Partie Appliquée

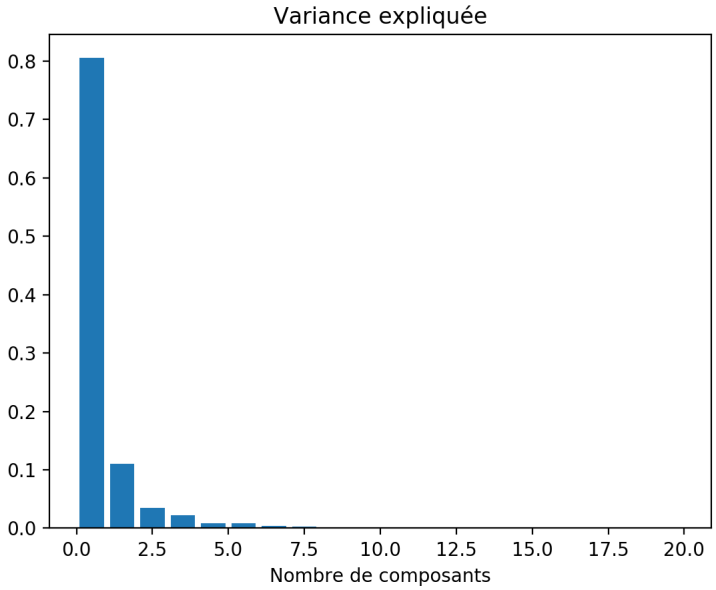
Dans cette section, nous allons tester et comparer les différents algorithmes de Kmeans, Kmeans++ et Kmeans++ semi-supervisées sur un jeu de données clients. Nos mesures de performances sont le coût tel qu'estimé par la fonction potentiel, ainsi que la durée d'exécution de l'algorithme.

La Base de Données

La base de données contient 60366 observations (clients) et 39 variables quantitatives et qualitatives. Tous les clients sont anonymisés se qui limitera l'interprétation de l'analyse.
Nous commençons par analyser la base de données. Ci-dessous, nous affichons la statistique d'une partie de nos données.

Index	g_distance_to_sho	n_distance_to_sho	x_distance_to_sho	roducts_purchase	e_products_purch	mount_purchased	avg_purchase	avg_price	shops_used	distance_shop_1	distance_shop_2	distance_shop_3	distance_shop_4	distance_shop_5	cts_purchased_sl
count	60366	60366	60366	60366	60366	60366	60366	60366	60366	60366	60366	60366	60366	60366	60366
mean	2030.23	1396.43	2942.67	1778.7	330.668	4235.46	8.53208	3.66651	2.38005	2496.63	2488.24	1924.97	2882.8	2020.9	887.813
std	1119.9	1048.41	1327.52	2185.04	236.002	5006.74	10.3154	9.13317	1.01218	1281.53	1417.36	1157.75	1730.09	1260.85	1438.48
min	6.64247	6.64247	6.64247	1	1	0.212	0.212	0.212	1	93.2834	11.1904	17.8443	6.64247	25.4607	0
25%	1250.24	573.611	2117.22	227	127	653.118	4.53291	2.53375	2	1609.58	1501.9	1253.84	1532.68	1173.41	49
50%	1936.88	1184.94	2869.2	925	304	2355.14	6.86278	2.94426	2	2288.67	2355.4	1746.17	2704.32	1852.93	292
75%	2569.71	1962.38	3580.67	2551.75	500	6054.46	10.4295	3.55811	3	3144.06	3370.72	2323.56	4056	2466.56	1072
max	9004.16	9004.16	9267.7	22131	1465	51588.7	787.569	787.569	5	8019.92	9004.16	7395.25	9273.69	7465.81	17016

Nous voyons qu'il existe une seule variable qualitative, la variable 'shops used', les autres étant toutes des variables quantitatives continues. Nous remarquons aussi que toutes les variables quantitatives affichent de grandes dispersions. Par exemple, en regardant le nombre de produits achetés, nous voyons que certains clients achètent très peu, soit 1 seul article, tandis que d'autres achètent jusqu'à 22 131 articles, la moyenne étant 2552.75. Ce type d'observation nous permettra de mieux interpréter les clusters proposés par les différentes versions des k-means.
Une Analyse en Composantes Principales est implémentée afin de mieux regrouper et comprendre les variables quantitatives. Le choix du nombre de composantes principales est basé sur le pourcentage de la variance expliquée ainsi que par la méthode du coude (une baisse soudaine de la variance expliquée entre deux composantes).



Nous voyons ici que les premières composantes principales expliquent une bonne partie de l'inertie, soit plus de 80% de la variance, et ainsi nous nous baserons uniquement sur l'analyse en composantes principales sur 2 et 3 dimensions.
Sur un plan 1-2 nous obtenons le graphique tel qu'affiché ci-dessous:

AFFICHER ACP SUR LE PLAN 1-2 (DIMENSION 2)
AFFICHER ACP SUR 3 DIMENSIONS

Choix du nombre de clusters

La méthode K-means est une méthode d'apprentissage non-supervisée. Lors de son application les données sont séparées en plusieurs classes prédéterminés de façon que les individus ayant le plus de similarité. C'est ainsi qu'une des tâches clefs est de trouver le nombre approprié de classes, k. Il existe plusieurs techniques pour déterminer le nombre de classes. Nous discuterons que des cas les plus connus :

1. Méthode du pouce:

   Cette méthode est une méthode approximative où le nombre de classes, k est déterminé par : $k \approx \sqrt{n/2}$

2. L'indice de qualité:

   Afin d'évaluer la qualité de la classification, les indices inertiels , soient l'inertie intra-classes et l'inertie l'inter-classes sont souvent utilisés. L'inertie intra-classes 'mesure le degré d'homogénéité entre les objets appartenant à la même classe' tandis que l'inertie inter-classes 'mesure le degré d'hétérogénéité entre les classes.' Il existe plusieurs indices de qualité, par exemple l'indice de Dunn, l'indice de Calinski et Harabasz (CH) ou encore l'indice de Silhouette. Le premier calcule la distance minimale inter-classes et ainsi plus cette distance est grande, meilleur sera la classification.

   Introduit par Kauffman et Rousseew, l'indice de Silhouette nous donne une représentation visuelle de la distance entre un point d'une classe avec les points des classes voisines. Plus le coefficient ainsi calculé est proche de 1, plus la distance avec les classes voisines (inertie inter) est grande. Ceci représente le nombre de classes optimale. À l'inverse, un coefficient proche de -1 nous indique une mauvaise classification de l'observation.

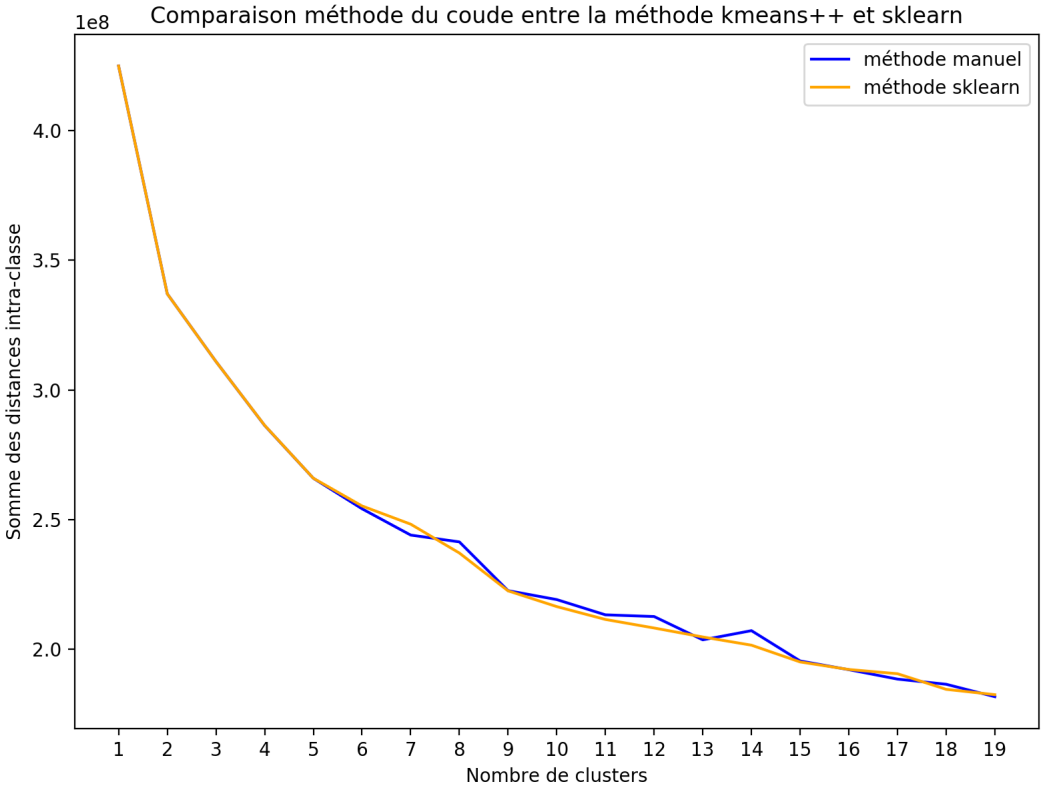
3. Méthode du coude:

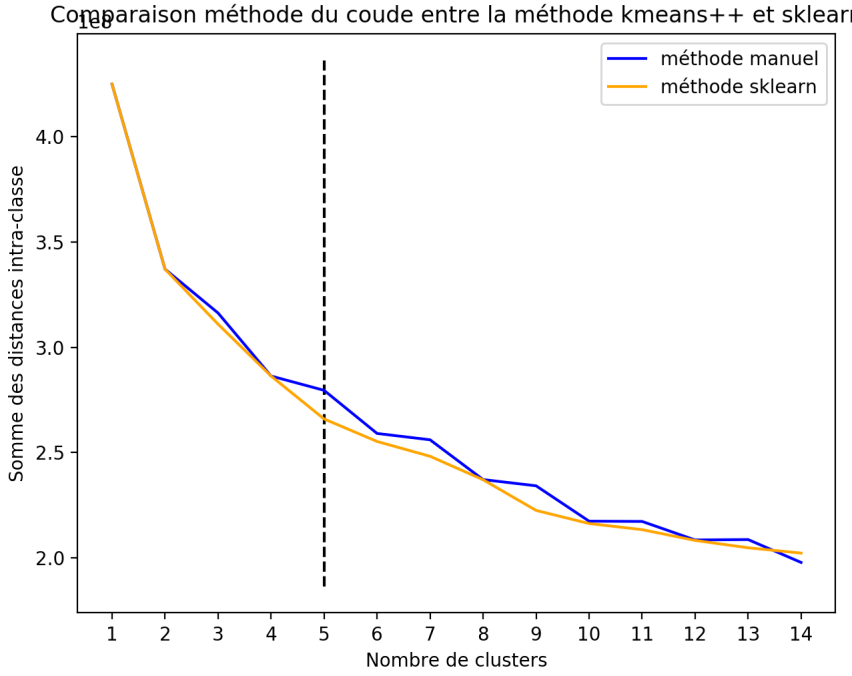
   La méthode du coude est une technique visuelle très connue. L'idée derrière cette technique est d'implémenter la méthode K-means en parcourant k valeurs. À chacune des k valeurs, la somme des erreurs au carré est calculée et est affiché sur un graphique, nous permettant à mieux visualiser les résultats. L'objectif est de choisir la valeur k (qui sera le nombre de classes) créant un effet de 'coude', c'est-à-dire provoquant une baisse plus conséquente, plus soudaine de la somme des erreurs au carré. Nous disons ceci en gardant en tête que la somme des erreurs aura toujours tendance à baisser, plus la valeur de k est grande.

4. La validation croisée:

   La validation croisée regarde la stabilité des classes. Les données sont séparées en au moins deux parties. La première est utilisée pour former les classes tandis que la deuxième sert de validation. Lorsque nous parlons de stabilité, nous parlons de la fréquence à laquelle des classes similaires sont formées lorsque plusieurs itérations sont effectuées. Ainsi une plus grosse fréquence de l'apparition de mêmes classes équivaut à une plus grosse stabilité de ces classes.

Dans le cadre de notre étude, nous choisissons de travailler avec la méthode du coude, une méthode que nous appliquons sur la méthode de kmeans++. Cette méthode est d'ailleurs comparée avec celle de sklearn afin de déterminer l'exactitude de l'algorithme utilisé. Le graphique ci-dessous affiche la répartition des sommes des erreurs au carré à la fois pour notre méthode, dite la méthode manuelle, ainsi que la méthode proposé par sklearn. Nous voyons une baisse plus soudaine lorsque nous avons 5 clusters. Ce sera ainsi le choix du nombre de clusters utilisé lors de l'application des algorithmes de clustering.





Observations et Résultats

Dans cette section nous cherchons à présenter les résultats obtenus par les algorithmes kmeans, kmeans++ et kmeans++ semi supervisées. De plus ce dernier a été analysé plus en détails afin d'afficher l'impact des différents pourcentages de données labellisés. Comme pour l'ACP la variable qualitative a été écartée de notre analyse. 6 essaies ont été effectués afin d'évaluer le temps d'exécution des algorithmes (en secondes). La moyenne des distances intra-classes a été obtenu en se basant sur 100 simulations. Le nombre de clusters, déterminé en utilisant la méthode du coude, est 5.
Les tableaux ci-dessous affichent les résultats obtenus pour les différents algorithmes.

Table 1: **Résultats en utilisant kmeans**

Temps d'exécution Minimum	Temps d'exécution Maximum	Temps d'exécution Moyen	Moyenne des distances intra-classe
0.867459	4.278867	1.867284	267024131.97

Table 2: **Résultats en utilisant kmeans++**

Temps d'exécution Minimum	Temps d'exécution Maximum	Temps d'exécution Moyen	Moyenne des distances intra-classe
0.952380	1.963426	1.603216	269311160.94

Table 3: **Résultats en utilisant kmeans++ semi-supervisées avec 60% de données labellisées**

Temps d'exécution Minimum	Temps d'exécution Maximum	Temps d'exécution Moyen	Moyenne des distances intra-classe
0.386259	0.616933	0.481936	265971783.28

Tel que prévu la durée d'exécution moyenne de l'algorithme de kmeans est plus longue que celle du kmeans++ et du kmeans++ semi-supervisée (avec 60% de données labellisées). Cette dernière est d'ailleurs en moyenne ¿ 3 fois plus rapide que le kmeans ou le kmeans++. Au niveau du coût de la fonction nous voyons que le kmeans ++ semi supervisée affiche à nouveau le meilleur résultat (avec la plus petite distance moyenne intra-classes sur 100 simulations). Il est toutefois étonnant de voir que la distance intra-classe moyenne lorsque kmeans est utilisé est plus petite que celle du kmeans++.

AFFICHER LE COUT DE LA FONCTION POUR N SIMULATIONS sous form de PLOT AVEC INTERVALLES DE CONFIANCE

1.
2.
3.
4.