# Estatística e Análise de Dados

Ana Sofia Oliveira, Fábio Pereira

May 2019

## 1 Introduction

The data set we chose for this project is the Car Fuel Consumption between 2000 and 2013 [1].

This data was originally published by the Vehicle Certification Agency (VCA), an Executive Agency of the United Kingdom Department for Transport.

The prediction task we will perform is to predict the Carbon Monoxide emissions of the different cars based on their characteristics.

First, we list the variables in the original data set. In the following section, we describe the preprocessing performed on the data. Then, we explore the processed data graphically and numerically. We perform unsupervised learning in the fifth section and supervised learning in the sixth. Finally, we present our conclusions.

## 2 Data description

The chosen data set has 44838 instances and 25 variables. The variables in the data set were [2]:

1. **year** (numerical) – year of the evaluation;
2. **manufacturer** (categorical);
3. **model** (categorical);
4. **description** (categorical);
5. **euro_standard** (categorical) – European emission standards define the acceptable limits for exhaust emissions of new vehicles sold in the European Union and EEA member states [3];
6. **transmission** (categorical);
7. **transmission_type** (categorical) – manual or automatic transmission;
8. **engine_capacity** (numerical) – the swept volume of all the pistons inside the cylinders of a reciprocating engine in a single movement from top dead centre (TDC) to bottom dead centre (BDC) in $cm^3$ [4];
9. **fuel_type** (categorical);

10. **urban_metric** and **urban_imperial** (numerical) – results of fuel consumption test, urban cycle in metric (l/100km) and imperial (mpg) units;
11. **extra_urban_metric** and **extra_urban_imperial** (numerical) – results of fuel consumption test, extra-urban cycle in metric (l/100km) and imperial (mpg) units;
12. **combined_metric** and **combined_imperial** (numerical) – average of the two parts of the test, weighted by the distances covered in each part in metric (l/100km) and imperial (mpg) units;
13. **noise_level** (numerical) – results of noise test for passenger cars in dB;
14. **co2** (numerical) – $CO_2$ emissions in g/km;
15. **co_emissions** (numerical) – CO emissions in mg/km;
16. **fuel_cost_12000_miles** (numerical) – calculated using the official combined fuel consumption figure and fuel prices which are assessed each year;
17. **fuel_cost_6000_miles** (numerical) – calculated using the official combined fuel consumption figure and fuel prices which are assessed each year;
18. **standard_12_months** (numerical) – "road tax" rates for 12 months for passenger cars based on $CO_2$ emissions;
19. **standard_6_months** (numerical) – "road tax" rates for 6 months for passenger cars based on $CO_2$ emissions;
20. **first_year_12_months** (numerical) – "road tax" rates for 12 months for new passenger cars based on $CO_2$ emissions;
21. **first_year_6_months** (numerical) – "road tax" rates for 6 months for new passenger cars based on $CO_2$ emissions;
22. **date_of_change** (numerical) – year in which a car's specifications changed.

# 3 Data Preprocessing

Firstly, we eliminated variables 18 through 22 as they all had a large portion of missing values.

The fuel cost variables (16th and 17th) represented the same value, differing only by a factor of 2. To deal with this, we copied the 17th attribute values to the 16th variable, multiplying them by 2. Even with this, the new fuel variable still had 158 missing cases. We chose to simply remove those cases, resulting in 0% of missing values while only sacrificing 158 of the instances.

After that, we got rid of the *description* and *model* variables because they presented new categories in too many instances. We eliminated the imperial units version of variables 10, 11 and 12, as they are just a conversion of other attributes. We also eliminated instances that had negative carbon monoxide emissions.

We finally mutated the categorical variables so that the levels that had an absolute frequency of less than 250 would become a new level – "Other".

# 4 Exploratory Data Analysis

## 4.1 Variable distributions

### Categorical variables

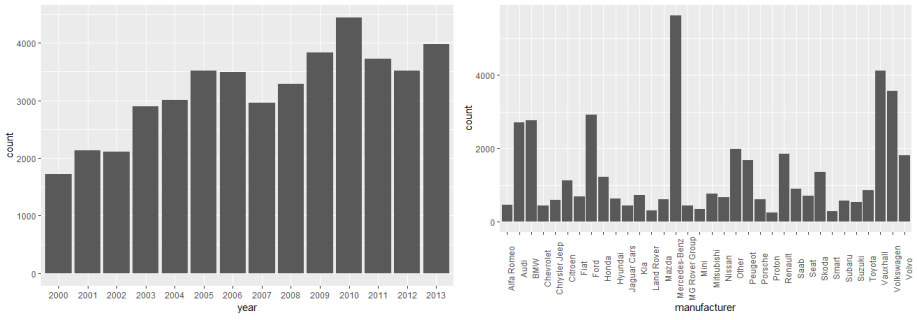


Figure 1: Bar plots for **year** and **manufacturer**.

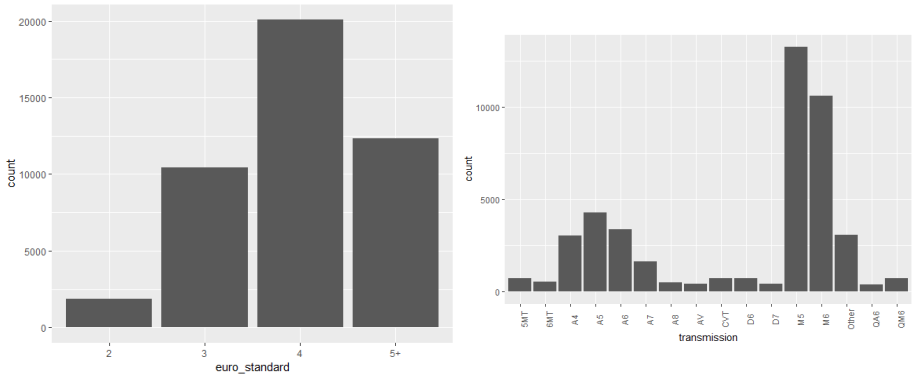


Figure 2: Bar plots for **euro_standard** and **transmission**.

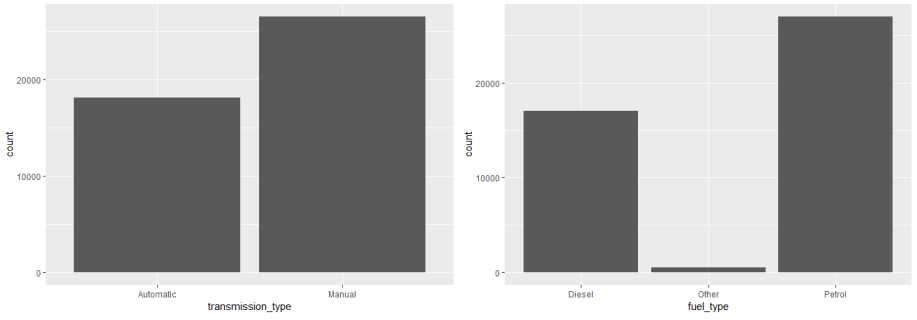


Figure 3: Bar plots for **transmission_type** and **fuel_type**.

### Numerical variables

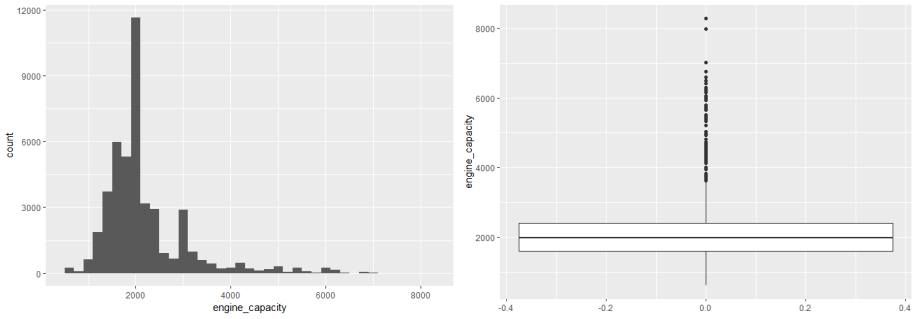


Figure 4: Histogram and boxplot for **engine_capacity**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 599 | 1598 | 1985 | 2176 | 2401 | 8285 |

Table 1: **engine_capacity** variable summary

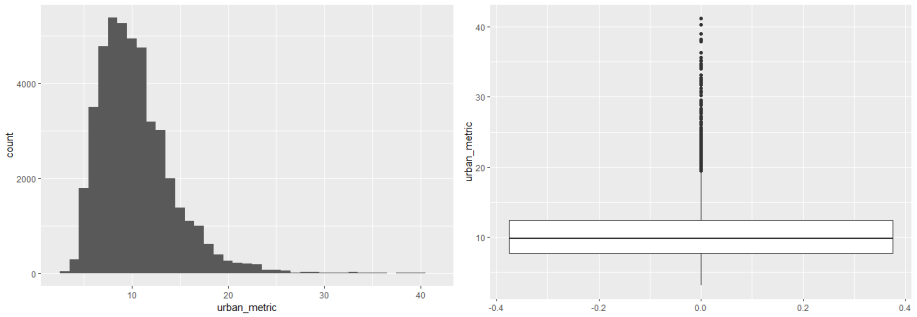


Figure 5: Histogram and boxplot for **urban_metric**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 3.10 | 7.70 | 9.80 | 10.45 | 12.40 | 41.20 |

Table 2: **urban_metric** variable summary

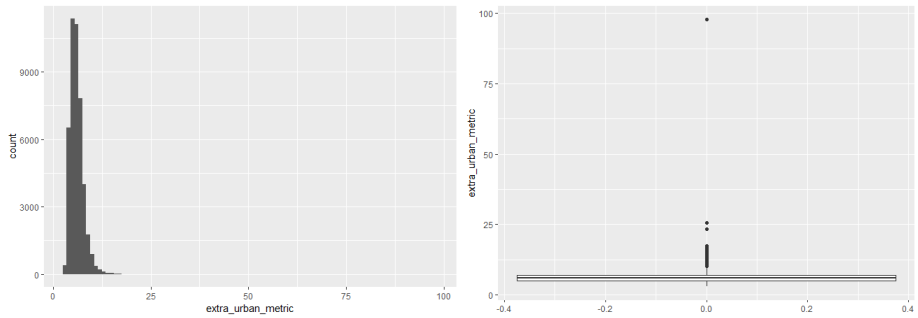


Figure 6: Histogram and boxplot for **extra_urban_metric**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 2.900 | 4.900 | 5.900 | 6.134 | 7.000 | 97.900 |

Table 3: **extra_urban_metric** variable summary

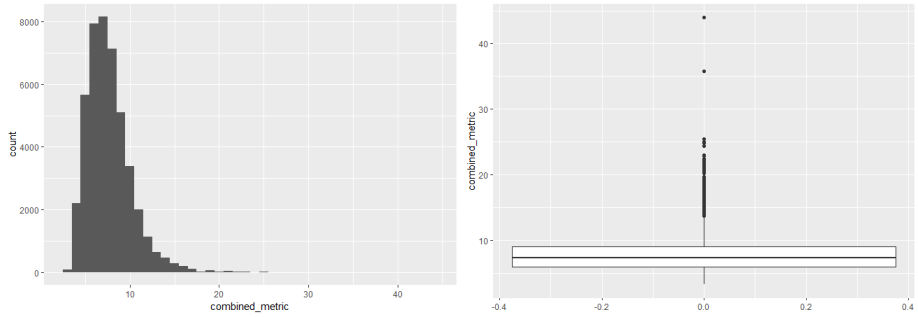


Figure 7: Histogram and boxplot for **combined_metric**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 3.200 | 5.900 | 7.300 | 7.712 | 9.000 | 44.000 |

Table 4: **combined_metric** variable summary

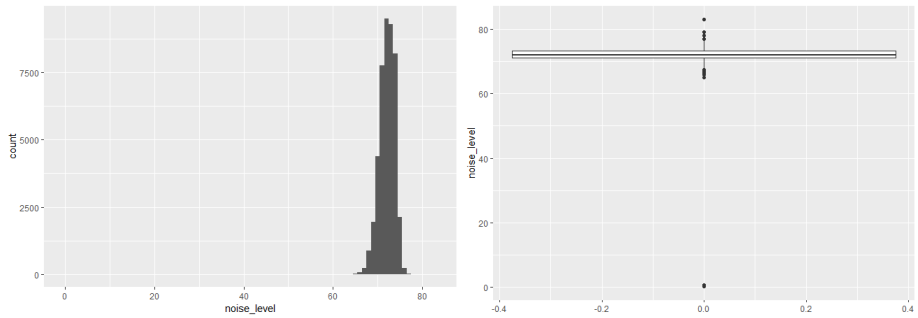


Figure 8: Histogram and boxplot for **noise_level**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 0.20 | 71.00 | 72.00 | 72.13 | 73.30 | 83.00 |

Table 5: **noise_level** variable summary

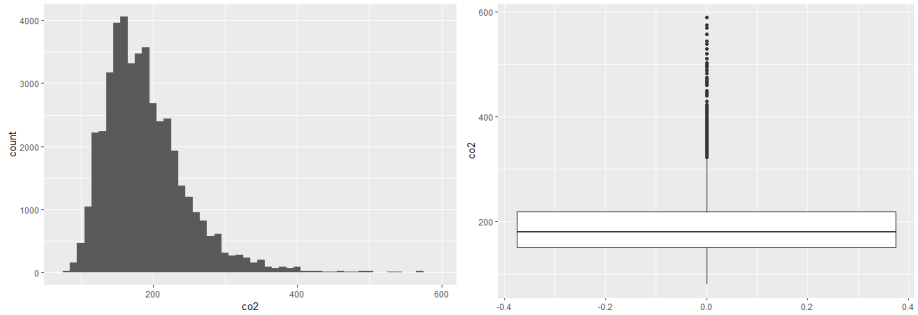


Figure 9: Histogram and boxplot for **co2**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 79.0 | 150.0 | 180.0 | 189.4 | 219.0 | 590.0 |

Table 6: **co2** variable summary

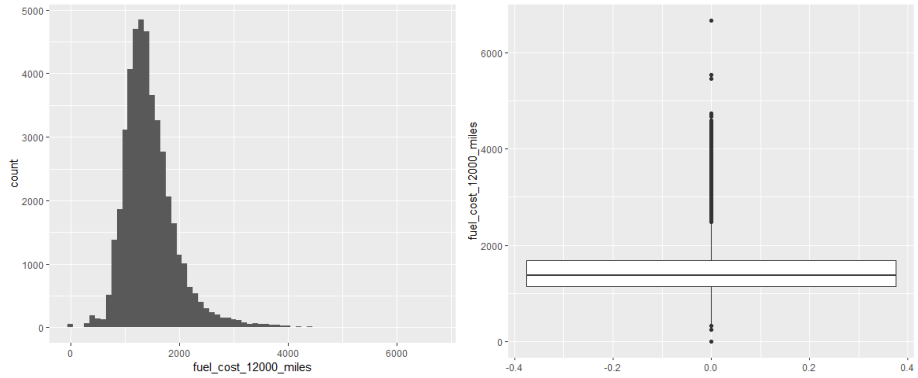


Figure 10: Histogram and boxplot for **fuel_cost_12000_miles**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 256 | 1142 | 1377 | 1462 | 1680 | 6658 |

Table 7: **fuel_cost_12000_miles** variable summary

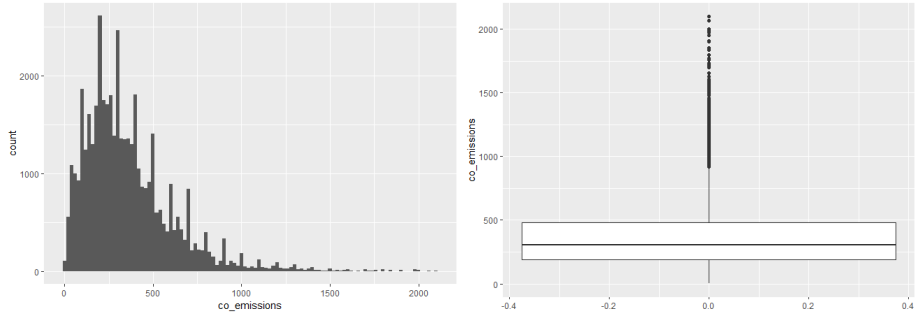


Figure 11: Histogram and boxplot for **co_emissions**

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|---|---|---|---|---|---|
| 0.07 | 189.00 | 302.00 | 361.27 | 480.00 | 2100.00 |

Table 8: **co_emissions** variable summary

## 4.2 Plotting the variables against the target

### Categorical variables

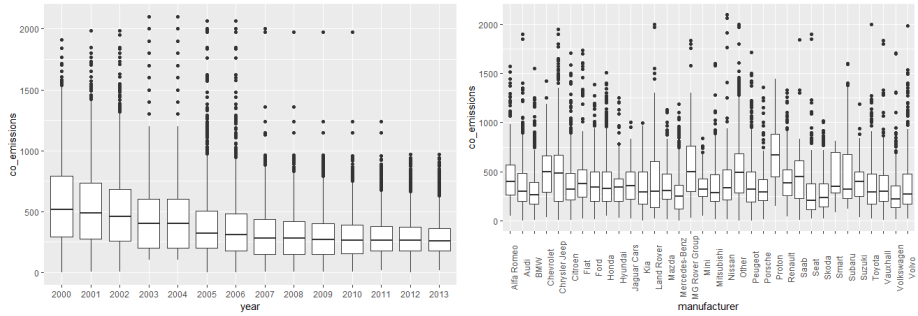


Figure 12: Boxplot of **year** and **manufacturer**, both against **co_emissions**

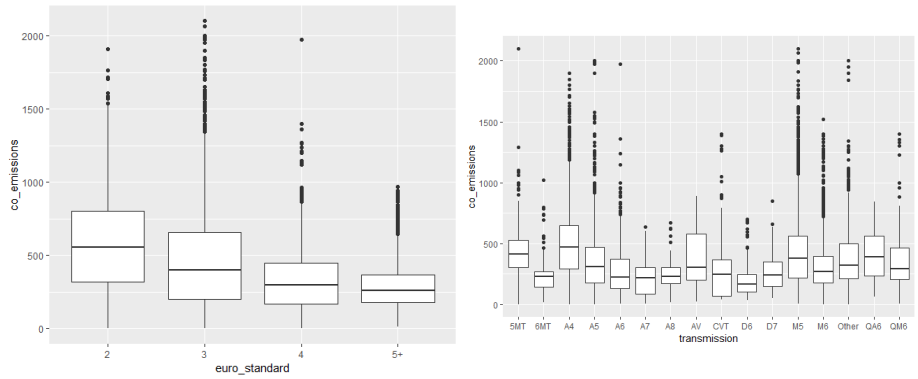


Figure 13: Boxplot of **euro_standard** and **transmissions**, both against **co_emissions**

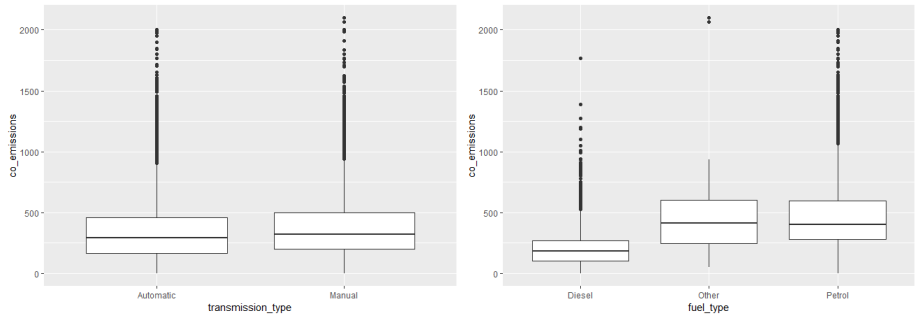


Figure 14: Boxplot of **transmission_type** and **fuel_type**, both against **co_emissions**

### Numerical variables

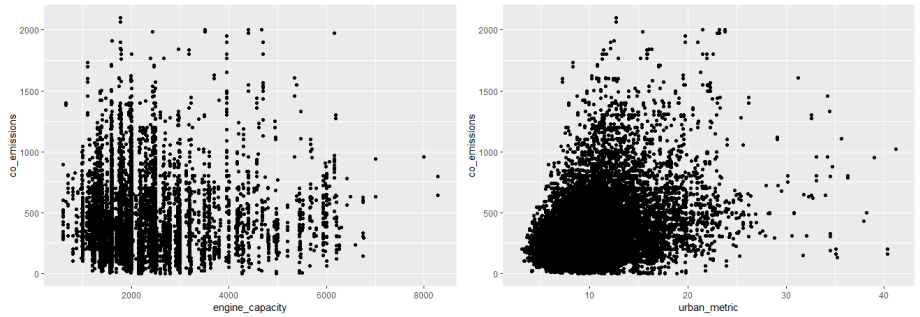


Figure 15: Scatter plot of **engine_capacity** and **urban_metric**, both against **co_emissions**

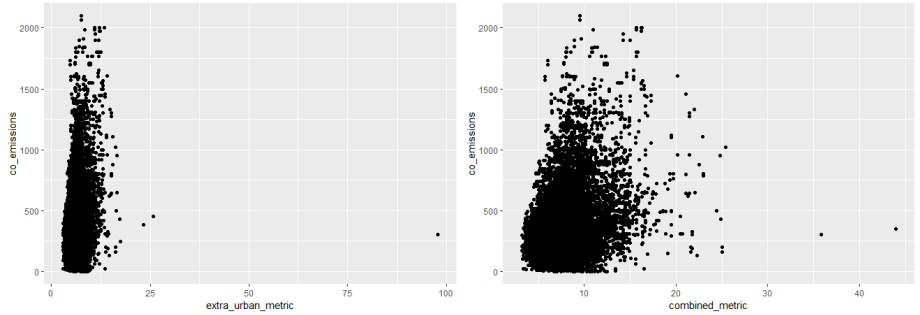


Figure 16: Scatter plot of **extra_urban_metric** and **combined_metric**, both against **co_emissions**

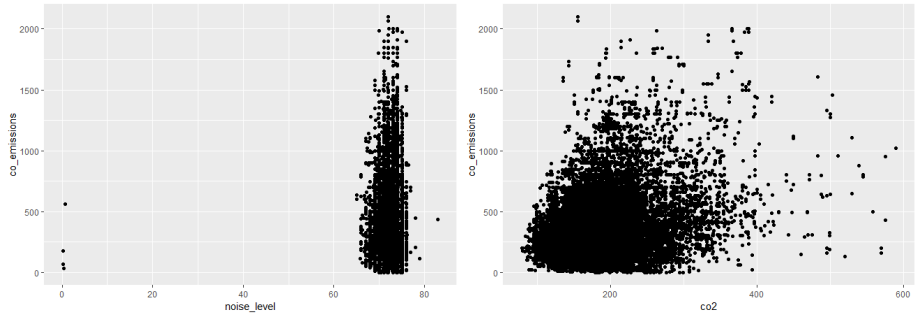


Figure 17: Scatter plot of **noise_level** and **co2**, both against **co_emissions**

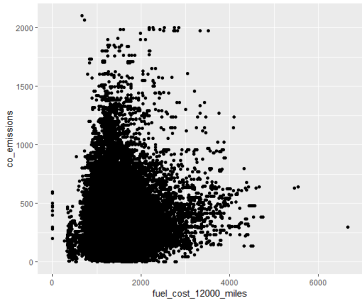


Figure 18: Scatter plot of **fuel_cost_12000_miles** against **co_emissions**

# 5 Unsupervised Learning

In this section, we first perform Expectation-Maximization on the target variable. Then, we cluster the data and lastly, hierarchically cluster the data. Throughout these subsections, we also give a brief introduction of each concept.

## 5.1 Expectation-Maximization

Expectation-Maximization is an iterative algorithm to determine the maximum likelihood estimates of parameters, allowing us to better understand the distribution of some variable.

To perform EM, we used the function *Mclust*. Given the data and number of clusters, this function returns the optimal model (parameters of Gaussian distributions) based on its BIC value.

The obtained Gaussian distributions and the real distribution can be seen in figure 19.

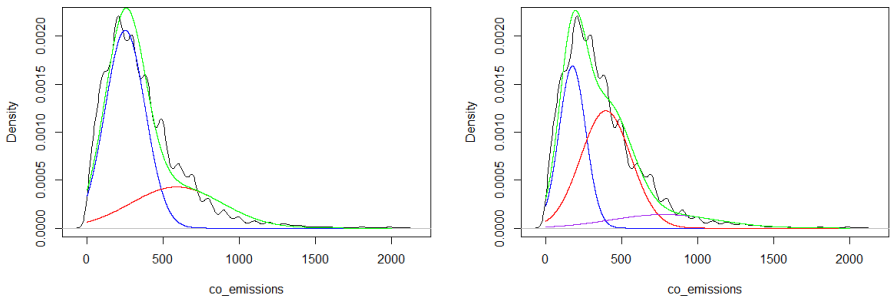


Figure 19: Expectation-Maximization of target variable for 2 and 3 distribution components. The black line is the real distribution and the green line is the combination of the remaining colors.

Note that the *Mclust* method, by default, chooses the best number of clusters for the data. However, because we thought it was overfitting the data (9 clusters was the optimal number), we chose to perform EM only with 2 and 3 clusters.

With the knowledge of the target variable distribution, one could now choose to model the data with a set of different models, each one focused on predicting a part of the distribution. We see no justification to do so in this project, as the target variable is similar to a Gaussian distribution.

## 5.2 k-Means

The k-means method for clustering (*kmeans* in R) groups the data accordingly to a given number of clusters.

The first question that arises is what is the optimal number of clusters for our data set. To determine this number, we used the function *NbClust* which, given a dataset, determines the best number of clusters by the majority rule. This method has a high computational cost, and so we had to limit the data fed to the method – instead of computing the best number of clusters for the whole data set, we computed it only for 5000 randomly selected instances (and it still took 24 minutes to finish). We assume that the results obtained for the 5000 instances hold in the whole data set. The resulting optimal number was, with 10 votes, 3 clusters.

In figures 20 and 21, we show all the plots for pairs of numerical variables, where each instance is coloured according to the cluster it belongs to.

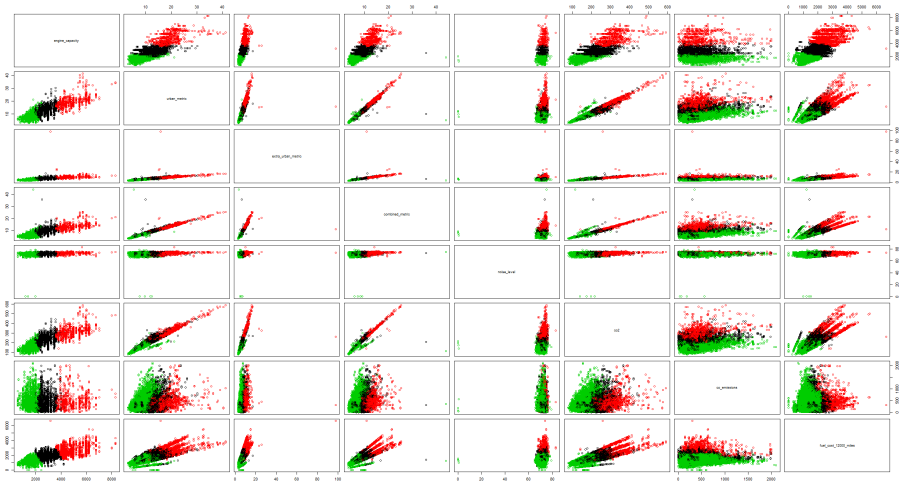

Figure 20: Plots of pairs of variables coloured according to the cluster of each instance, using 3 clusters.

Usually, one could, visualizing these figures, infer that the different clusters represent something that is not visible directly in the data. As we are not familiar with the car industry, we could not identify a conceptual difference between the clusters. This new knowledge about the clusters of data could be of the uttermost importance, since one could now get and/or engineer more data related to the concepts in question. Building a different model for each of the clusters could also be valuable to a supervised learning problem.

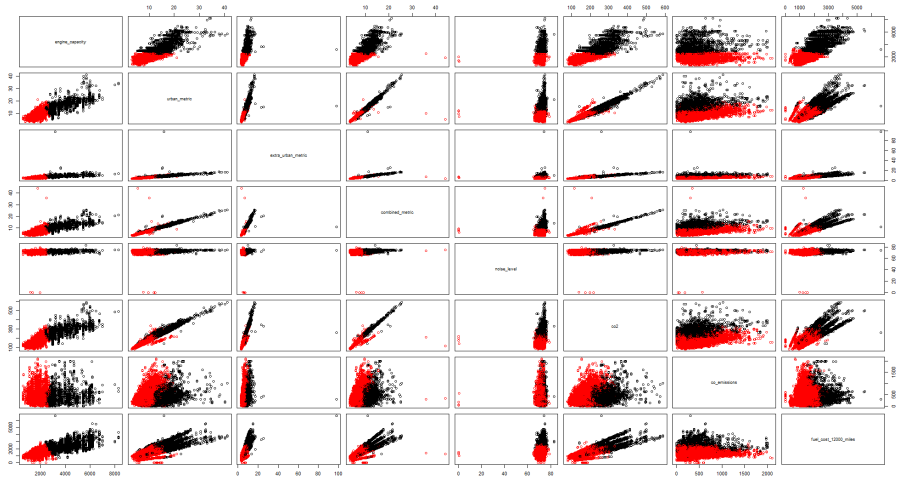

Figure 21: Plots of pairs of variables coloured according to the cluster of each instance, using 2 clusters.

## 5.3 Hierarchical Clustering

Hierarchical clustering, in contrast to just clustering, builds a hierarchy of clusters. This means that there is no predefined number of clusters. Instead, the method will cluster instances together by their similarity, and recursively do that to the previously created clusters themselves, until it reaches a root cluster that contains all the instances.

In figure 22, we show both the obtained hierarchical clusters and a heatmap. In the heatmap, the color represents the value of a given instance in a given variable, in respect to the range of values that that variable takes. It is also plotted, on the top and left side of the heatmap, a hierarchical clustering on both the instances and variables, respectively.

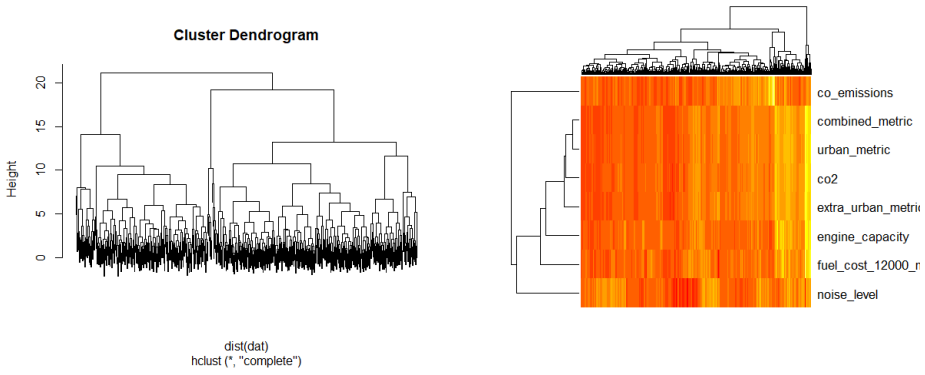


Figure 22: Dendogram and heatmap both computed on 500 random instances.

# 6 Supervised Learning

In this section, we show the achieved models, having as the target both the original target variable – regression problem – and its binary translation – classification problem. In both parts, we also evaluate the fitted models using both the holdout method and 10-fold cross-validation.

## 6.1 Regression

We trained linear regression, ridge regression and the lasso to predict the variable **co_emissions**. The ridge and lasso models were trained using the library **glmnet**. The resulting root mean squared error (RMSE) is shown in table 9.

| | Holdout Method | Cross-Validation |
|---|---|---|
| **Linear Regression** | 196.6471 | 197.1171 |
| **Ridge Regression** | 196.6258 | 197.5252 |
| **Lasso** | 196.7321 | 197.0168 |

Table 9: Resultsing RMSE for trained models.

The RMSE are all very similar and the differences in RMSE are small so the models perform approximately equally well.

To train the ridge and lasso models, we first performed cross-validation to select the $\lambda$ that resulted in the minimal mean cross-validated error. We then used this calculated $\lambda_{min}$ and trained the final model with only this value for $\lambda$.

We present in figures 23 and 24 the cross-validation curve as a function of $\lambda$ values used and the coefficient profile plot as a function on the log of $\lambda$ for both the ridge and lasso models.

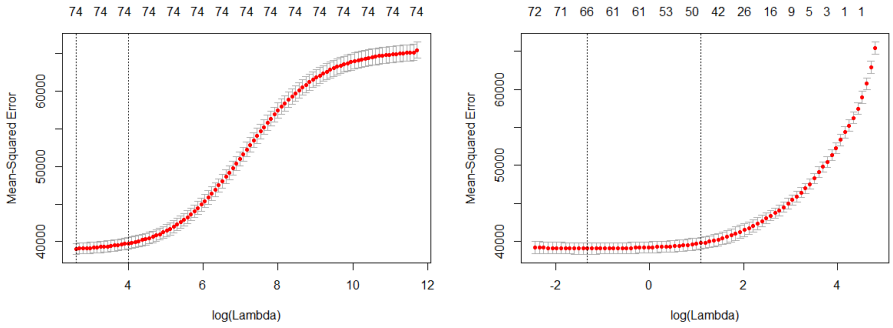


Figure 23: Cross-validation curve, and upper and lower standard deviation curves, as a function of the $\lambda$ ridge regression (left) and the lasso method (right). The numbers on the top of the graphic are the degrees of freedom for each model.

We present also, in figure 25, the coefficients of linear regression, ridge regression and lasso, for comparison.

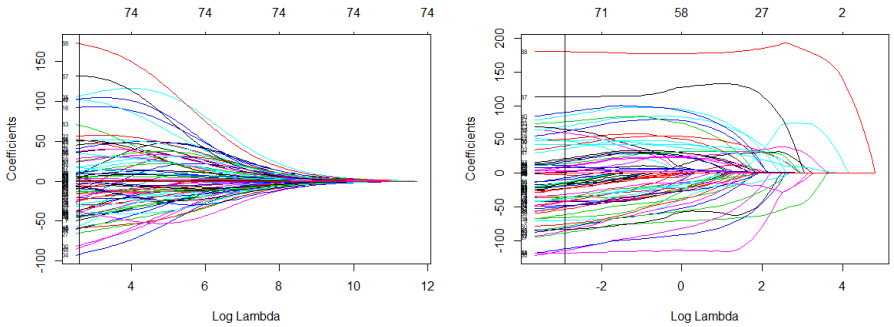


Figure 24: Coefficient profile plot of the coefficient paths as a function of $\lambda$ for ridge regression (left) and the lasso method (right). The vertical lines represent the chosen lambdas.

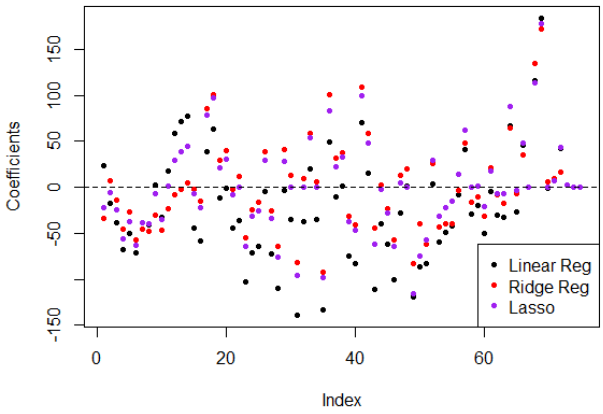


Figure 25: Coefficient values for linear regression, ridge regression and lasso.

### Bootstrap

Bootstrap is a technique that relies on random sampling with replacement from data to obtain a bootstrap sample of the same size as the original data. It allows the estimation of the properties of an estimator by measuring these properties on several bootstrap samples.

To test the power and utility of this method, we tried to use it to get an estimate of the standard deviation of each parameter in the Ridge regression.

In figure 26, the means and error bars for each parameter of the Ridge regression can be seen.

## 6.2 Classification

For classification, we used the logistic regression (LR), linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) methods. We converted

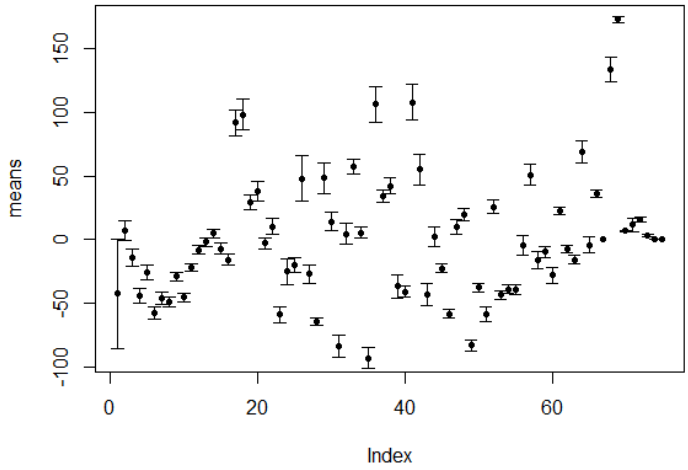


Figure 26: Mean and errors bar for each parameter of the Ridge regression.

the original target variable (**co_emissions**) into a binary variable, **y**, with values *TRUE* and *FALSE* according to whether the original value was bigger than the mean.

The accuracy results are presented in Table 10.

| | Holdout Method | Cross-Validation |
|---|---|---|
| **LR** | 0.770 | 0.769 |
| **LDA** | 0.768 | 0.767 |
| **QDA** | 0.708 | 0.717 |

Table 10: Resulting accuracies for trained models.

As can be seen, the linear models, both LR and LDA, give better results than QDA. This is also evident in the ROC curves for these models, Figure 27. These curves were calculated using the results of the holdout method and the package **ROCR**.

The LDA and LR give very similar results, with the accuracy in LR being slightly larger (with a 0.5 threshold).

To visualize the boundaries that each method produces, we trained a model with only 2 predictive variables, **engine_capacity** and **urban_metric**. The resulting boundaries are in figures 28 and 29.

In general, LR and LDA give similar but different results and boundaries because they calculate the coefficients differently. In this example, the boundaries look the same but the accuracy is higher by approximately 0.04 in LDA (with a threshold of 0.5 and using the holdout method).

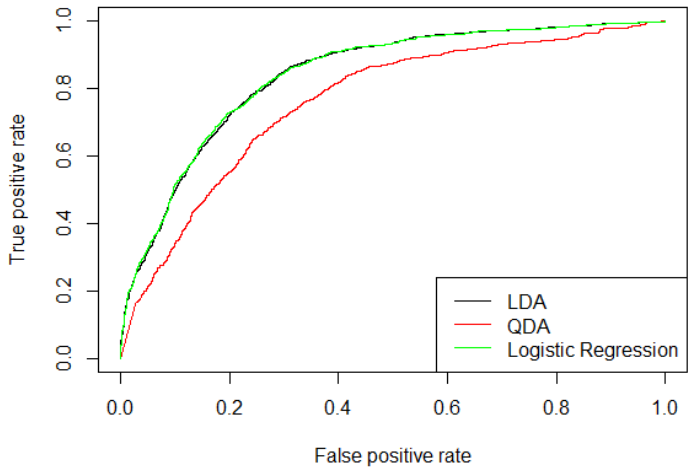


Figure 27: ROC curves for LR, LDA and QDA.

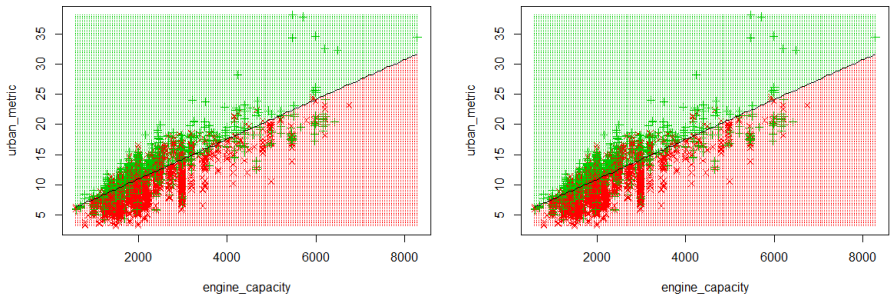


Figure 28: LDA (left) and LR (right) boundaries for models with 2 predictive variables.

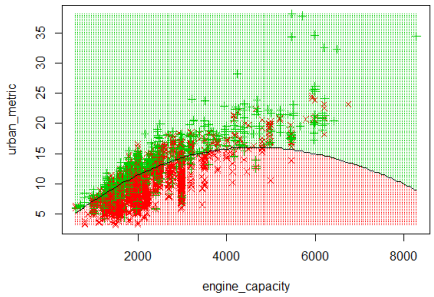


Figure 29: QDA boundary for model with 2 predictive variables.

# 7 Conclusions

In this report, we first showed the preprocessing of the data where we were able to eliminate only a few instances and maintain most of the information. Then, we visualized the variables' distributions and their distribution against the target variable.

Afterwards, we show the utility of unsupervised learning, using both EM and the k-Means method for clustering and hierarchical clustering. Next, in the supervised learning section, we divided the problem in 2 – regression and classification. For the regression part, we showed the obtained results of linear regression, ridge regression and the lasso. Furthermore, we show the utility of the bootstrap method. For the classification part, we displayed and compared the obtained results of LR, LDA and QDA.

# References


[1] https://www.kaggle.com/rasty1980/car-fuel-consumption-20002013

[2] https://carfueldata.vehicle-certification-agency.gov.uk/additional/aug2013/VCA-Booklet-text-Aug-2013.pdf

[3] https://en.wikipedia.org/wiki/European_emission_standards

[4] https://en.wikipedia.org/wiki/Engine_displacement