UNSUPERVISED LEARNING IN CAR FUEL EMISSIONS ANALYSIS

Jiawei Xu, Iván López Anca
Statistical Learning.
Bachelor in Data Science and Engineering.

Access to our raw dataset. (https://www.kaggle.com/datasets/mohamedsl0960/car-fuel-and-emissions-2000-2013/data)

Iván López Anca,
Jiawei Xu.

MOTIVATION

Is it really true that older cars contaminate more? Do recent cars have more modern equipment, or are old cars better? Is there a relation between the price range of the cars and their emissions? In this case study, we will tackle these issues.

DATA CLEANING

Removal, Mixture and Transformation of Variables

The data cleaning process involved removing irrelevant features to our analysis like "file", "description", and imperial consumption metrics, as well as eliminating duplicate rows. Variables with over 50% missing data were also dropped, while empty string values in categorical variables were corrected to NA.

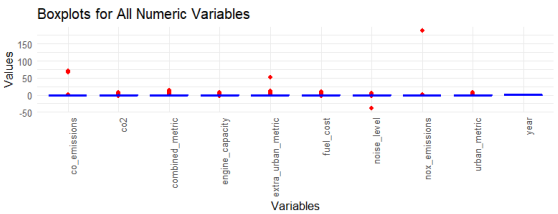
Then, "fuel_cost_12000_miles" and "fuel_cost_6000_miles" were combined into a single "fuel_cost" variable due to the sequence of appearance in the data.

Finally, we transformed some numerical and categorical variables into factors, optimizing the dataset for further analysis.

Outliers

We had to check if there were outliers in our dataset, and if so, to see if they are genuine, factor to keep them or remove them in case they are errors.

To do so we started getting a subset with the numeric values, the only ones that can possibly have outliers. Then, we applied the 3-sigma rule to count how many there were. After seeing that there were many we made boxplots to identify them

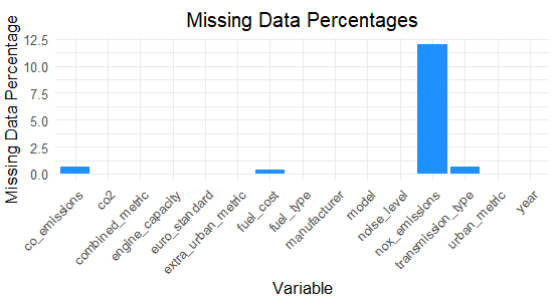


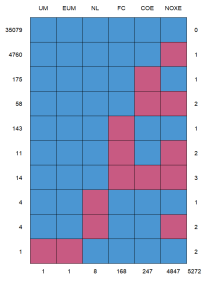
After this, we just removed the outliers that didn't make sense, such as nox emissions of 237000 when the rest are barely above 700, or negative co_emissions, which is impossible. We kept the majority of the outliers since they seem normal.

Missing Values

To end with our data preprocessing, we dealt with NA values firstly at recognizing them in the graph of the left. Then we proceed to remove "transmission_type" observations with NA's to avoid a biased analysis, since we can't guess this value.

Then the graph of the right was analyzed to decide how to proceed with the remaining variables. Variables with very few missing

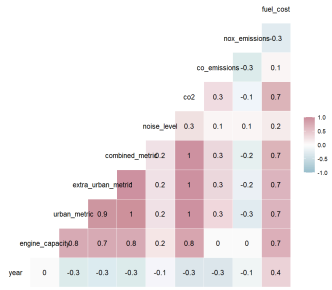




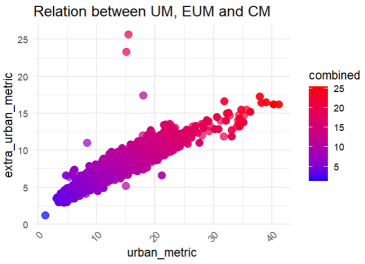
values, such as "noise_level", "co_emissions", and "fuel_cost", were excluded from the dataset. For variables like "urban_metric" and "extra_urban_metric", missing values were filled using "combined_metric". Finally, "nox_emissions" was imputed using multiple imputation with the MICE package, and a final check confirmed no remaining NAs.

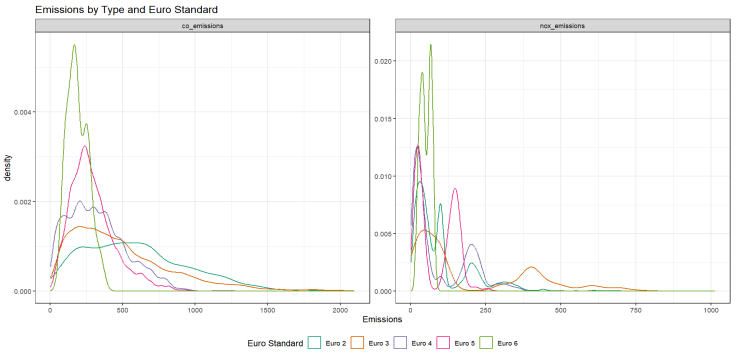
VISUALIZATION TOOLS

With the data cleaned, we decided to use our knowledge about graphics to briefly interpret the relationship between the variables. to understand our dataset.



With this graph we dealt with the correlations between the variables, having as a result that most of the variables are extremely correlated between them, like the metrics. For which we plotted this graph to show it



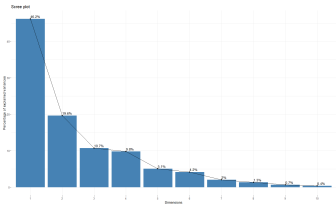


We also were surprised to see that the co and the nox emissions had nothing to do between them so we wanted to check if they were somehow related and we took the euro_standard variable to see it, proving that even if they aren't correlated they seem to share somehow any relationship between them.

PRINCIPAL COMPONENT ANALYSIS (PCA)

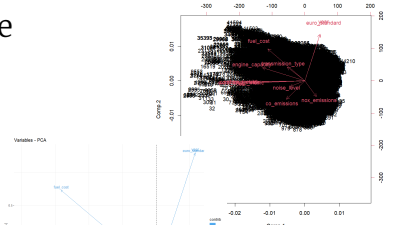
Having analyzed the correlation between the variables with the visualization tools, we could then reduce dimensionality with PCA.

First we converted euro_standard and transmission_type to numerical variables and we removed manufacturer and fuel_type since they are categorical variables. We also scaled the data to perform PCA, though this may lead to loss of variability.

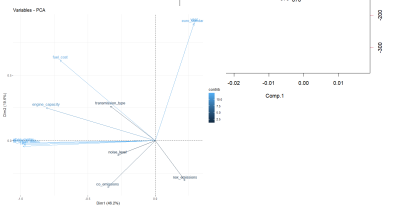


After performing the PCA, we got as result the following graph, where we can see how with 2 variables we can explain the 65.8% of the variability, while with 3 we can already explain the 76.4%.

Then, analyzing the loadings and contributions of each variable, we reached the conclusion that PC1 is the environmental-friendliness of the car, PC2 its modernity and PC3 the efficiency of the engine.

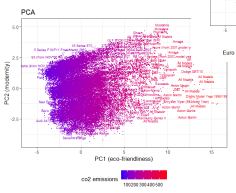


Then, since PC3 kinda overlaps with PC1 and PC2, we performed a biplot with PC1 and PC2, but we couldn't really see anything.



Therefore, we made another biplot using contributions instead of loadings, and we could see the influence and importance of each variable more clearly, finding for example the positive correlation between euro standard an year, and how much is their effect in both, PC1 and PC2. We can also see how the majority of the variables have negative impacts on these PCs.

To end with the PCA, we elaborated more graphs with the scores to answer some questions, such as the relation between PC1 and PC2 and euro standard, which had a high effect on both PCs.



As we can see, using Euro Standard as color, we get clear clustering of points, which highlight its effectiveness to measure modernity. We can also see how the graph goes upward, which makes sense since the more modern a car is, the more eco friendly it should be. We also made other graphs using co2 as the color, and we could see how it affects PC1.

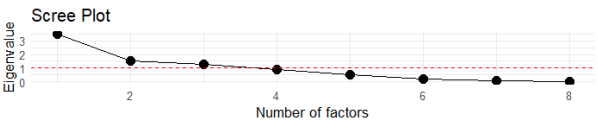
Finally, using these scores we could answer matters such as which manufacturers have the most environmentally friendly cars or the most efficient engines.

FACTOR ANALYSIS (FA)

Having determined the correlations between the variables we can do a Factor Analysis of them to identify underlying relationships by grouping them as factors.

For this we'll use just the numeric observations of our data to improve efficiency and we will eliminate those highly correlated variables (urban_metric and extra_urban_metric) to do the analysis because otherwise, errors could rise.

Then, we have to get the optimal number of factors given our data, which is done by calculating the eigenvalues of our correlation matrix. The number of factors will typically be the number of eigenvalues bigger than 1, but it's done by identifying that point where the eigenvalues start to level off.



From the graph we concluded that our data can be optimally splitted into 3 factors.

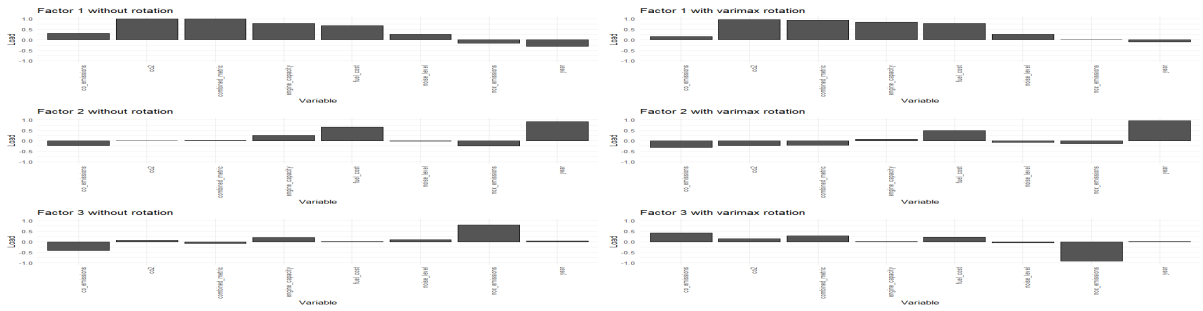
So now it's time to start our analysis by using the R command factanal(), first without rotation and then with varimax.

From the first case, we get that the variance explained by the factors is equal to a 69% and the division of factors by their loadings like this:

- Environmental impact (high loadings for co2, combined metrics, engine_capacity, and fuel_cost, which means the factor is related to them)
- The evolution of the price of the fuel in the time (high loadings for year and fuel)
- Chemical emissions (high loadings for the nox emissions and inverted high loadings for the co emissions)

We also have checked the high quality of the analysis by both proving that the values for the uniqueness and the loadings are higher than 0.4 and well distributed in the factors.

Then we try to use the varimax rotation, receiving the same explanations for the variance and the same factors. However the contribution of the variables to these factors changes a bit.



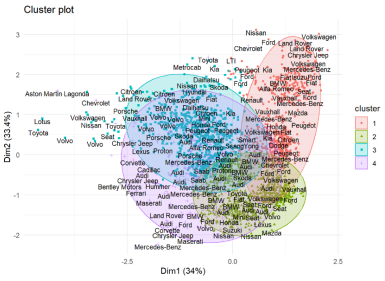
Having considered all of this we have taken the value for the factor analysis without rotation, having reduced the dataset complexity, offering insights into vehicle environmental performance, economic trends, and emissions

CLUSTERING TECHNIQUES

The clustering consists in splitting the data into groups, for it we have done several techniques starting by picking a sample of the data as it was very huge:

K-means & K-medoids (PAM)

We arbitrarily chose a k (5) to split the data into k clusters by the k-means and then we compared it with the k-medoids method
At both of them we got a little relationship (known by the silhouette algorithm which calculates the goodness of the clustering) between the data in the cluster, that's because of the arbitrary selection of the clusters.
To improve that result, we have used the fviz_nbclust() function with wss, silhouette and gap_stat, getting as a result k = 4.
We repeated the algorithm for both the k-means and medoids, getting a relation between the clusters of the 76% and what is more important, a relation inside the clusters.



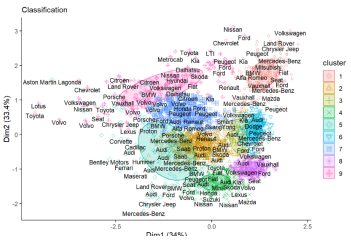
Cluster 1 is related with affordable, mass-market brands; Cluster 2 with Mid-range to premium brands; Cluster 3 with high and luxury brands and the fourth one with brands with unique features.
Depending on this clusters we can see this brands share aspects related to the environment impact, modernity and engine efficiency given the price of the car.

EM Clustering

We used the Mclust() function from the mclust package to fit a Gaussian finite mixture model on the scaled data, getting as a result VEV parameterization (ellipsoidal clusters with equal shape) with 9 components based on BIC (Bayesian Information Criterion)

BIC = -18669.28: Indicates the model's goodness of fit while penalizing complexity.

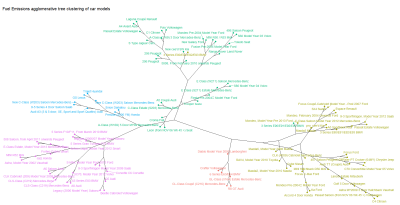
ICL = -19398.36: Similar to BIC but more stringent, incorporating clustering quality.

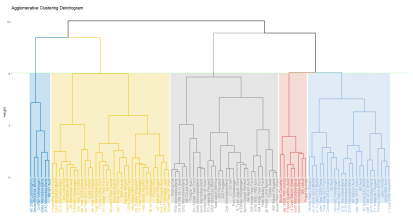


Hierarchical Agglomerative Clustering

After performing this type of clustering of the dataset after the PCA, using the first 5 components, which explained a 91.4% of the variance, and taking a sample of 100 observations (so we can see clearly the trees), we thought that the best distance to use would be the Euclidean since we are doing it over the PCA, and Ward's Linkage since single and complete linkage are really outliers-sensitive.

We performed a dendrogram and a phylogenetic tree and we successfully divided the observations in 5 clusters based on their emissions.





Hierarchical Divisive Clustering

The divisive clustering was performed using the diana() function from the cluster library and it gave us similar results with the agglomerative one but with some isolated outliers, due to their unique characteristics.