Aprendizagem Automática 2022/2023

First Home Assignment

Group 6: Alexandre Sobreira (59451), André Dias (59452), Tiago Rodrigues (49593)
Hours contributed to the work: Alexandre - 29; André - 29; Tiago - 29

Objective 1

The main purpose of Objective 1 is to produce the best regression model using the variable 'motor_UPDRS'. The first thing to address is: which regression models can we use to tackle this task? The answer is Linear Regression, Ridge Regression, Lasso Regression, and Decision Tree Regression. In this work, all the mentioned models were tested to make comparisons between them and evaluate which one provides the best results.

Initially, the data was read into a pandas dataframe, and the existence of null and duplicate values was assessed. The data was then divided into two separate dataframes (X_Park and y_Park), the last one corresponding to the variable 'motor_UPDRS'. Following this, these dataframes were converted into arrays so they could be properly used for the various models.

A good rule of thumb in machine learning is to properly validate the models created. With this in mind, the data was split into an IVS (Independent Validation Set) and a working set (X_TRAIN and y_TRAIN). To train and validate models this working set was split into a training set (X_train and y_train) and a testing set (X_test and y_test).

To validate the models, K-Fold cross-validation (CV) was used with 10 folds. K-Fold CV was chosen instead of simple CV because K-Fold tends to minimize sampling bias which is more prominent in simple CV. Since the latter only splits the data into train and test set once, instead of k times, the data in this test set can be unrepresentative of the whole dataset and therefore generate bias. K-fold CV provides a solution to this problem by dividing the data into folds and ensuring that each fold is used as a testing set at some point. Leave-one-out CV was not tested because this type of validation is very costly computationally and only suggested for small datasets or if the computation can be handled. Since the data has almost 6000 rows and several columns, it would require a very high computational usage and take a lot of time. On top of this, Leave-one-out CV makes (total rows – 1) splits which can cause it to inherit the bias of the data.

By fitting the Multiple Linear Regression to our data without any constraints, it was possible to observe that the basic statistics for this model were very poor (Table 1). The Ratio of Variance Explained (RVE) is closer to 0 which means that the variance of our model is not well explained, only capturing 15% of the data variance. The Root Mean Squared Error (RMSE) is relatively high given the range of the 'motor UPDRS' variable (between 5 and 40). Although there is a resemblance of positive linearity between our truth and predicted values, this relationship is moderate since the correlation-score (CS) is close to 0.4. The Maximum Error (ME) and Mean Absolute Error (MAE) are both also high considering the range of the 'motor UPDRS'.

Given the clinical context of the problem, the metric chosen as most relevant was the MAE because it provides an unweighted measurement of the average error between paired truth-prediction observations and uses the same scale as the data being measured, making it easily interpretable. As such, upcoming optimizations were performed with the aim of minimizing the MAE of the model.

Following this, Linear Regularized Models were tested. These were applied to our data using both Ridge and Lasso Regressions. To fit the data to these models, a standard scaler was firstly fitted to our X_train and applied to both X_train and X_test. Next, in order to identify the "sweet spot" of the alpha values for each regression, different alphas were tested, whose MAE statistics can be seen in Figure 1.

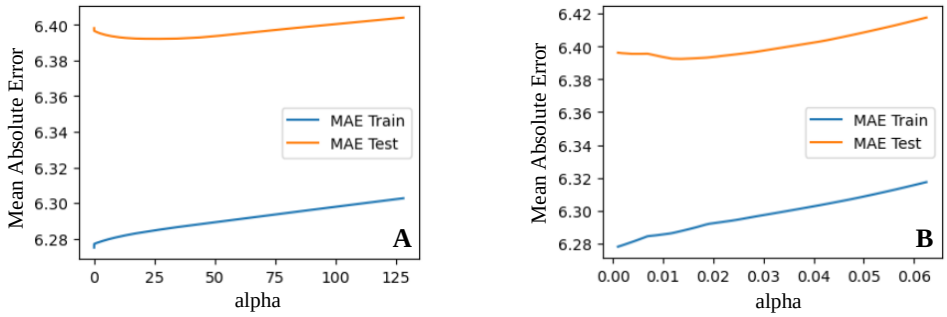

Figure 1 – Mean Absolute Error obtained for several models of Ridge (A) and Lasso (B) Regression using different values of alpha.

Analyzing these images, it is possible to visualize that the "sweet spot" can be found around an alpha value of 20 for Ridge Regression and 0.015 for Lasso Regression. Using these alphas, a K-Fold CV was performed for both, whose statistics can be seen in Table 1. The obtained statistics demonstrate that regularization did not have a positive impact on the regression's statistics, since they saw negligible differences from Multiple Linear Regression.

For the last regression model, the Decision Tree Regressor was tested. Initially, a model without any constraints was fitted to the data. The results obtained from its K-Fold CV can be seen in Table 1. As observed, all metrics showed a clear improvement compared to the previous models, making it a good candidate for further optimization with a parsimonious tree in mind. To investigate the potential of this model, some hyperparameters were tested: max_leaf_nodes, max_depth, min_samples_split and min_samples_leaf. These were chosen because they seem to have the most observable impact on the tree, making them easier to explain and comprehend. To obtain a broad picture of the impact each hyperparameter has on MAE, a model was fitted for each value in a range of 2 to 100, as presented in Figure 2.

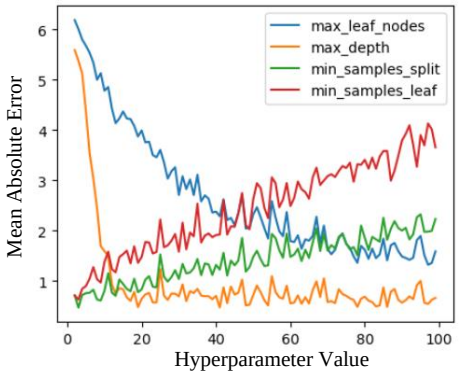

Figure 2 – Mean Absolute Error obtained for several Decision Tree Regressors with a range of distinct hyperparameters from 2 to 100.

From Figure 2, it was possible to conclude that the min_samples_leaf and min_samples_split should not be changed from their default value, since any increase led to worse MAE values. Furthermore, the MAE for max_depth reached a plateau around a value of 20 while for max_leaf_nodes it continuously decreases up until a value of 100.

Based on these results, three models with the following hyperparameters were tested: max_depth = 17; max_leaf_nodes = 150; both hyperparameters together. The K-Fold CV results for each model can be observed in Table 1.

As evidenced by the results obtained, the unconstrained model was the model with the best MAE. As such, this model was validated with the IVS, whose results can be seen in Table 1. The statistics obtained by simple CV using the IVS with the unconstrained model trained for all the data (X_TRAIN), suggest that this model is adequate for our data since they did not worsen.

Table 1 – Basic statistics from the main Regression models tested. A color grade was applied to MAE to visually distinguish between the best models.

Objective 1 Models		Statistics				
		RVE	RMSE	CS*	ME	MAE
Mutiple Linear Regression		0.1504	7.4721	0.3882	40.8879	6.3240
Ridge Regression		0.1524	7.4632	0.3905	36.1390	6.3239
Lasso Regression		0.1515	7.4671	0.3893	33.0061	6.3287
Decision Tree Regression	Unconstrained	0.9268	2.1930	0.9636	23.4010	0.6222
	max_depth = 17	0.9269	2.1916	0.9636	23.9350	0.6396
	max_leaf_nodes = 150	0.9239	2.2357	0.9617	23.4366	1.1673
	max_depth = 17 max_leaf_nodes = 150	0.9245	2.2276	0.9620	23.4366	1.1608
	IVS (Unconstrained model)	0.9378	2.0490	0.9688	22.9790	0.5626

*all p-value ≤ 0.001

Objective 2

In the case of Objective 2, the main task is to produce best binary classification model using the variable 'total_UPDRS'. Some of the known binary classification models include the Decision Tree Classifier and Logistic Regression, which were both tested.

In a similar fashion to Objective 1, the data was prepared and processed, but this time the 'total_ UPDRS' was used as a dependent variable (y_Park) and converted into a NumPy array with the condition 'total UPDRS' > 40 applied (the instance was considered positive if > 40 and negative if <= 40). The same splitting and validation procedures done in Objective 1 were also made.

Given the clinical context of this problem, the F1-score was chosen as the most significant statistic. On one hand it takes into consideration the precision, which is very important in this context to discriminate the patients that truly have Parkinson (True Positives), from the ones that were diagnosed with Parkinson but don't have it (False Positives). On the other hand, it also considers the recall which takes into consideration the patients who were diagnosed as not having Parkinson but have it (False Negatives). Both metrics are important in a diagnosis context and since the F1-score acts as a middle ground between them, it was chosen.

The first model tested for this objective was the Logistic Regression. This regression requires the scaling of the data before fitting, which was performed as described in Objective 1. A K-Fold CV was then applied to this model and some basic statistics were calculated, which can be seen in Table 2.

From an analysis of the statistics obtained for the Logistic Regression, it is possible to see that the precision is very low, indicating that are several instances where the patients were diagnosed with Parkinson's but did not actually have it (False Positives). Recall is extremely low which means that several patients that had Parkinson were not detected. Since the F1-score and the Mathews Correlation Coefficient take into consideration the precision and recall, their extremely low values were expected. The overall statistics of this model were very poor and, as such, no further optimization attempts were made.

Moving onto the Decision Tree Classifier, an unconstrained model was first fitted to our data and validated using a K-Fold CV, whose results can be seen in Table 2. Using this model, it was possible to see that all the basic statistics improved drastically when compared to the Logistic Regression model. As such, hyperparameter tunning was performed in an attempt to further improve this model. Similarly to the Decision Tree Regression in Objective 1, the chosen hyperparameters were max_leaf_nodes, max_depth, min_samples_split and min_samples_leaf for the same reasons. The same procedure described in Objective 1 was applied for the Decision Tree Classifiers to optimize the F1-score (Figure 3).

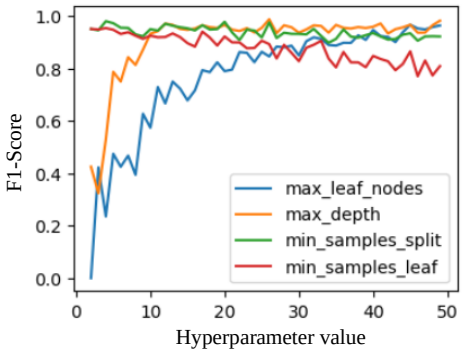

Figure 3 – F1-score obtained for several Decision Tree Classifiers with a range of distinct hyperparameters from 2 to 50.

From Figure 3, it was possible to conclude that the min_samples_leaf and min_samples_split should not be changed from their default value, since an increase in either led to a decrease of the F1-score. Furthermore, the F1-score reached a plateau around a value of 10 for max_depth while for max_leaf_nodes it continues to increase up until a value of 50. Based on these results, three models with the following hyperparameters were tested: max_depth = 10; max_leaf_nodes = 50; both hyperparameters together. The K-Fold CV results for each model can be observed in Table 2.

Table 2 – Basic statistics from the main Binary Classifier models. A color grade was applied to F1-score to visually distinguish between the best models.

Objective 2 **Models**		**Statistics**			
		Precision	Recall	F1	MCC
Logistic Regression		0.4000	0.0202	0.0384	0.0571
Decision Tree Classifier	Unconstrained	0.9637	0.9698	0.9667	0.9599
	max_depth = 10	0.9614	0.9093	0.9346	0.9223
	max_leaf_nodes = 50	0.9740	0.9433	0.9584	0.9503
	max_depth = 10 max_leaf_nodes = 50	0.9608	0.8955	0.9270	0.9136
	IVS (Unconstrained model)	0.9614	0.9387	0.9499	0.9391

As evidenced in Table 2, the unconstrained model was the model with the best F1-score. As such, this model was validated with the IVS after fitting it to X_TRAIN. The slight decrease of the basic statistics seen when validating this model with the IVS could result from a possible biased IVS sample. Another possibility may be that our model overfitted to the data, which is a real possibility since the model is unconstrained. Despite these observations the basic statistics for the unconstrained model are still very good.

Final considerations

From all the testing performed, the unconstrained models seem to be the best for both objectives. This could be undesirable since unconstrained models are more likely to overfit. To further validate our models a prospective validation would be recommended. In depth hyperparameter tunning would also be advised using pre established functions that analyze good combinations of hyperparameters (GridSearchCV or RandomizedSearchCV). Another possibility to enhance the models would be to do some form of feature selection. This could be made by pre-established functions (e.g. feature_importances), by evaluating the betas for different alphas in Lasso or by analyzing the trees and see the variables responsible for initial branch splits of the tree root (age, sex and test_time in this case). On the topic of trees, despite choosing the best model only based on MAE or F1-score, a consideration should be made for the visual complexity of each tree, i.e., a slightly worse model based on MAE/F1-score may result in a simpler tree, which could be desirable.