Description du projet

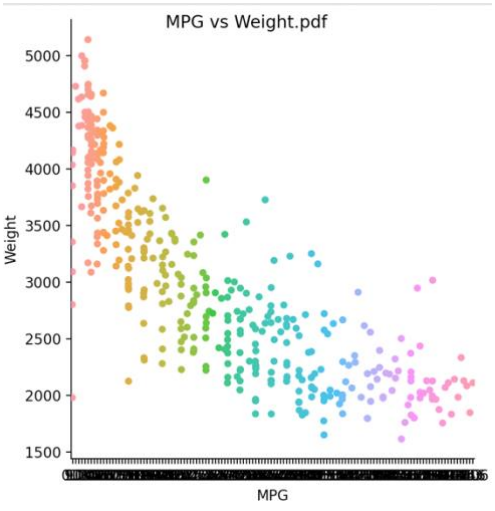
Algo2 – Jérémie Dumez/Paul Raphanaud

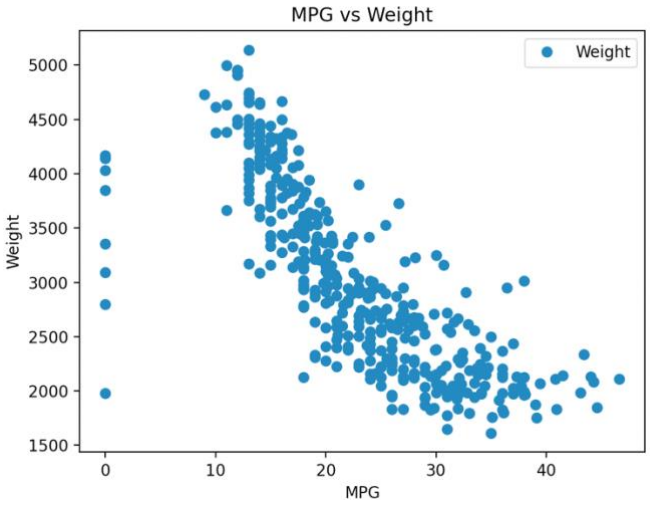
Pour ce projet nous avons utilisé un dataset de voiture. Le dataset comportait différente indication :

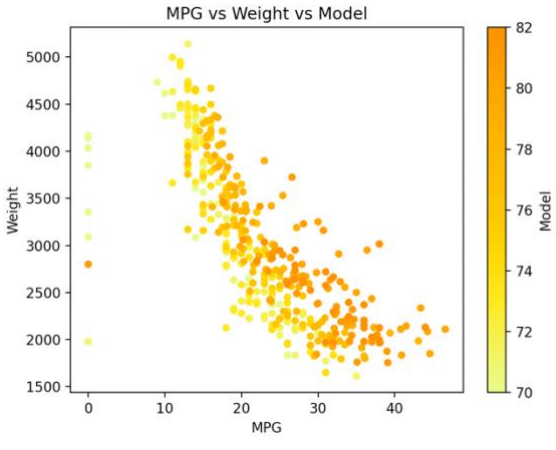
- Car
- MPG
- Cylinders
- Displacement
- Horsepower
- Weight
- Acceleration
- Model
- Origin

Nous nous sommes focalisés sur les MPG (Miles per Galon) et le poids (Weight)

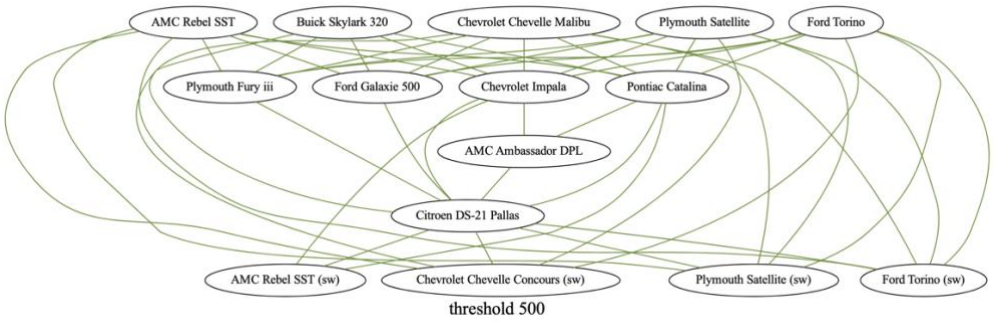
Comment la consommation d'essence évolue en fonction du poids d'un véhicule ?







Pour le processing, dans le fichier build_metrics.py. Nous avons fait un graph de similarité en fonction de leurs poids et de leurs consommations mais nous avons rajouté l'origine du véhicule dans le calcul de la distance euclidienne. Les valeurs de dissimilarité varient entre 0 et 1200 environ. Nous avons donc mis un seuil à 500.



threshold 500

Ensuite nous avons décidé de faire une clusterisation des mêmes valeurs avec notre dataset. Nous avons donc implémenté un algorithme K-means.

L'algorithme de K-means place un nombre de centroide. Ensuite il calcul la distance entre la data et les centroids et assigne au centroide le plus proche.

Pour un nombre d'itération définit, nous déplaçons les centroids pour obtenir le résultat le plus précis.

Ci-dessous un exemple avec 3 centroids créant 3 clusters distinct. On peut remarquer qu'à chaque itération les centroids se déplacent et forment de nouveaux clusters plus précis.

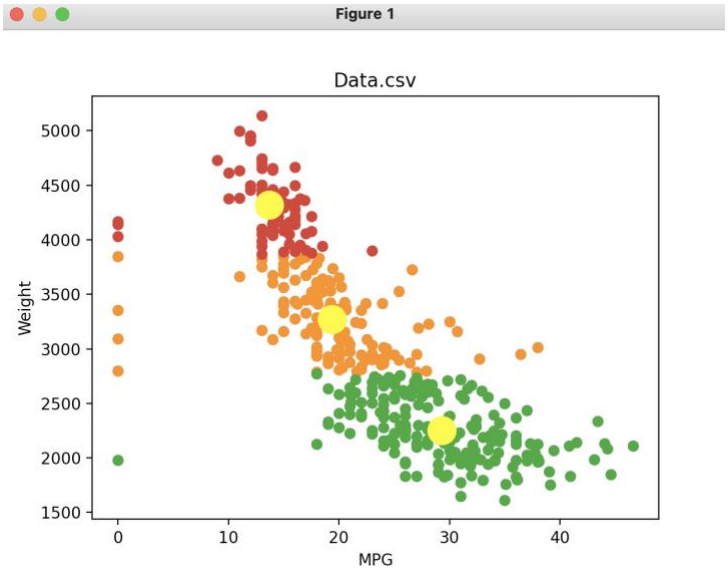
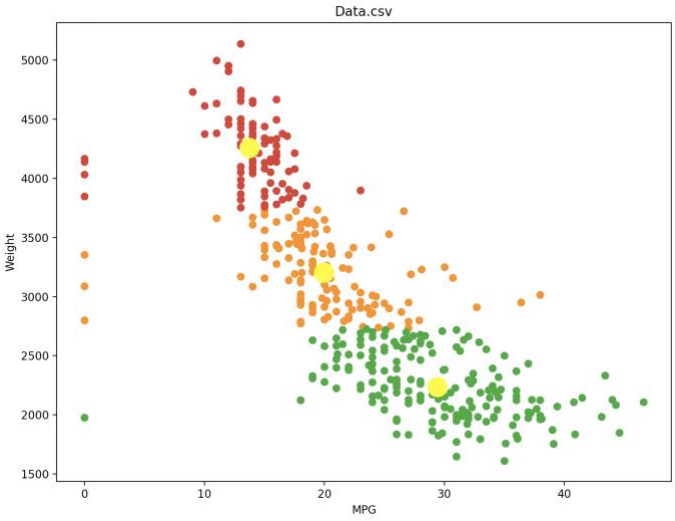
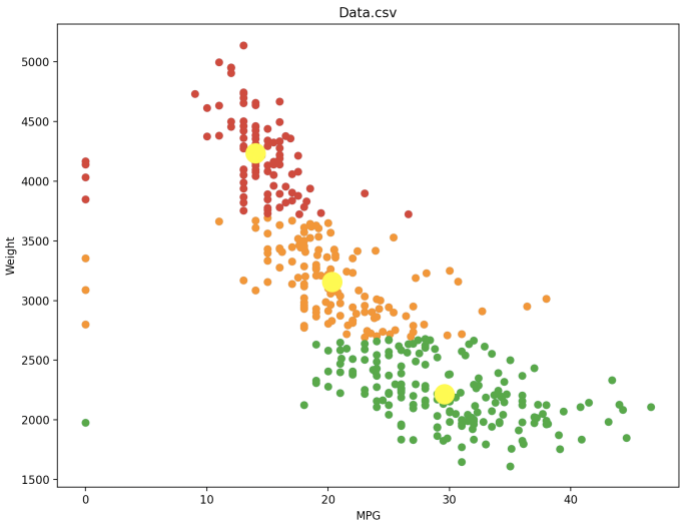


Figure 1
Data.csv
Weight
MPG
5000
4500
4000
3500
3000
2500
2000
1500
0
10
20
30
40


Data.csv
Weight
MPG
5000
4500
4000
3500
3000
2500
2000
1500
0
10
20
30
40


Data.csv
Weight
MPG
5000
4500
4000
3500
3000
2500
2000
1500
0
10
20
30
40

Pour définir le nombre de centroide nous avons mis en place 2 methodes :

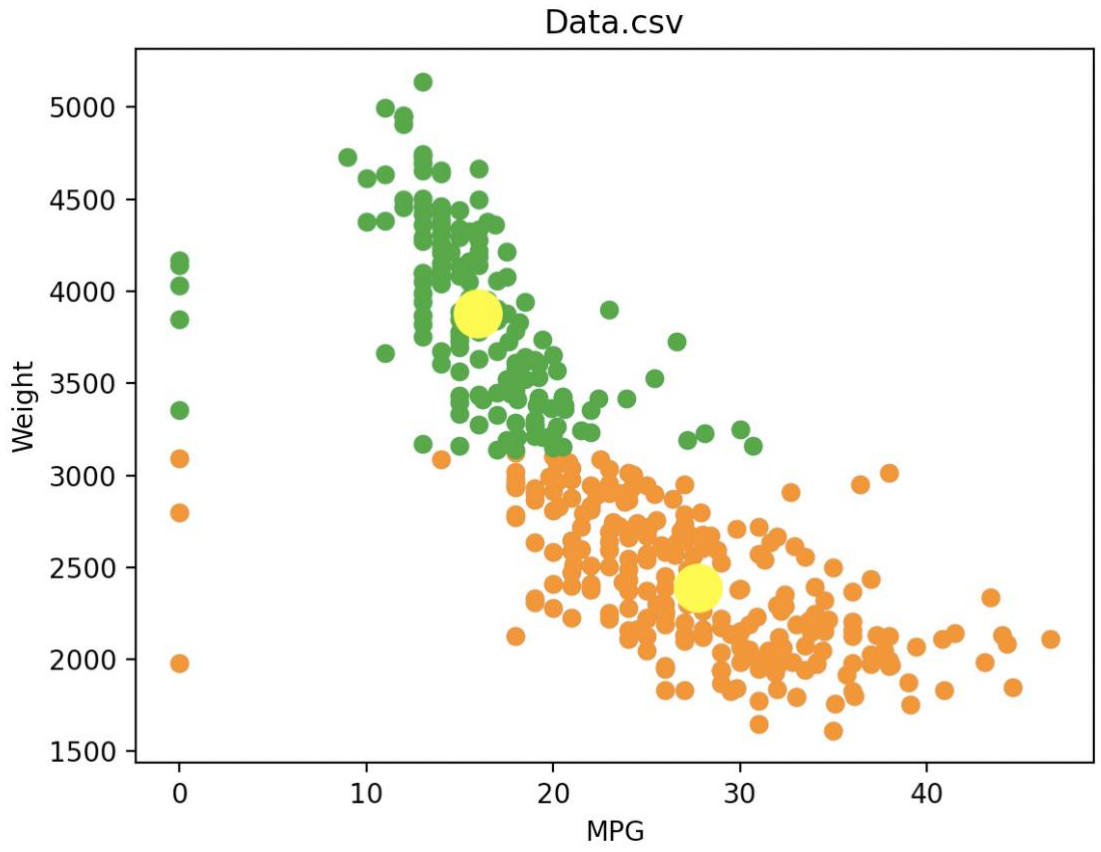
- L'elbow method
- Silhouette Analysis

Pour l'« elbow method », il faut analyser la courbe afficher et regarder lorsque la courbe change de direction à la manière d'un coude.

Pour la « Silhouette Analysis », on calcule un coefficient de silhouette et lorsque le score et le plus élevé : il s'agit du nombre de cluster.

Les 2 méthodes nous ont retourné le même résultat : 2 centroids.

Nous avons choisis le K-means car il était rapide d'exécution et simple de compréhension.



On remarque après cette clusterisation, que les 2 clusters confirment que plus le poids du véhicule est élevé plus la consommation est haute.