Group: 33 **Members:** António Saraiva (52795) - 35h; Matilde Carvalho (54838) - 35h; Rita Rodrigues (54859) - 35h

Introduction

With this work we intend to answer two main questions which are: **"What is the best regression model to predict "motor_UPDRS" from 16 voice measures?"** and **"What is the best binary classification model to predict "total_UPDRS" from 16 voice measures?"**. For this purpose, we use the data set in the "parkinsons_updrs.data" file, which is composed of 5,875 voices from 42 early-stage Parkinson's patients. The voice measures are the following:

- Jitter(%),Jitter(Abs),Jitter:RAP,Jitter:PPQ5,Jitter:DDP - Several measures of variation in fundamental frequency
- Shimmer,Shimmer(dB),Shimmer:APQ3,Shimmer:APQ5,Shimmer:APQ11,Shimmer:DDA - Several measures of variation in amplitude
- NHR,HNR - Two measures of ratio of noise to tonal components in the voice
- RPDE - A nonlinear dynamical complexity measure
- DFA - Signal fractal scaling exponent
- PPE - A nonlinear measure of fundamental frequency variation

The work developed includes two full testing and validation cycles of model selection and evaluation to answer each question. It was coded in python and several machine learning packages were used: pandas, numpy, scikit-learn, matplotlib and scipy.

Variables definition

The dataset included variables that are irrelevant to predict UPDRS, namely "subject#", "age", "sex" and "test_time". Because of that, we defined the independent variable **X** as the 16 voice measures only. To solve the question for regression, **Yr** was defined as an np.array of "motor_UPDRS". And to answer the question for classification, **Yc** was defined as another np.array of "total_UPDRS". As assumed in the statement, all positive instances (represented as 1 in the array) have values of total_UPDRS > 40 and all negative instances (represented as 0) are the remaining cases. After defining the variables, the number of positive and negative instances was calculated. The data set contains 1006 positive instances and 4869 negative ones.

Objective 1 - Produce the best regression model for 'motor UPDRS'

A model selection and validation pipeline was programmed. The steps were to split the data set into a training set and an independent validation set (IVS). Then the training set was used for training different models, changing and tuning hyperparameters, algorithms, and independent variables. During this training, for each model and its configuration, K-Fold cross validation was used. We assigned the number of partitions (k) to 5. The next steps in K-Fold are to train the model with k-1 partitions, make predictions with the non-used partition, repeat until all partitions have been processed, and evaluate the final model with the predicted values of all the test sets. The metric used to compare models with the training set was the RMSE (Root Mean Squared Error) as it evaluates the average error of the model.

The first algorithm we tried was the **Decision Tree Regressor**. In the following table is represented the hyperparameters that were tried and their corresponding values.

Hyperparameter	**Values**
splitter	["best","random"]
max_depth	range(1, 10)
min_samples_leaf	range(1,5)
min_weight_fraction_leaf	[0.1,0.2,0.3,0.4,0.5]
max_features	["auto","log2","sqrt",None]
max_leaf_nodes	[None,10,20,30,40,50,60,70,80,90]

Then **linear regression** was applied.

The linear regression model finds the line that best fits the data with coefficients that minimize the total error of the model. By using this model, we intend to establish a linear relationship between the independent variables and the response variable and to understand which variables best explain the model. With our data set, the objective will be to

build a good model that uses the independent variables of the X matrix to predict the clinical motor value UPDRS (Yr). Since we are in a multiple linear regression model the expression is:

$$\mu_x = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_p x_p \ .$$

We enter the code to get the model summary. We got an **R-squared of 0.106.** R-squared indicates the quality of the model, that is, the proportion of the variability of the dependent variable (response) that is explained by the independent variable (explanatory). As we obtained a very low value (close to 0) we can assume that the model is very weak. For this model we obtained a **p-value of 1.68e-94** which implies rejecting H0 to 5%. Thus, there is evidence in the sample that the beta vector is significantly different from 0, which leads us to believe that there may be a linear relationship between Y and at least one of the variables in X. Consequently, we define a new multiple linear regression model, model 2, in order to improve our model (get an R-squared value as close as possible to 1).

For this second model, we went to model 1 and saw all the variables for which we rejected H0 to 5%. Thus, the variables that reject H0 are the variables for which their beta is significantly different from 0, being Jitter(Abs), Shimmer:APQ5,Shimmer:APQ11, NHR, HNR, DFA, PPE and therefore they are in this second model. It should be noted that none of the chosen variables has a value of 0 in the confidence interval.

We removed from X_TRAIN the columns corresponding to the variables that left the model and executed the code of this second model.

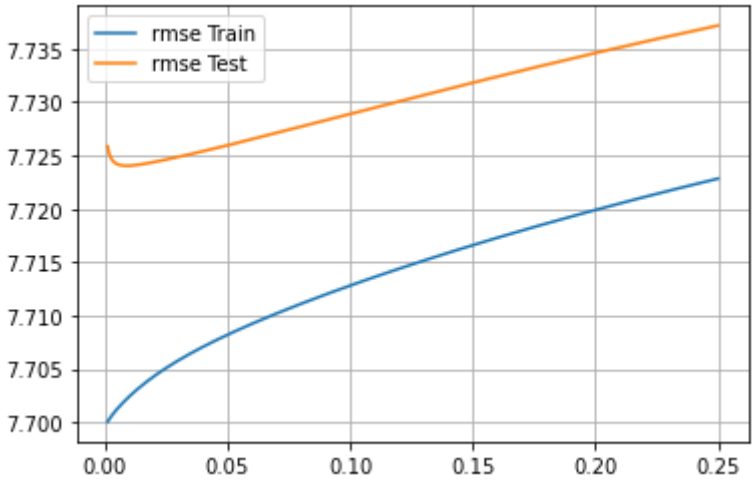
We got an R-squared of 0.101, an even lower value than model 1, which is surprising. The **Durbin-Watson value is 1.999**, which indicates that there is no correlation between the errors and therefore this regression assumption is verified. Another value to note is that the **Omnibus value is 436.351**, as this value is not close to 0, we conclude that the normality of Y given X (one of the assumptions of multiple regression) is not verified.

In this way, we investigated whether the assumptions of the multiple linear regression were verified for model 1. The assumptions are:

- Linearity of the relationship between X and y
- Homoscedascity of the residuals variance
- Independence of observations
- Normality of y given X

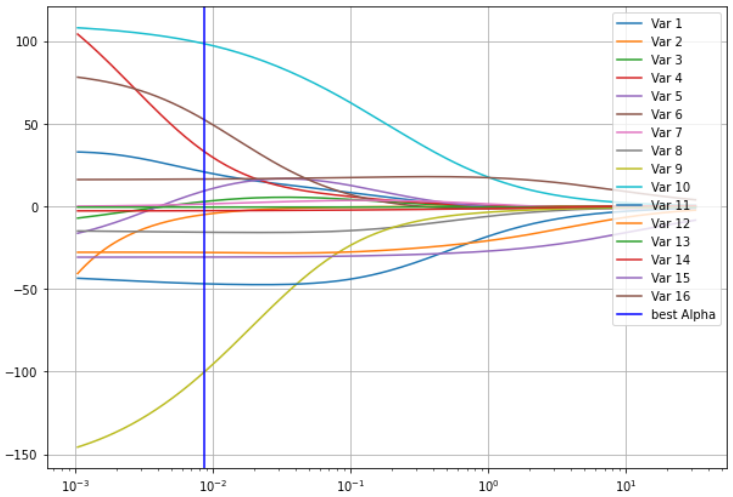
Given that in model 1 the value of Omnibus is equal to 368,531 we see that the assumption of normality of Y given X is not verified and as one of the 4 assumptions fails so, we cannot fit the data to a linear regression model. For this reason we should have verified if the 4 assumptions were verified in model 1 instead of creating a second model.

For the third algorithm, we decided to use the **Ridge Regression**. The alpha parameter was changed in order to obtain the alpha that minimizes the RMSE for the test set.



The best RMSE obtained was 7.7240 with a value of alpha equals 0.0086685.

After this, we computed all 16 parameters for different values of alpha. With this computation, we can say that the variables Shimmer:APQ5 and Shimmer:APQ11 are between the most important to explain Yr, since the penalization was higher and this same penalization emphasizes the importance of each variable in its contribution for explaining Yr. This is according to the variables that were most important in linear regression.



For the fourth algorithm, **Lasso Regression** was used. This model is very similar to Ridge Regression but uses L1 regularization,which not only minimizes the MSEs (Mean Squared Error) but also the modulus values of the parameters estimators.

It also has the alpha parameter like the Ridge Regression, which defines how much to penalize the actual parameters and this penalization will emphasize the importance of each variable in its contribution to explain Yr. But there is another difference, which is that for higher values of alpha, only a few variables will enter the model, which is a very good criterion to identify the most important variables in a model.

Similar to what we did with the Ridge Regression, we started by changing the hyperparameter of alpha, to get the best value of RMSE. In this model, the best RMSE was 7.7318 with a value of alpha equals 0.0010110.

Just like in the previous algorithm, we computed all 16 coefficients for different values of alpha and concluded that the variables Shimmer:APQ11 and Shimmer:DDA are between the most important to explain Yr.

Since the assumptions of linear regressions didn't hold, to choose the best model we compared the best RMSE of Decision Tree Regressor, Ridge Regression and Lasso Regression. The following table summarizes the results.

Algorithm	**RMSE**
Decision Tree Regressor	7.732927649867136
Ridge Regression	7.724038425264214
Lasso Regression	7.731844331098596

To finish the model pipeline, the best model, Ridge Regression with alpha equals to 0.008668511500530142 and max_iter to 100000 was fitted with all the training set data and the IVS was used to assess the model quality.

Objective 2 - Produce the best binary classification model model for 'total UPDRS'

A pipeline as described in the previous objective was also programmed to solve this objective. The algorithms used in the training set were only Decision Trees and Logistic Regression. The metric used in this case was the Matthews Correlation Coefficient (MCC). We concluded that it would be a good metric because the dataset has 1006 positive instances, which is much smaller than the 4869 negative instances.

The hyperparameters tried for decision trees were the following:

Hyperparameter	**Value**
criterion	["gini", "entropy"]
max_depth	range(1,10)
min_samples_split	range(2,10)
min_samples_leaf	range(1,5)

The best MCC obtained was 0.177224.

Logistic Regression with max_iter = 1000 was also tried and the result was a MCC of 0.0, meaning this model is the equivalent of an average random prediction.

Obviously the best model to solve the problem was a decision tree. The parameters are: 'ccp_alpha': 0.0, 'class_weight': None, 'criterion': 'entropy', 'max_depth': 6, 'max_features': None, 'max_leaf_nodes': None, 'min_impurity_decrease': 0.0, 'min_samples_leaf': 1, 'min_samples_split': 6, 'min_weight_fraction_leaf': 0.0, 'random_state': None, 'splitter': 'best'. This model was fitted with all the training set data and the IVS was used to assess the model quality.

Results

The best regression model for 'motor UPDRS' is Ridge Regression with alpha equals to 0.008668511500530142 and max_iter to 100000. The statistics obtained were the following:

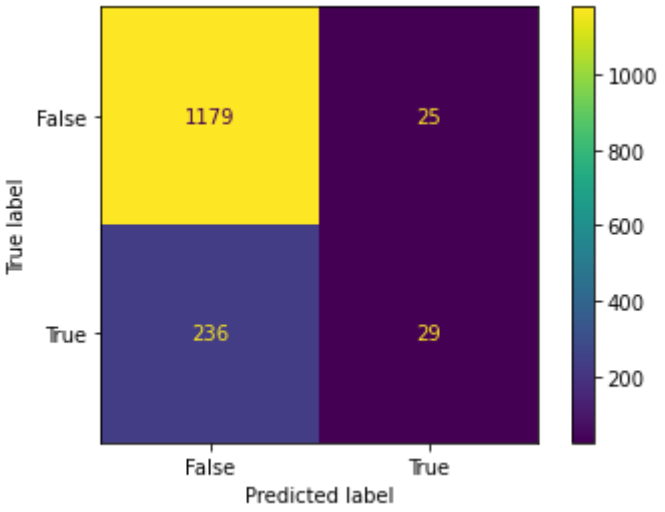
Statistic	**Value**
RVE	0.10082289301363301
RMSE	7.754671500165321
Correlation Score	0.3182 (p-value=6.538159e-36)
Maximum Error	19.09144291185206
Mean Absolute Error	6.573919178000186

RVE is a value between 0 and 1 and we obtained 0.1008, meaning the overall quality of the model is bad. A correlation score of 0.3182 tells us that the truth values and the predictions aren't that correlated.

The best binary classification model model for 'total UPDRS' produced the statistics:

Statistic	**Value**
Precision	0.5370
Recall	0.1094
F1 score	0.1818
MCC	0.1812

Confusion Matrix



The precision indicates the fraction of true positives within all identified positives and the value obtained is not that bad. Recall indicates the fraction of true positives out of all existing positives. The result obtained is too low. The same happens for F1 score and MCC.

Conclusions

With this work we were able to answer "What is the best regression model to predict "motor_UPDRS" from 16 voice measures?" and "What is the best binary classification model to predict "total_UPDRS" from 16 voice measures?" using machine learning. The best regression model has a Maximum Error of 19.09144. By observing the column of motor_UPDRS this seems a significant error. Since we are trying to predict values that might influence how a patient is diagnosed and treated, we conclude that there is a need to try different approaches as this is unacceptable. Regarding the best classification model, in the confusion matrix we can see that the model classifies 236 cases as False that are actually True. This is again not acceptable, because saying that a patient doesn't have a disease when he/she has, can be the difference between getting the adequate treatment or not.