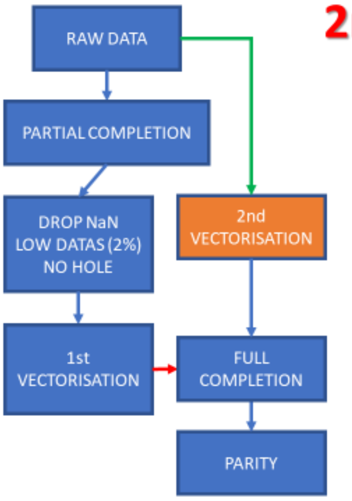
1st VECTORISATION

FULL COMPLETION

PARITY

TARGET	VALUE 2	VALUE 3
1	10	CAT
0	20	CAT
1	20	DOG
1	30	FISH

TARGET	VALUE 2	VALUE 3
1	10	0
0	20	0
1	20	2
1	30	1



RAW DATA

PARTIAL COMPLETION

DROP NaN
LOW DATAS (2%)
NO HOLE

1st
VECTORISATION

2nd
VECTORISATION

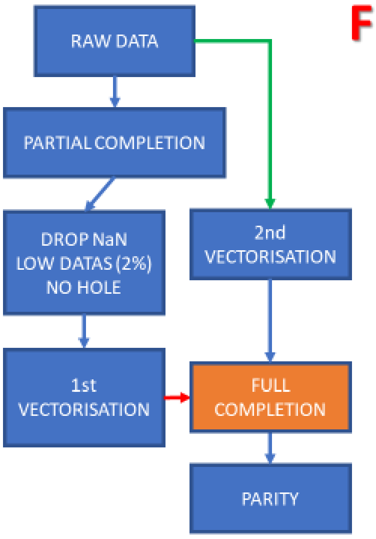
FULL
COMPLETION

PARITY

2nd VECTORISATION

TARGET	VALUE 2	VALUE 3
1	10	CAT
0	NaN	CAT
1	NaN	DOG
1	20	NAN
1	30	FISH
0	NaN	NaN

TARGET	VALUE 2	VALUE 3
1	10	0
0	NaN	0
1	NaN	1
1	20	NAN
1	30	2
0	NaN	NaN



RAW DATA

PARTIAL COMPLETION

DROP NaN
LOW DATAS (2%)
NO HOLE

1st
VECTORISATION

2nd
VECTORISATION

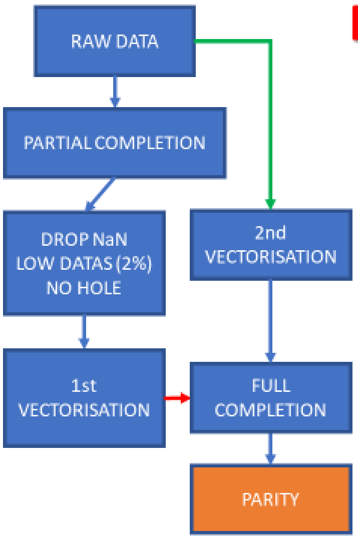
FULL
COMPLETION

PARITY

FULL COMPLETION

TARGET	VALUE 2	VALUE 3
1	10	0
0	NaN	0
1	NaN	1
1	20	NAN
1	30	2
0	NaN	NaN

TARGET	VALUE 2	VALUE 3
1	10	0
0	20	0
1	20	1
1	20	0,75
1	30	2
0	20	0,75



RAW DATA

PARTIAL COMPLETION

DROP NaN
LOW DATAS (2%)
NO HOLE

1st
VECTORISATION

2nd
VECTORISATION

FULL
COMPLETION

PARITY

PARITY

TARGET	VALUE 2	VALUE 3
1	10	0
0	20	0
1	20	1
1	20	0,75
1	30	2
0	20	0,75

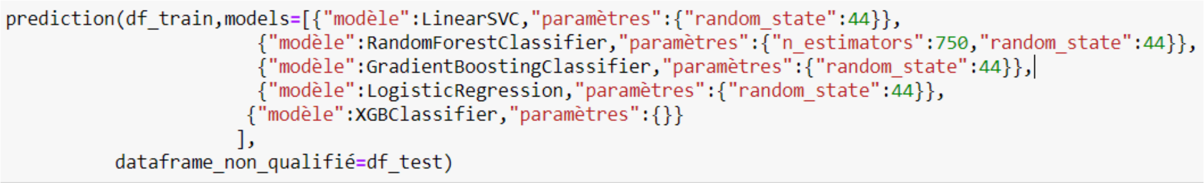
TARGET	VALUE 2	VALUE 3
1	10	0
0	20	0
1	20	1
0	20	0,75

Machine learning :

For this project we have used five different models:

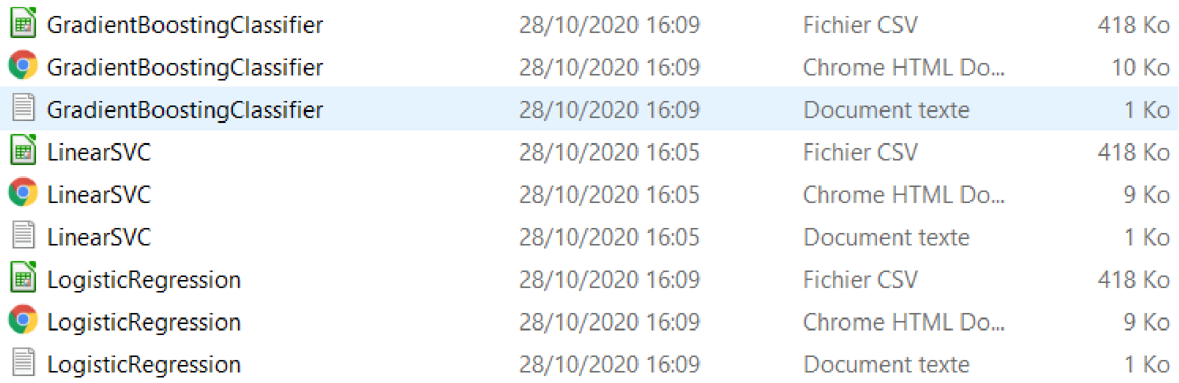
- LinearSVC
- RandomForestClassifier
- GradientBoostingClassifier
- LogisticRegression
- XGBClassifier

We created a function in the file machinelearning.py that would take as arguments a dataset, a list of models with parameters :



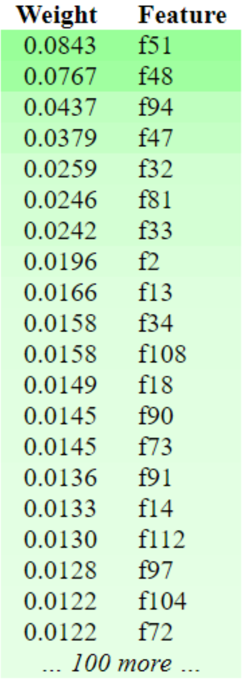
```
prediction(df_train,models=[{"modèle":LinearSVC,"paramètres":{"random_state":44}},
                             {"modèle":RandomForestClassifier,"paramètres":{"n_estimators":750,"random_state":44}},
                             {"modèle":GradientBoostingClassifier,"paramètres":{"random_state":44}},
                             {"modèle":LogisticRegression,"paramètres":{"random_state":44}},
                            {"modèle":XGBClassifier,"paramètres":{}}
                           ],
             dataframe_non_qualifié=df_test)
```

The models would be trained and saved with some statistics saved using mlrun. Also, some csv files of predictions and vis would be produced.



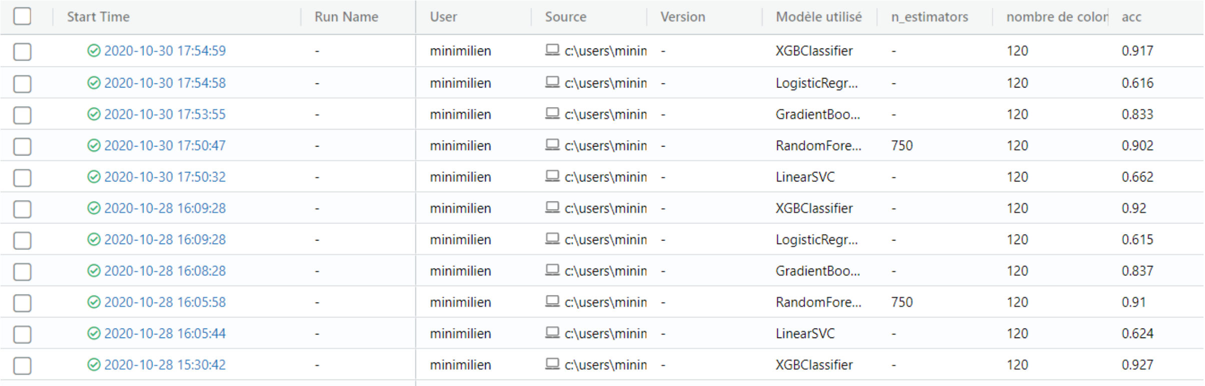
GradientBoostingClassifier	28/10/2020 16:09	Fichier CSV	418 Ko
GradientBoostingClassifier	28/10/2020 16:09	Chrome HTML Do...	10 Ko
GradientBoostingClassifier	28/10/2020 16:09	Document texte	1 Ko
LinearSVC	28/10/2020 16:05	Fichier CSV	418 Ko
LinearSVC	28/10/2020 16:05	Chrome HTML Do...	9 Ko
LinearSVC	28/10/2020 16:05	Document texte	1 Ko
LogisticRegression	28/10/2020 16:09	Fichier CSV	418 Ko
LogisticRegression	28/10/2020 16:09	Chrome HTML Do...	9 Ko
LogisticRegression	28/10/2020 16:09	Document texte	1 Ko

The HTML files would contain details about the models :



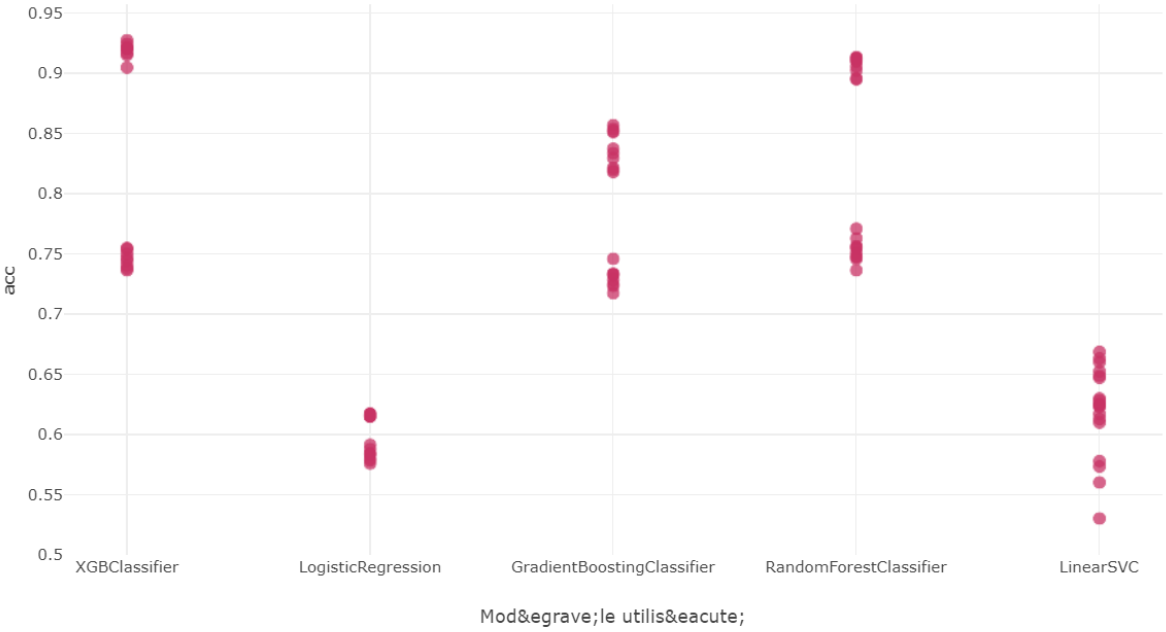
Weight	Feature
0.0843	f51
0.0767	f48
0.0437	f94
0.0379	f47
0.0259	f32
0.0246	f81
0.0242	f33
0.0196	f2
0.0166	f13
0.0158	f34
0.0158	f108
0.0149	f18
0.0145	f90
0.0145	f73
0.0136	f91
0.0133	f14
0.0130	f112
0.0128	f97
0.0122	f104
0.0122	f72
... 100 more ...	

Mlruns allows to have informations on the models :

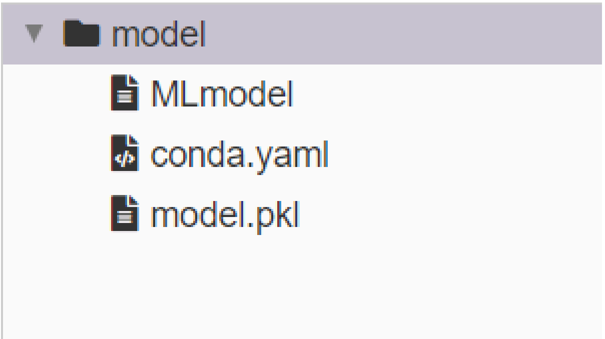


☐	Start Time	Run Name	User	Source	Version	Modèle utilisé	n_estimators	nombre de color	acc
☐	✓ 2020-10-30 17:54:59	-	minimilien	c:\users\minin	-	XGBClassifier	-	120	0.917
☐	✓ 2020-10-30 17:54:58	-	minimilien	c:\users\minin	-	LogisticRegr...	-	120	0.616
☐	✓ 2020-10-30 17:53:55	-	minimilien	c:\users\minin	-	GradientBoo...	-	120	0.833
☐	✓ 2020-10-30 17:50:47	-	minimilien	c:\users\minin	-	RandomFore...	750	120	0.902
☐	✓ 2020-10-30 17:50:32	-	minimilien	c:\users\minin	-	LinearSVC	-	120	0.662
☐	✓ 2020-10-28 16:09:28	-	minimilien	c:\users\minin	-	XGBClassifier	-	120	0.92
☐	✓ 2020-10-28 16:09:28	-	minimilien	c:\users\minin	-	LogisticRegr...	-	120	0.615
☐	✓ 2020-10-28 16:08:28	-	minimilien	c:\users\minin	-	GradientBoo...	-	120	0.837
☐	✓ 2020-10-28 16:05:58	-	minimilien	c:\users\minin	-	RandomFore...	750	120	0.91
☐	✓ 2020-10-28 16:05:44	-	minimilien	c:\users\minin	-	LinearSVC	-	120	0.624
☐	✓ 2020-10-28 15:30:42	-	minimilien	c:\users\minin	-	XGBClassifier	-	120	0.927

To compare them :



To download them :


model
MLmodel
conda.yaml
model.pkl