**AEDA**

Final Project

Raúl García Gómez

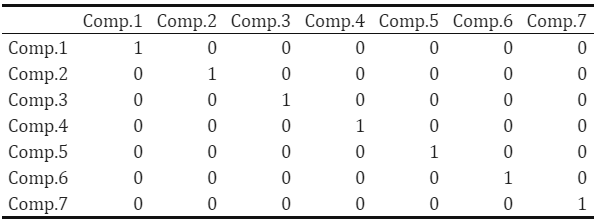
Marc Monclús

Ali Muhammad

**Problem 1**

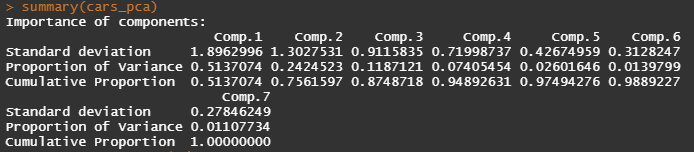
This dataset consists of a list of cars detailing a number of their characteristics. Our objective of using PCA will be to find some kind of relation between variables or to explain what common points the cars might have between each other.

To do so we have to begin by cleaning the dataset. The first step will be to get rid of the boolean variables that this set has because PCA is a model that uses quantitative variables. After that we will check for any outliers through plotboxes and we will erase anyone that falls outside the 1.5 times the IQR from Q1 or Q3 distance. Finally we checked the correlation matrix to see if there was any dependency between the variables, which we didn’t find.

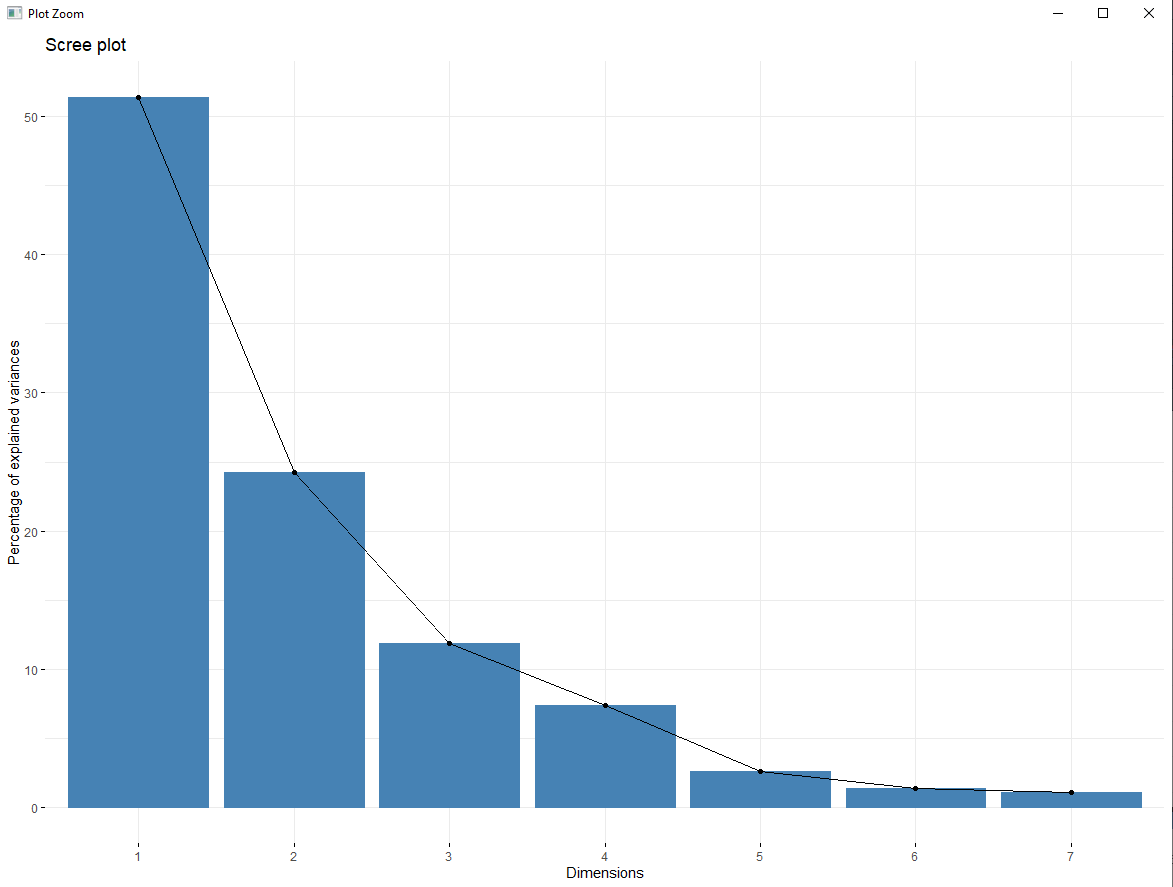


*Correlation Matrix*

With the EDA done, we could finally start the PCA. We obtained the PCA of the data with the ‘princomp’ function and checked that the first two loadings, the ones that should carry most of the importance of the variation, were not correlated, which they weren’t. In fact, these two loadings carried each 51,3% and 24,2% of the total variance, reaching 75,5% which is a solid figure. However the subsequent screeplot hinted through the position of its elbow that adding the third component would be more correct, which would make the loadings reach the 87,37% of the total variance explained but still maintaining the model within the three graphically representable dimensions.

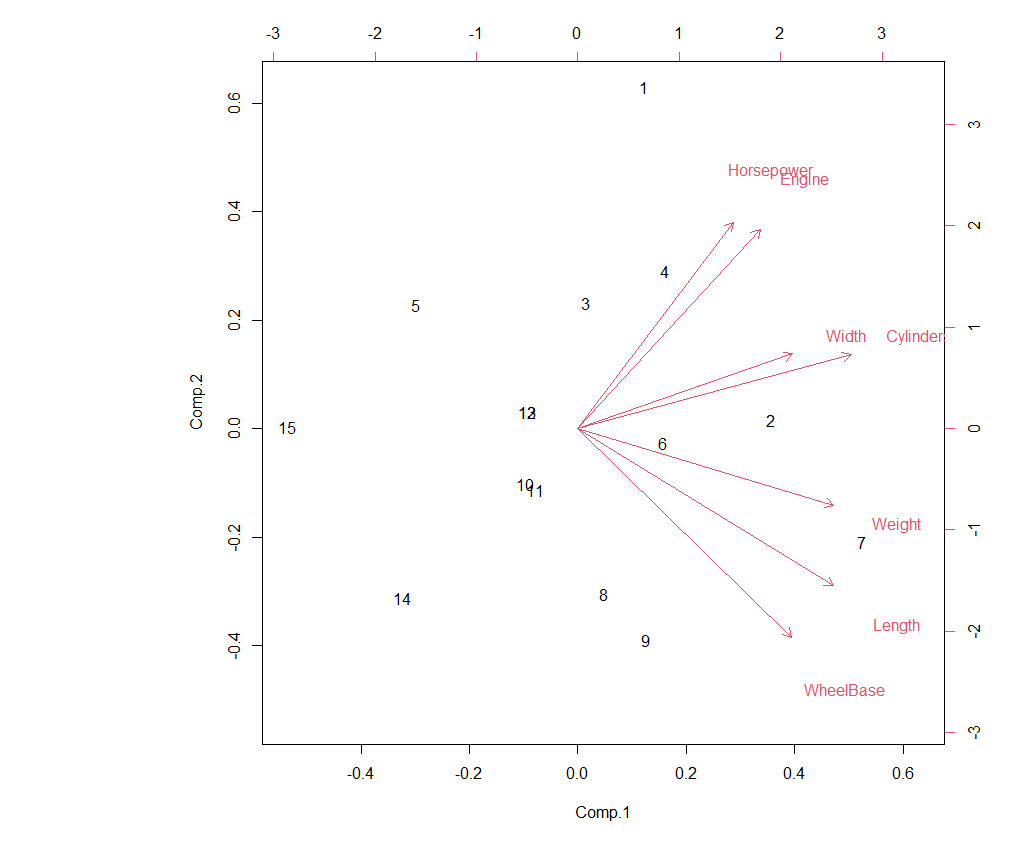


*PCA result*

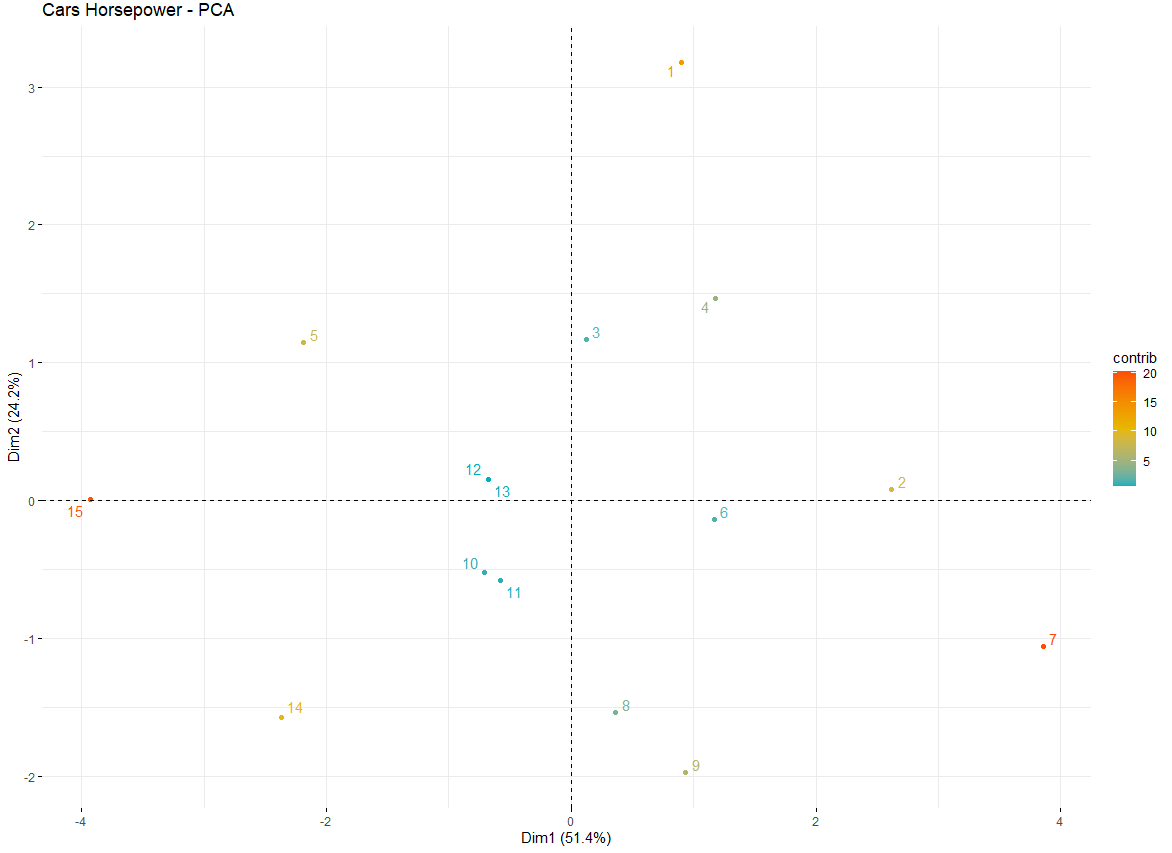


*Screeplot*

After this we decided to focus on the horsepower as the variable to look for revelations in our PCA. However the biplot didn’t show any great hint that would lead us to a characteristic that could explain higher or lower values of horsepower. Engine seemed that it could be the most important characteristic, followed by Width and Cylinder but with these plots, conclusions like these would border the bold statement status so we could reason that this analysis was inconclusive on the horsepower front.



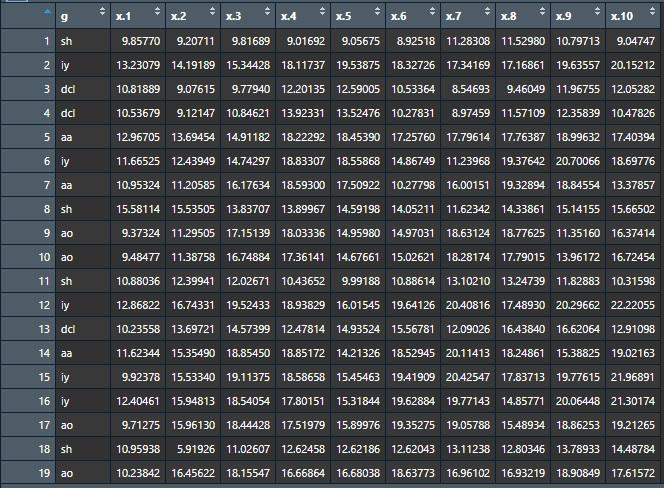
*Biplot*



*Contributions*

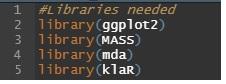
**Problem 2**

I understand the dataset of this problem to be the sounds that people make when speaking certain phonemes divided into small parts and transformed into numerical data (x.1 to x.256) classified into the sound they were making and the speaker that made that sound. For ease of representation and calculation time we will only use the first ten numerical columns and the column ‘g’ that tells what sound was being reproduced. The objective of using LDA would be to recognise a pattern that differentiates one phoneme from another in order to, for example, make a speech-to-text app be more precise with its translation.



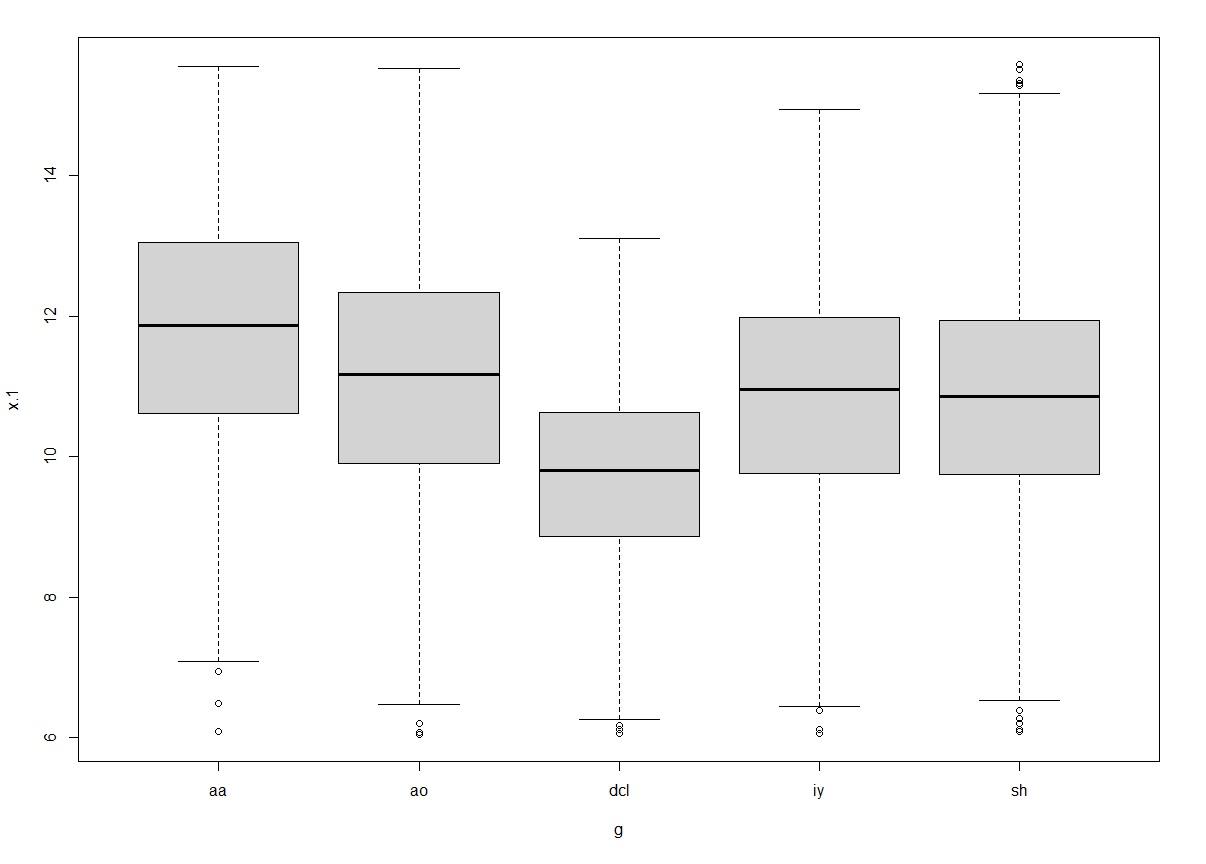
*Dataset to be processed*

First of all, we will load the needed libraries. ‘ggplot2’ is needed to add colour to plots, this will be very useful when checking on the plot whether we have found differentiated group or not after doing the LDA. ‘MASS’ is the library that contains the function to do the LDA and QDA analysis, while ‘klaR’ and ‘mda’ are the libraries that contain the function to do the RDA and MDA analyses respectively.



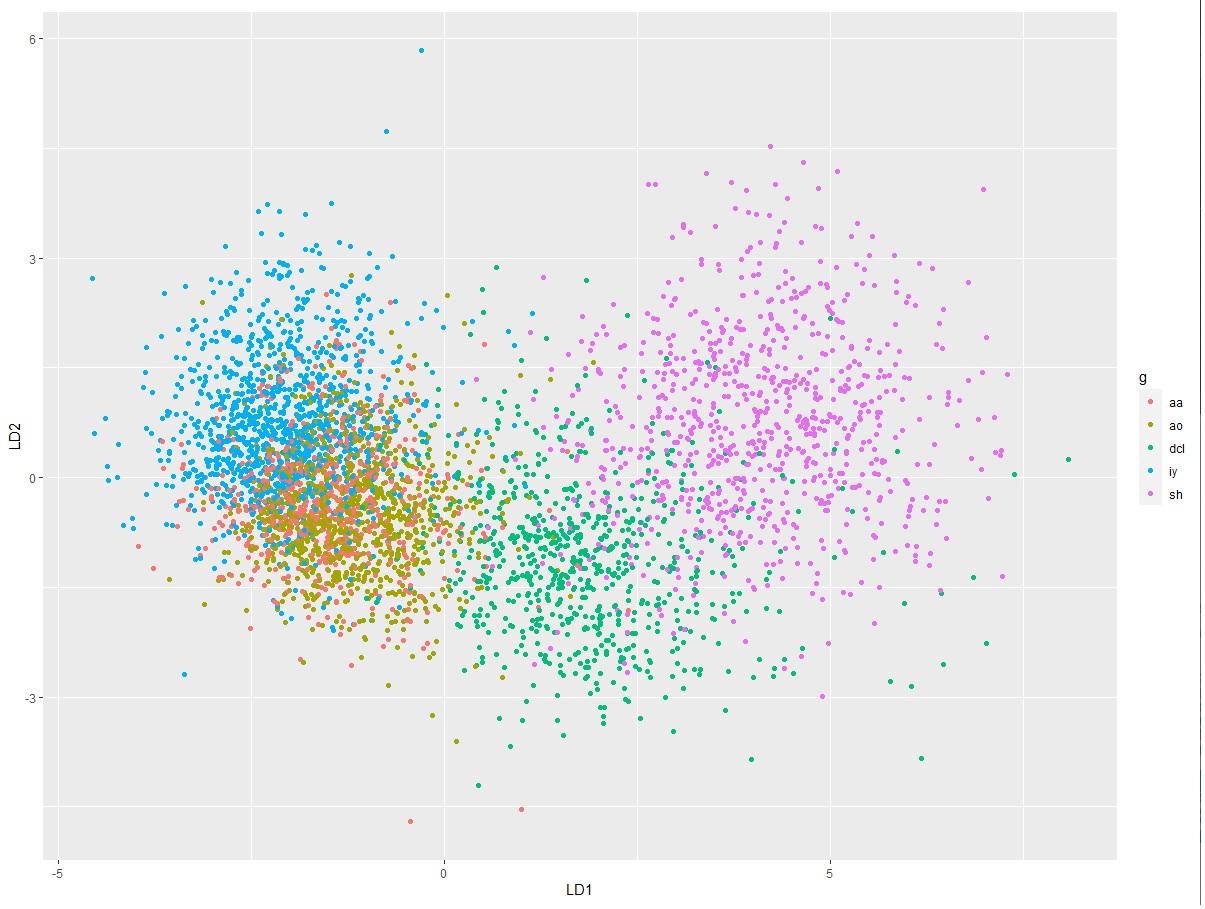
*Image showing libraries used*

For the EDA we will find out the outliers for each part of the phoneme using boxplots. That means the outliers will be those that are outside 1.5 times the IQR from Q1 or Q3. Once found through the combination of ‘boxplot.stats’ and ‘which’ functions we will erase them from the dataset and create a new one that we will feed to the models.

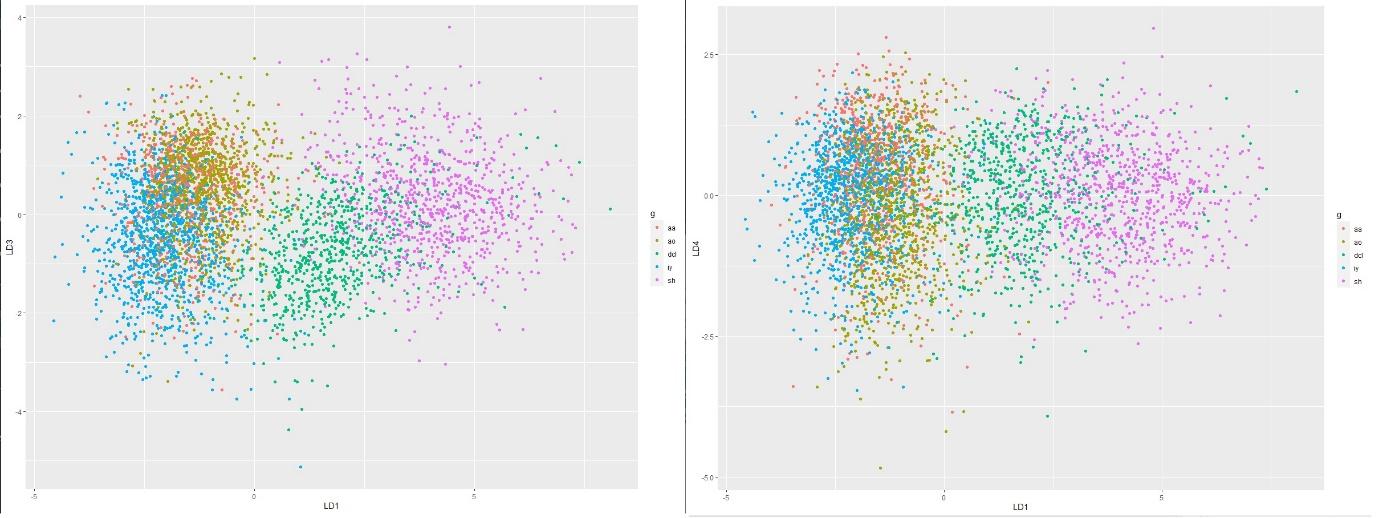


*Boxplot image showing outliers from x.1*

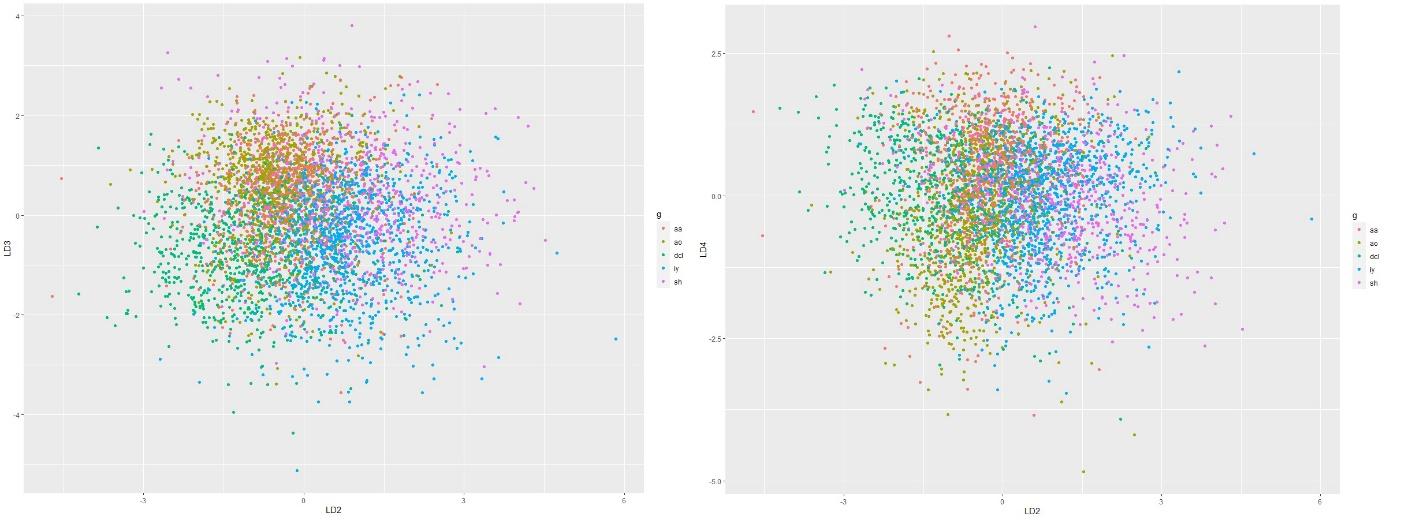
The first model used with the new dataset is LDA. The most interesting part about this LDA is that from the four linear discrimiants it finds, the first one is responsible for the 86,95% of the tracing. That means that just looking at this first discriminant we can have a pretty good idea of what phoneme we are looking at. To prove that we can use ggplot, the plots that use LDA1 show a clear distinction between the “dcl”, “sh” and the rest, while “iy” and “ao” can also be separated between the two although with “aa” mixed in. However, plots that do not use LDA1 show information that cannot be clearly separated or distinguished.



*LDA: LDA1 VS LDA2*

**

*LDA: LDA1 VS LDA3 + LDA1 VS LDA4*

**

*LDA: LDA2 VS LDA3 + LDA2 VS LDA4*

In this particular case QDA doesn’t offer much new information while the RDA and MDA models introduce a new concept to judge their models, the misclassification rate. In this concrete case the misclassification rate is around 25% (23,8% cross-validated to 24,2% in the RDA and 26,6% in the MDA). However, we can’t fully trust it because these models use the same dataset for the training and testing, which gives a bias to the misclassification rate.

Finally, we check the posterior probability of the models and of the dataset. It’s interesting to note that QDA, RDA and MDA give very similar results between each other while LDA has very similar probabilities to the dataset’s, especially in regards to the phonemes “sh” and “dcl” which were the most easily identifiable ones in the LDA plot. From this we can conclude that LDA would be the best model to use for this shortened dataset and we believe this to be because this dataset has a normal distribution and very close variances, the last point we believe being easy to see in the MDA analysis. Just as we have seen on the LDA analysis, a single linear discriminant is practically enough to judge quite correctly what phoneme are we looking at and it is that which makes the LDA the best model because the other models try to look for probabilistic patterns and variance relations that in this dataset seem to not be there. Although, it is also true that in the case of “aa” and “iy”, the two phonemes that seem to be the two that resemble each other and mix the most, the other models have a slight edge against LDA.