Rapport Data Analytics

1. Composition de l'équipe

Le code se trouve sur github : https://github.com/Barnini-Nicholas/Data-Analytics-Python
Notre équipe est composée de 4 personnes :

- Karl Ferrero (alternant Air France)
- Cécile Melay (alternant Air France)
- Julien Hubert (alternant Thales)
- Nicholas Barnini (alternant Orange)

2. Technologies

Nous avons pour ce projet préféré développer en python car nous avions déjà quelques connaissances avec le framework Sklearn et pandas de part nos auto-formations et notre expérience professionnel.

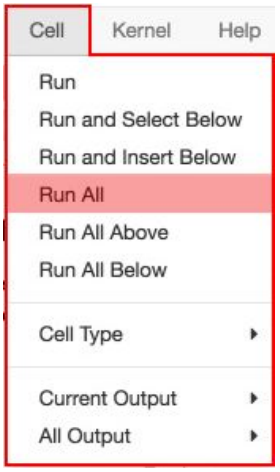
Ce projet a été développé sous Jupyter Notebook permettant une avancée plus rapide grâce aux multiple avantages donné par le notebook, en voici une description : "*Notebook documents (or "notebooks", all lower case) are documents produced by the Jupyter Notebook App, which contain both computer code (e.g. python) and rich text elements (paragraph, equations, figures, links, etc…). Notebook documents are both human-readable documents containing the analysis description and the results (figures, tables, etc..) as well as executable documents which can be run to perform data analysis.*"

Il permet également une exécution du code bloc par bloc tout en gardant en mémoire le code exécuté précédemment permettant de n'avoir qu'à relancer le code nécessaire.

3. Comment lancer le projet ?

Nécessite Jupyter Notebook et l'installation de 2 packages en plus : missingno (permet de réaliser des graphiques sur les valeurs NaN par colonne) ainsi que catboost (gradient boosting).

Ensuite il faut exécuter le notebook "*cluster_vehicule EDA.ipynb*" puis "*Client_EDA.ipynb*" en utilisant pour les deux le "Run All" comme sur cette image ci-dessous.



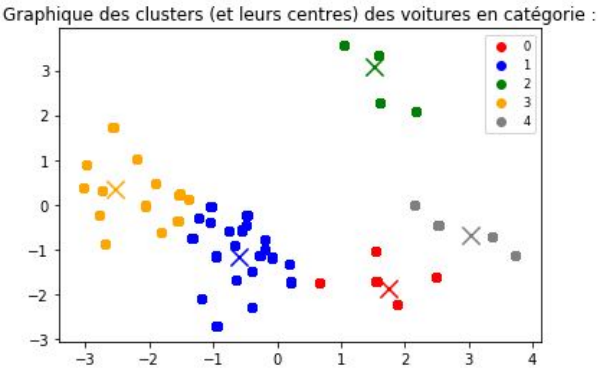
4. Choix effectués lors du projet en termes de gestion et d'analyse des données

Pour prédire un véhicule pour des nouveaux clients nous avons découpé notre projet en deux grandes parties:

- Une clusterisation des véhicules (via Kmeans)
- Concaténation des clusters de véhicule aux clients d'apprentissage via l'immatriculation, et apprentissage supervisé (via catboost) pour prédire le cluster le plus approprié pour un nouveau client et prendre une voiture appartenant à ce cluster.

Pour la première partie, concernant la clusterisation des véhicules, elle s'est déroulé en plusieurs étapes :

- Nettoyage de la donnée avec, par exemple, la conversion des données catégoriques en plusieurs colonnes binaires, l'utilisation d'un labelencoder pour la marque, la gestion des données manquantes.
- Analyse de la donnée via des graphiques (histplot, heatmap, distribution, etc.)
- Application d'une normalisation via StandardScaler
- Application d'une PCA pour réduire en 2 dimensions
- Application d'une clusterisation avec un Kmeans
- Exportation du résultat immatriculation et cluster correspondant dans un CSV



Pour la partie client et la prédiction de son futur véhicule, cela s'est déroulé tel quel :

- Nettoyage de la donnée d'entraînement avec, par exemple, la conversion des données catégoriques en plusieurs colonnes binaires, l'utilisation d'une partie de l'immatriculation pour le département, la gestion des données manquantes.
- Analyse de la donnée via des graphiques (histplot, heatmap, distribution, etc.)
- Application du même processus de nettoyage pour la donnée de test et vérification de la conformité des colonnes entre les deux datasets post-processing.
- Normalisation de la data avec StandardScaler
- Application d'un CatBoostClassifier (~10 min) utilisant le gradient boosting
- Récupération d'une immatriculation dans le pool du cluster prédit.

Il y a aussi un troisième notebook "*notebook_code_connect_Oracle_and_Hadoop.ipynb*" contenant le code utilisé pour retirer les fichiers depuis HDFS et les tables depuis Oracle.

5. Interprétations & réactions

On constate par exemple un réel regroupement post Kmeans de voiture ayant énormément de point commun, avec un groupe de voiture longue, puissante et cher, d'autre voiture plus abordables.

Pour l'instant le choix de la voiture n'est pas optimal étant donné un random dans le cluster, nous ne vérifions pas si la voiture est déjà utilisé ou non.

J'ai utilisé CatBoostClassifier pour explorer ce type d'algorithme ML après avoir constaté que la majorité des haut placés sur les concours de la plateforme Kaggle utilisaient le gradient boosting.

6. Résultats

	age	taux	nbEnfantsAcharge	2eme voiture	Femme	Homme	Célibataire	En Couple	Marié(e)	cluster	imma_choisi
0	21	1396.0	0	0	1	0	1	0	0	0	3642 VD 36
1	35	223.0	0	0	0	1	1	0	0	3	9696 XQ 62
2	48	401.0	0	0	0	1	1	0	0	0	5156 UT 65
3	26	420.0	3	1	1	0	0	1	0	1	5925 UE 11
4	80	530.0	3	0	0	1	0	1	0	2	4209 HL 10
5	27	153.0	2	0	1	0	0	1	0	3	369 RP 35
6	59	572.0	2	0	1	0	0	1	0	3	4024 QY 13
7	43	431.0	0	0	1	0	1	0	0	3	8286 ZL 28
8	64	559.0	0	0	0	1	1	0	0	0	2975 VB 13
9	22	154.0	1	0	0	1	0	1	0	3	3021 JH 58
10	79	981.0	2	0	1	0	0	1	0	3	9738 YU 24
11	55	588.0	0	0	0	1	1	0	0	3	7628 SP 24
12	19	212.0	0	0	1	0	1	0	0	3	2677 NZ 32
13	34	1112.0	0	0	1	0	0	1	0	4	8246 SQ 13
14	60	524.0	0	1	0	1	0	1	0	0	3679 NK 62
15	22	411.0	3	1	0	1	0	1	0	1	7958 LL 72
16	58	1192.0	0	0	0	1	0	1	0	4	6398 KR 37
17	54	452.0	3	1	1	0	0	1	0	1	1805 UP 63
18	35	589.0	0	0	0	1	1	0	0	3	1202 VT 15
19	59	748.0	0	1	0	1	0	1	0	0	595 EC 73

age,taux,nbEnfantsAcharge,2eme voiture,Femme,Homme,Célibataire,En Couple,Marié(e),cluster,imma_choisi
21,1396.0,0,0,1,0,1,0,0,0,3642 VD 36
35,223.0,0,0,0,1,1,0,0,3,9696 XQ 62
48,401.0,0,0,0,1,1,0,0,0,5156 UT 65
26,420.0,3,1,1,0,0,1,0,1,5925 UE 11
80,530.0,3,0,0,1,0,1,0,2,4209 HL 10
27,153.0,2,0,1,0,0,1,0,3,369 RP 35
59,572.0,2,0,1,0,0,1,0,3,4024 QY 13
43,431.0,0,0,1,0,1,0,0,3,8286 ZL 28
64,559.0,0,0,0,1,1,0,0,0,2975 VB 13
22,154.0,1,0,0,1,0,1,0,3,3021 JH 58
79,981.0,2,0,1,0,0,1,0,3,9738 YU 24
55,588.0,0,0,0,1,1,0,0,3,7628 SP 24
19,212.0,0,0,1,0,1,0,0,3,2677 NZ 32
34,1112.0,0,0,1,0,0,1,0,4,8246 SQ 13
60,524.0,0,1,0,1,0,1,0,0,3679 NK 62
22,411.0,3,1,0,1,0,1,0,1,7958 LL 72
58,1192.0,0,0,0,1,0,1,0,4,6398 KR 37
54,452.0,3,1,1,0,0,1,0,1,1805 UP 63
35,589.0,0,0,0,1,1,0,0,3,1202 VT 15
59,748.0,0,1,0,1,0,1,0,0,595 EC 73