ANNEXE

Data Challenge ENS
Qube Research & Technology
Reconstruction of Liquid Asset Performance

Elyas Benyamina
Alexis Ibrahim
Antoine Mathis

Professeur : Bertrand Michel
Cours : Apprentissage statistique

2021-2022

Table des matières

1 Présentation du problème **1**

2 Valeurs manquantes **2**

3 Répartitions des actifs **3**

4 K-neighbors Classifier **3**

5 Feature Engineering **4**

5.1 Matrice de Corrélation 4

5.2 Plus proches vosisins 5

5.3 Selection des features 7

1 Présentation du problème

On présente ici les données à notre disposition. Le dataset X_{train} contient différentes colonnes. On retrouve 100 colonnes qui représente la performance de 100 actifs illiquides de la forme 'RET_'+ IDdel'actif, une colonne permettant d'identifier à quel jour est associé la ligne sur laquelle on se trouve (colonne 'ID_DAY') et une colonne permettant de désigner l'ID de l'actif liquide qui doit être prédit à partir des valeurs sur la même ligne.

ID	ID_DAY	RET_216	RET_238	RET_45	RET_295	RET_230	RET_120	RET_188	RET_260	RET_15	...	RET_122	RET_194	RET_72	RET_293	RE
0	3316	0.004024	0.009237	0.004967	NaN	0.017040	0.013885	0.041885	0.015207	-0.003143	...	0.007596	0.015010	0.014733	-0.000476	0.0
1	3316	0.004024	0.009237	0.004967	NaN	0.017040	0.013885	0.041885	0.015207	-0.003143	...	0.007596	0.015010	0.014733	-0.000476	0.0
2	3316	0.004024	0.009237	0.004967	NaN	0.017040	0.013885	0.041885	0.015207	-0.003143	...	0.007596	0.015010	0.014733	-0.000476	0.0
3	3316	0.004024	0.009237	0.004967	NaN	0.017040	0.013885	0.041885	0.015207	-0.003143	...	0.007596	0.015010	0.014733	-0.000476	0.0
4	3316	0.004024	0.009237	0.004967	NaN	0.017040	0.013885	0.041885	0.015207	-0.003143	...	0.007596	0.015010	0.014733	-0.000476	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
267095	3028	0.025293	-0.003277	-0.028823	-0.006021	-0.008381	0.006805	0.018665	-0.010479	0.005288	...	0.013698	-0.007358	0.022241	0.008688	0.0
267096	3028	0.025293	-0.003277	-0.028823	-0.006021	-0.008381	0.006805	0.018665	-0.010479	0.005288	...	0.013698	-0.007358	0.022241	0.008688	0.0
267097	3028	0.025293	-0.003277	-0.028823	-0.006021	-0.008381	0.006805	0.018665	-0.010479	0.005288	...	0.013698	-0.007358	0.022241	0.008688	0.0
267098	3028	0.025293	-0.003277	-0.028823	-0.006021	-0.008381	0.006805	0.018665	-0.010479	0.005288	...	0.013698	-0.007358	0.022241	0.008688	0.0
267099	3028	0.025293	-0.003277	-0.028823	-0.006021	-0.008381	0.006805	0.018665	-0.010479	0.005288	...	0.013698	-0.007358	0.022241	0.008688	0.0

267100 rows × 102 columns

Le dataset Y_{train} contient les données cibles à savoir, la performance de l'actif liquide à prédire ainsi qu'un colonne ID qui correspond à l'index.

ID	RET_TARGET
0	-0.022351
1	-0.011892
2	-0.015285
3	-0.019226
4	0.006644
...	...
267095	0.002080
267096	-0.002565
267097	-0.018406
267098	0.045101
267099	0.005056

267100 rows × 1 columns

Afin de répondre au mieux au problème et coller au plus près à la métrique du problème, on transforme notre problème en un problème de classification.

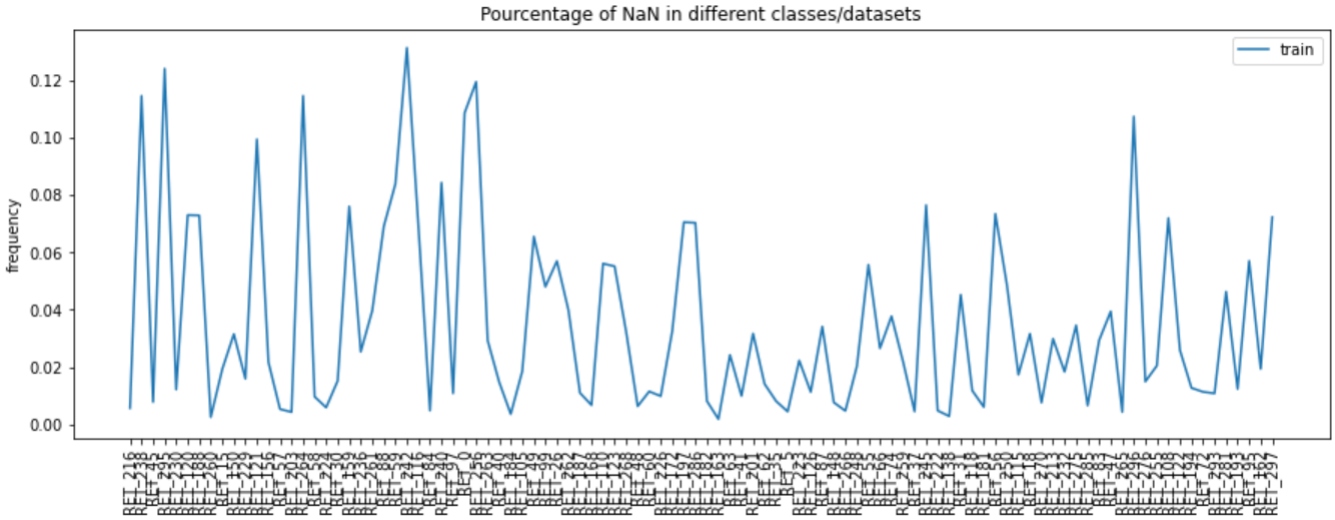
Pour les valeurs positives, on attribue la valeur 1 et pour celles négatives on attribue la valeur -1.

ID	RET_TARGET_CLASS
0	-1
1	-1
2	-1
3	-1
4	1
...	...
267095	1
267096	-1
267097	-1
267098	1
267099	1

267100 rows × 1 columns

2 Valeurs manquantes

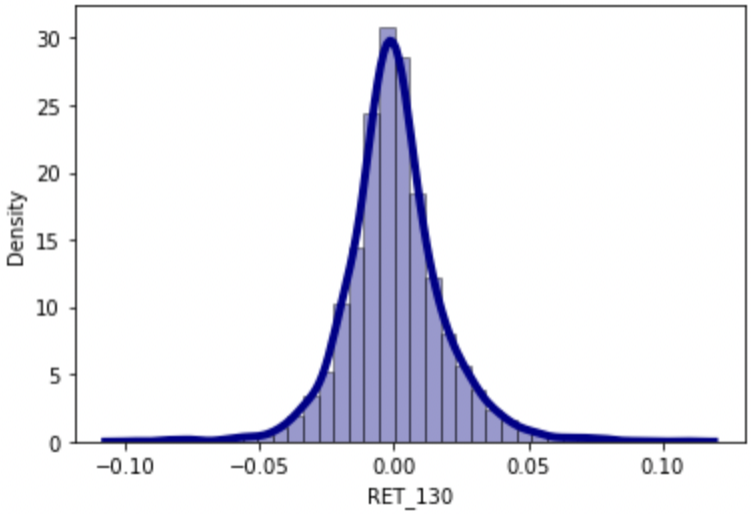
Comme on peut le voir ici, notre dataset d'entrainement, présente un certain nombre de valeurs manquantes. Ce nombre peut varier significativement et atteindre jusqu'à 13% comme le montre le graphique suivant :



Pour pallier à ce problème différentes, nous avons eu plusieurs approches décrites dans le rapport.

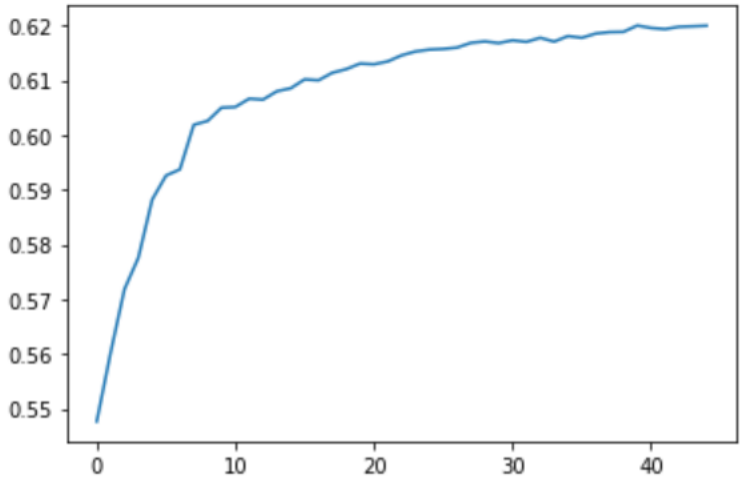
3 Répartitions des actifs

Nous remarquons que la valeur des actifs semble suivre une répartition gaussienne. A titre d'illustration, voici la répartition de l'actif 130.



4 K-neighbors Classifier

Evolution des scores en fonction du nombre de plus proches voisins choisis. On voit que la courbe décolle au début. Néanmoins, la méthode des plus proches voisins permet difficilement d'atteindre des scores supérieurs à 0.65. En abscisse le nombre de plus proches voisins, et en ordonnée le score :



5 Feature Engineering

5.1 Matrice de Corrélation

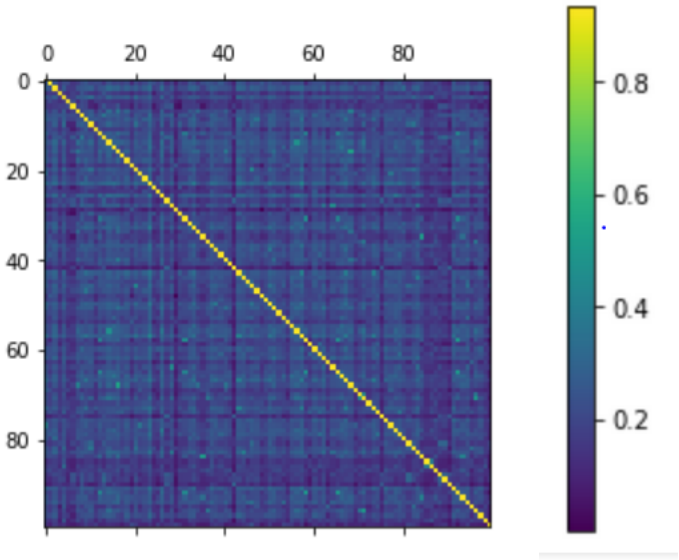
Afin d'évaluer l'indépendance ou l'inter-dépendance de nos variables, il peut être utile d'utiliser une matrice de corrélation.
Une matrice de corrélation est utilisée pour évaluer la dépendence de plusieurs variables en même temps. Le résultat est une table contenant les coefficients de corrélation entre chaque variable et les autres.

La librairie pandas permet de calculer la matrice de correlation de nos dataframes. Cette matrice nous a permis de voir quels lien existaient entre nos features et de tester l'indépendance de ceux-ci (l'hypothèse d'indépendance est important notamment pour des modèles comme Naîve Bayes). Ci-dessous on représente les premières lignes de la matrice. On voit que les valeurs peuvent varier entre 0.05 et 0.40 environ. La corrélation entre certains actifs peut donc être assez importante et il peut alors y avoir une certaine redondance.

```
The correlation DataFrame is:
          RET_216   RET_238    RET_45   RET_295   RET_230   RET_120   RET_188  \
RET_216  1.000000  0.290556  0.307389  0.192316  0.312515  0.223624  0.215869
RET_238  0.290556  1.000000  0.302032  0.200759  0.281364  0.238525  0.218923
RET_45   0.307389  0.302032  1.000000  0.192448  0.308940  0.214584  0.203748
RET_295  0.192316  0.200759  0.192448  1.000000  0.216340  0.178906  0.245587
RET_230  0.312515  0.281364  0.308940  0.216340  1.000000  0.205517  0.224344
...           ...       ...       ...       ...       ...       ...       ...
RET_281  0.242030  0.219768  0.237278  0.153407  0.325866  0.147786  0.140722
RET_193  0.248730  0.195596  0.263676  0.157176  0.241165  0.154444  0.169908
RET_95   0.283889  0.277490  0.323173  0.199405  0.269425  0.263912  0.208522
RET_162  0.195464  0.185792  0.205889  0.115382  0.305888  0.123634  0.162926
RET_297  0.195431  0.178038  0.205409  0.187885  0.193273  0.155867  0.134513

          RET_260    RET_15   RET_150  ...   RET_108   RET_122   RET_194  \
RET_216  0.277588  0.299477  0.337627  ...  0.120924  0.350959  0.227415
RET_238  0.215219  0.282123  0.317240  ...  0.127396  0.329848  0.233206
RET_45   0.264626  0.283780  0.309432  ...  0.174409  0.305434  0.268368
RET_295  0.192359  0.296062  0.244596  ...  0.230978  0.237861  0.200576
RET_230  0.312375  0.315839  0.332104  ...  0.139398  0.289553  0.255334
...           ...       ...       ...  ...       ...       ...       ...
RET_281  0.248959  0.204399  0.285392  ...  0.074022  0.204993  0.197135
RET_193  0.268529  0.247045  0.221643  ...  0.113493  0.271154  0.307005
RET_95   0.256026  0.287325  0.303066  ...  0.177961  0.335225  0.283045
RET_162  0.233991  0.179836  0.248184  ...  0.085689  0.206265  0.134786
RET_297  0.138981  0.192684  0.212572  ...  0.113504  0.212589  0.085287
```

On peut utiliser le module matshow pour une représentation plus claire de la matrice de correlation.



On voit que certains actifs illiquides sont totalement décoréllés des autres. C'est le cas notamment des actifs illiquides numérotés entre 85 et 95.

5.2 Plus proches vosisins

Afin de voir la nature du lien qui existait entre les différents actifs nous avons mis en oeuvre une classification pour chaque actif liquide. Cette classification consiste à repertorier les actifs illiquides "les plus proches" en fonction de leurs "classes levels". Le classe level permet de savoir si un actif illiquide est spécialisé dans tel ou tel domaine de l'industrie comme précisé sur la page du challenge :

The supplementary dataset is composed of 5 columns:

- • ID_asset: the ID of a liquid or illiquid asset
- • CLASS_LEVEL_j with $1 \leq j \leq 4$: a sector/industry group identifier to which belongs the corresponding asset. The higher the level j , the more specific the industry domain is.

267100 samples corresponding to 2748 unique days are available for the training datasets while 114468 samples corresponding to 1177 unique days are used for the test datasets.

The train/test split has been performed randomly along the day variable. As a consequence, no day is shared between the training and test datasets.

ID_asset	CLASS_LEVEL_1	CLASS_LEVEL_2	CLASS_LEVEL_3	CLASS_LEVEL_4
216	2	2	12	20
238	2	2	12	21
45	3	5	20	32
295	10	22	49	77
230	4	10	28	47
...	...	...	...	...
241	3	8	26	42
214	2	2	13	22
102	1	1	5	12
145	2	2	12	20
155	2	2	7	14

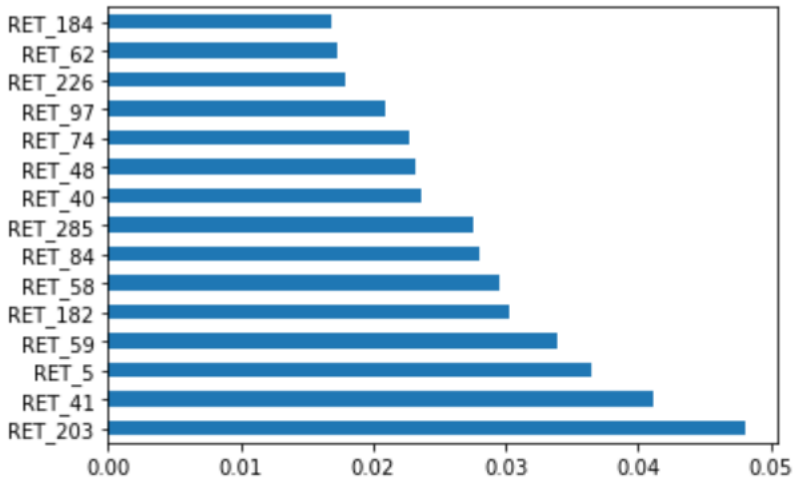
200 rows × 4 columns

Cette approche bien qu'elle ait pu nous paraître intéréssante ne nous a pas permis d'obtenir des résultats significativement meilleurs.

5.3 Selection des features

Comme mentionné dans le rapport, certaines features peuvent avoir une plus grande importance que d'autres pour notre classification. Aussi, il peut s'avérer utile de sélectionner seulement les features les plus déterminants pour entrainer nos modèles. Différentes approches sont possibles, nous présentons ici les résultats en lien avec Random Forest.

Il est par exemple possible de selectionner les features les plus importants en entraînant un modèle de classification Random Forest. On obtient la classification des features suivante :



Ainsi, en fonction du modèle souhaité et de la compléxité associée, il nous a été possible de choisir uniquement les actifs les plus déterminants afin d'essayer d'optimiser le score obtenu.