Data Analytics

Implémentation de K-Means Exploration d'annonces

DJILANI Amira et OUAZZANI CHAHDI Nizar

Table des matières

1 Résultats des expériences KMeans

Dans cette section nous allons présenter les résultats obtenus en effectuant des expériences sur notre implémentation de KMeans sur PySpark. Notre code a été développé sur le service Google Colab.

1.1 Test sur les données de la base Iris

Dans cette expérience nous avons comparé la distribution de la distance intra-cluster donnée après 4 itérations par l'algorithme KMeans que nous avons implémenté (à droite) et celui déjà présent dans la librairie mllib (à gauche).

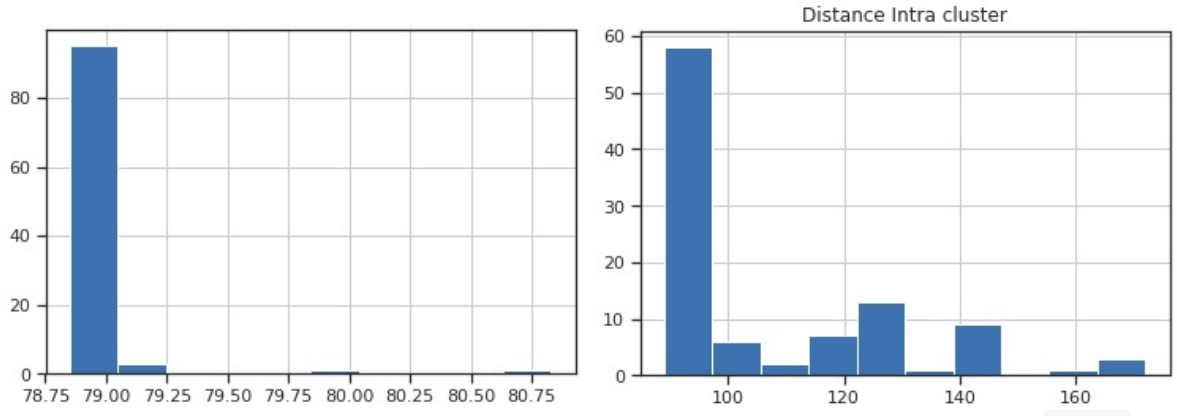

FIGURE 1 – Comparaison des résultats des deux implémentations

Pour 4 itérations, notre implémentation prend environ 34 secondes en moyenne pour s'exécuter alors que celle de mllib prends quelques millisecondes. En sortie, les distances intra-cluster de notre algorithme sont plus dispersées avec une forte tendance à converger vers 80 alors que l'implémentation de mllib converge toujours vers 80.

1.2 Test sur des données générées aléatoirement

Dans cette partie nous avons utilisé le générateur de données aléatoires que nous avons implémenté pour générer 120 points de données artificiellement avec un nombre de clusters différents à chaque fois qui suivent des lois normales avec des caractéristiques d'espérance et d'écart type différents pour tester la robustesse de notre algorithme. Nous avons également choisi de créer des données dans le plan (dimension 2) afin de faciliter la visualisation des résultats.

1.2.1 4 Clusters facilement séparables

Dans ce test nous allons vérifier que notre algorithme arrive à bien séparer 4 clusters bien distincts (ie) générer à partir de lois normales avec des espérances loin l'une de l'autre dans le plan.
On prend pour vecteur d'espérance [[**1 1**], [**100 100**], [**250 250**], [**500 500**]] en prenant un ecart type de [**5 5**].

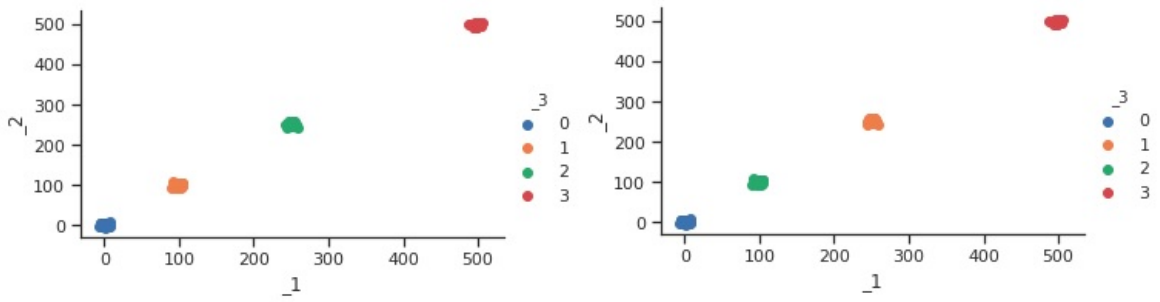

FIGURE 2 – Affectation réelle des clusters vs clustering KMeans

On remarque qu'après 4 itérations l'algorithme réussi à faire une séparation exacte.

1.3 4 Clusters proches

Dans ce test nous allons rapprocher les clusters de l'expérience précédente. On prend le vecteur des espérances suivant : [[**1 1**], [**15 15**], [**30 30**], [**60 60**]] tout en gardant le même vecteur d'écart type.

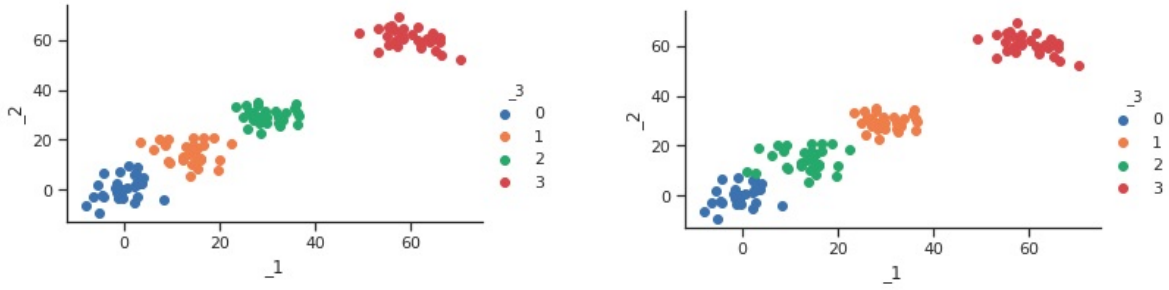

FIGURE 3 – Affectation réelle des clusters vs clustering KMeans

On voit qu'après 4 itérations l'algorithme réussi à faire le clustering tout en se trompant sur quelques points.

1.3.1 4 Clusters chauvechés

Dans cette expérience on garde les mêmes vecteurs d'espérance en modifiant les vecteur d'écart type, on considère maintenant le vecteur suivant : [[**5,5**],[**5,5**],[**5,5**],[**15,15**]].
Cela nous permettra de générer des points plus dispersés autour du 4ème cluster.

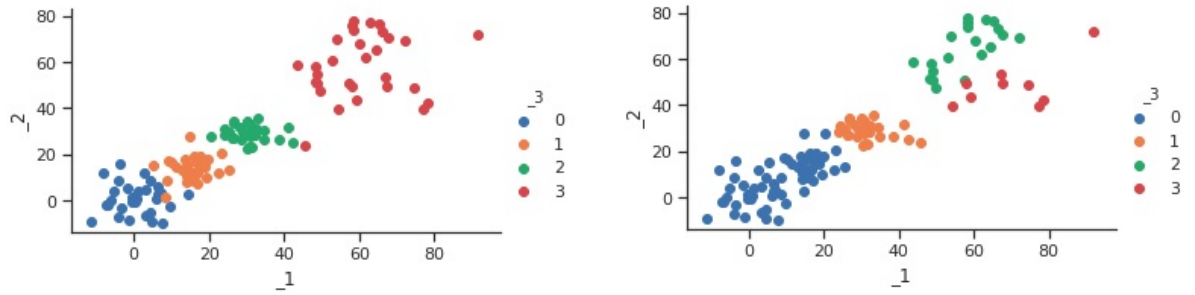

FIGURE 4 – Affectation réelle des clusters vs clustering KMeans

On remarque que dans ce cas, l'algorithme ne réussi pas à retrouver les clusters initiaux.

1.3.2 2 Clusters disproportionnés

Dans cette expérience, nous générons 2 clusters à partir des lois normales suivantes : **N([25,25],[5,5])** et **N([100,100],[20,20])**.
Ces données n'étant pas chevauchées, l'algorithme réussi à bien les séparer.

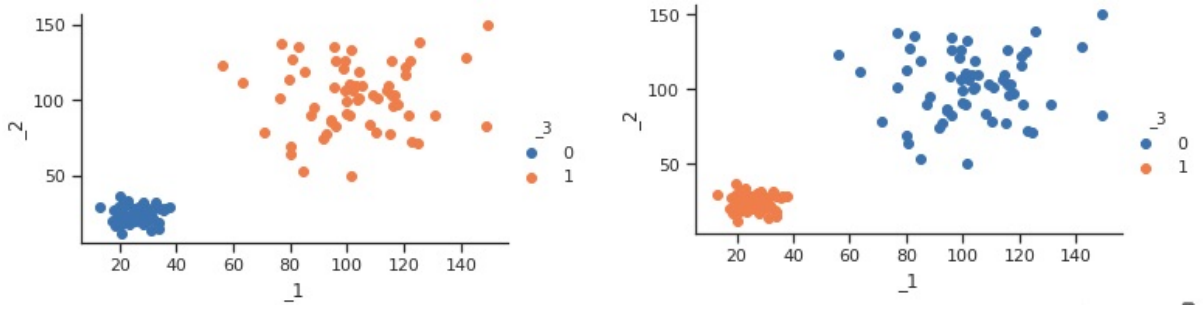

FIGURE 5 – Affectation réelle des clusters vs clustering KMeans

1.3.3 2 Clusters disproportionnées et chevauchés

Dans cette expérience nous rapprochons les deux clusters précedents ([[**25,25**],[**70,70**]]).
On voit que là aussi l'algorithme ne réussit pas à retrouver les clusters d'origine.

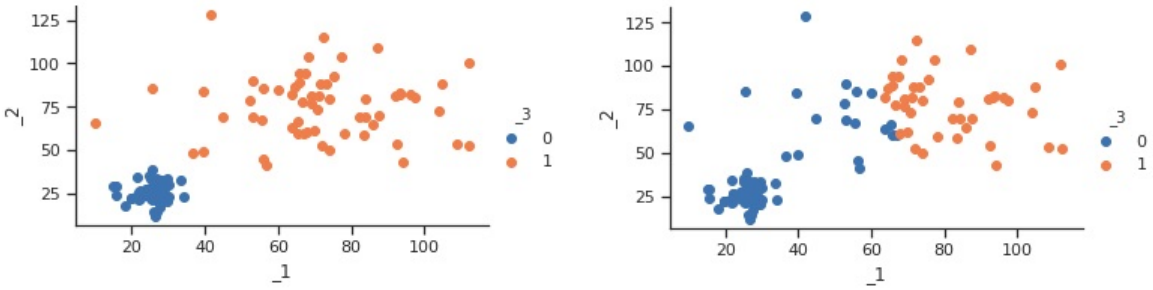

FIGURE 6 – Affectation réelle des clusters vs clustering KMeans

1.3.4 5 Clusters proches

Dans cette expériences nous créons 5 clusters à partir des vecteur d'espérance et écart type suivants :
mu = [[10,1],[10,100],[100,10],[50,50],[100,100]]
std = [[10,10],[10,10],[10,10],[25,25],[10,10]]

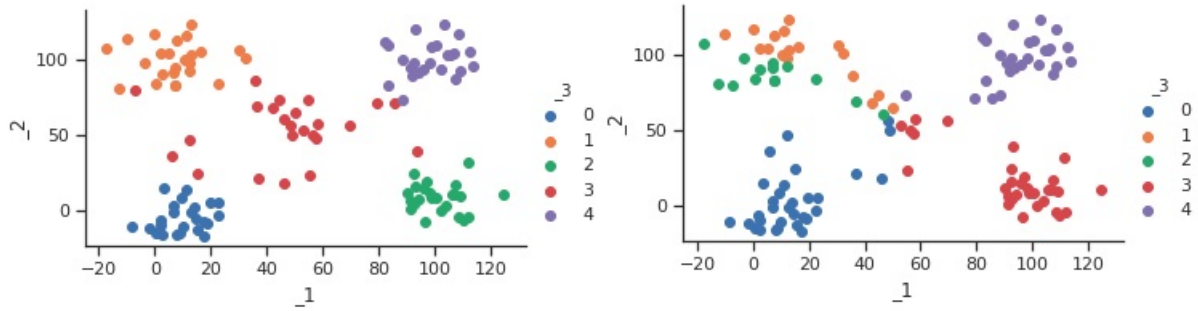

FIGURE 7 – Affectation réelle des clusters vs clustering KMeans

On peut voir que dès que les clusters sont proches l'un de l'autre l'algorithme trouve du mal à retrouver les données comme elles ont été séparées initialement.

2 Exploration et analyse d'une base d'annonces immobilières

Dans cet exercice nous avons choisi d'analyser des données relatives au marché de l'immobilier en Île de France.

2.1 Base de données utilisées

Les données que nous avons utilisé ont été recueillis sur le site **leboncoin**. Il s'agit de 800 annonces de vente de biens immobiliers dans tous les départements d'île de France. La base rassemble les variables suivantes : :

```
{"nb_photos": "nombre de photos dans l'annonce",
 "titre_annonce": "titre accompagnant l'annonce",
 "commune": "commune du bien",
 "code_postal_commune": "code postal de la commune",
 "code_postal_departement": "code postal du d partement",
 "prix": "prix du bien",
 "date": "date de publication de l'annonce",
 "lien": "lien de l'annonce",
 "details_annonce": "consommation, annonceur pro ou part...",
 "description": "corps de l'annonce",
 "type_bien": "maison, appartement, terrain..",
```

```
"pieces": "nombre de pieces du bien",
"surface": "surface du bien",
"surface_int": "contenu du corpus 1",
"prix_m ": "prix au metre carr  (variable calculee)",}
```

Afin de collecter ces données nous avons écrit un script python utilisant des librairies de web scrapping, notamment **selenium**.

2.2 Exploration des données

2.2.1 Histogramme des surfaces des biens

On remarque qu'il y a plus d'annonces dans le premier quartile (des biens avec des surfaces allons de 0 à 100), il s'agit de studios et d'appartements. Dans le dernier quartile on retrouve les terrains.

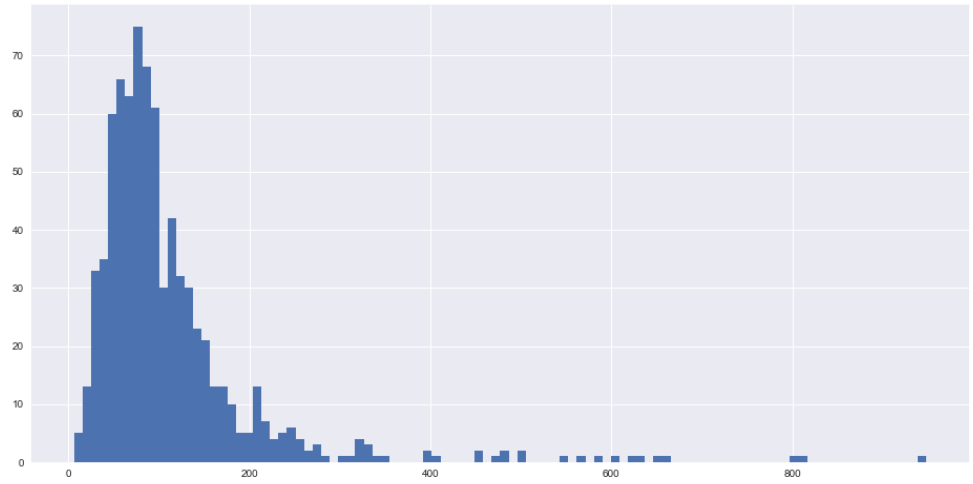

FIGURE 8 – Histogramme de la distribution des surfaces des biens

2.2.2 Prix au m en fonction de la surface

On affiche le nuage de points des prix au m en fonction de la surface et du type du bien ((ie) Parking Appartement Maison). On peut d'or et déjà émettre des hypothèses sur la localisation des biens à partir de ce nuage. Par exemple, on peut supposer que plus on s'éloigne du centre de Paris plus le prix au m diminue.

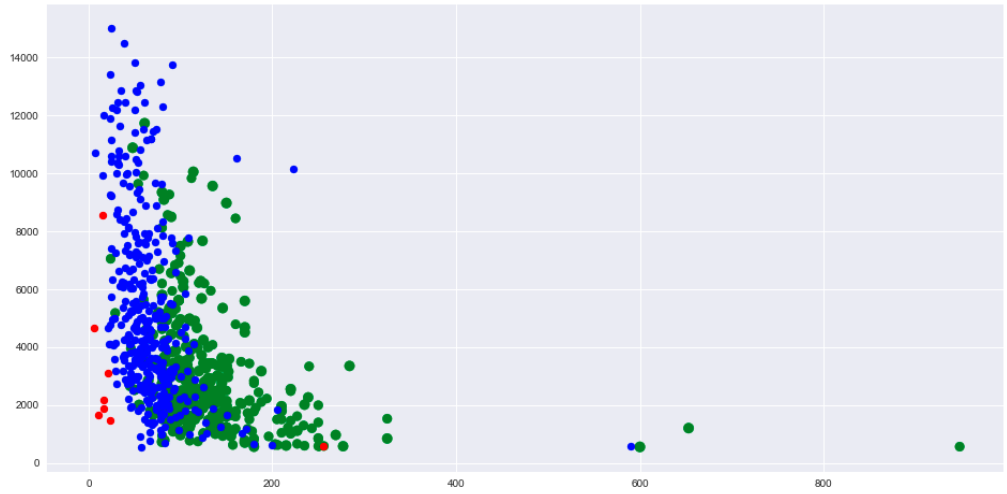

FIGURE 9 – Nuage des points du prix au m en fonction des surface

On peut également remarquer qu'il y a une tendance indiquant que plus il y a de m dans le bien plus le prix au m diminue, cette tendance n'est pas vérifiée dans la partie gauche du nuage.

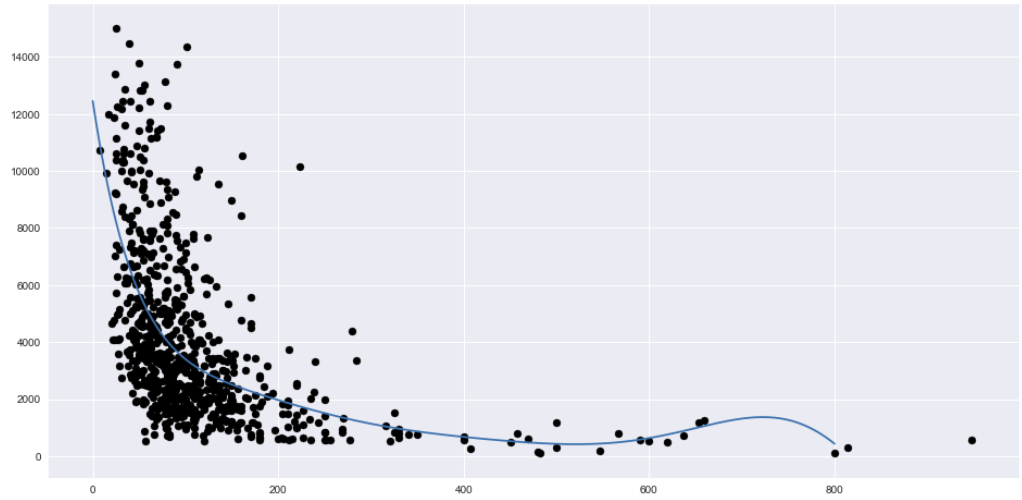

FIGURE 10 – Courbe de tendance des prix

Cette courbe montre un polynôme de degré 8 entraîné sur les données.

2.3 Jointure avec des données géographiques

Dans cette partie, nous allons nous intéresser aux mêmes données que nous avons utilisées précédemment tout en ajoutant la dimensions géographique.
Pour cela, nous avons réalisé la jointure entre les codes postaux des communes et des départements des annonces et des données géographiques open data de France.

2.3.1 Prix au m par département

On remarque que le m atteint son maximum sur Paris intra-muros et sa couronne et diminue en s'y éloignant. Ces données étant agrégées au niveau départemental ne permettent pas de dégager beaucoup d'informations utiles.

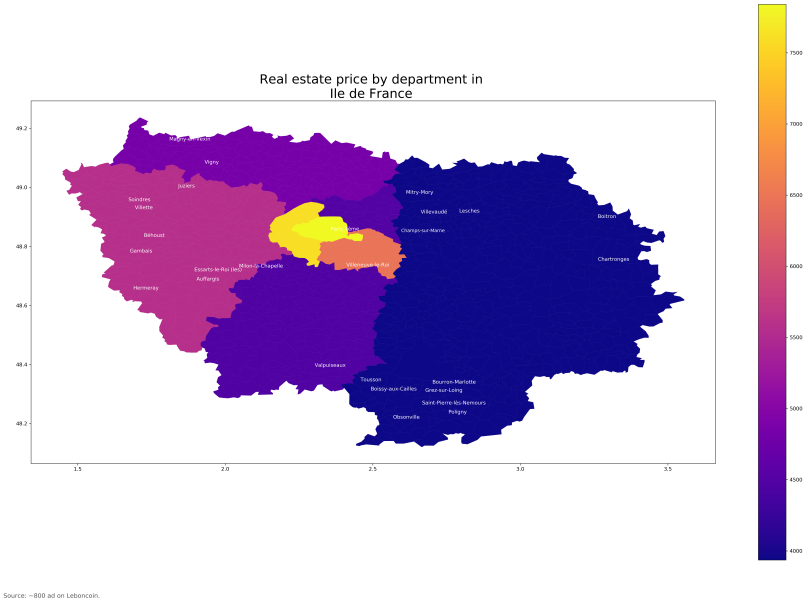

FIGURE 11 – Prix au m par département

2.3.2 Prix au m par commune

Nous avons tenté de descendre au niveau communal, mais faute d'annonce nous n'avons pas pu avoir des informations pour toutes les communes.

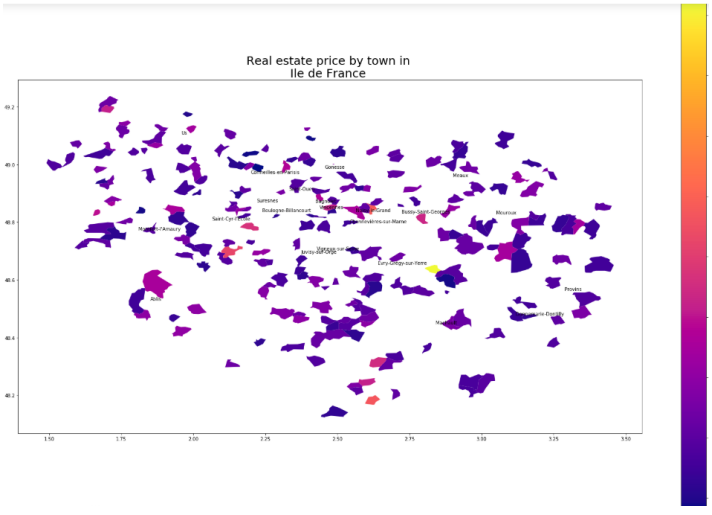

FIGURE 12 – Prix au m par commune

Pour réaliser le graphe précédent nous avons utilisé la librairie **geopandas**

2.4 Latent Semantic Analysis des description

Dans cette section nous allons tentons d'extraire des concepts associant des termes similaires conceptuellement en effectuant une analyse sémantique latente sur les descriptions associées à chacune des annonces.

Résultats :

Concept : 0, 5 et 8 Composants du bien
m, maison, salle, hors, prix, wc, honoraires, chambres, appartement, étage
composé rdc, composé rdc entrée, desservant pièce vivre, entrée desservant pièce, espace cuisine grande, handicapés wc, logements individuels, m espace, m espace cuisine, normes handicapés wc
euros, volets, rer proche, double vitrage, vitrage, f4 comprenant, f4 comprenant entrée, type f4 comprenant, type f4, 157 lots

Concept : 1 Energie et consommation
air, hors, basse consommation rt, box domotique, box domotique, gestionnaire, consommation rt, consommation rt 2012, domotique, gestionnaire, domotique gestionnaire énergie, détecteurs fumée

Concept : 2, 3 et 4 Prix et agence
prix, hors honoraires, nangis, hors, honoraires, bourillon, century21, ttc, prix hors, bien hors
nangis, bourillon, century21, copropriété, agence bourillon, bourillon nangis, lots, annuelles, charges, charge vendeur propos
charge vendeur, vendeur, honoraires charge vendeur, honoraires charge, charge, maison, sol total, honoraires, référence, référence annonce

Commentaire
Tout d'abord, à partir de cet exercie, on peut voir la force de LSA qui permet d'extraire un volume important d'informations sans avoir à faire des analyses individuelles des annonces.
On peut voir par exemple ce que les annonceurs affichent le plus quand il s'agit de la consommation d'énergie de leurs biens ou de ses composants.
On peut imaginer que cette analyse aurait pu nous fournir plus d'informations si nous avions beaucoup plus d'annonces, dans ce cas il aurait été intéressant de faire des segmentations par ville ou département pour voir les caractéristiques qui seront remontées dans chaque ville selon la façon dans les annonceurs présentent leurs biens dans la ville en question.