INFO 411: Assignment 2 report

Farzane Lalkhakpoor, Morten Jørgensen and Alejandro Marcano ()

October 14, 2023

1 Introduction

This is the report for assignment 2 in the INFO411 course. This report will contain a summary of all the files created and what has been learned when creating the files.

2 Summary of files

The Github repository consists of a data folder including raw and imputed train and test datasets. Scripts for EDA, Random Forest, Multivariate and Ridge Regression, SVM, and XGBoost are placed in under the main directory.

3 Exploratory Data Analysis

Note: For DS2, exploratory analysis was performed along with prepossessing procedures. We kept EDA and Pre-processing in one script and functions in another script to facilitate comparison and efficiency.

3.1 DS1

We began by loading and preparing the data. We visualized the data using scatter and histogram plots, examined attribute correlations, and inspected potential clusters using k-nearest neighbors (KNN). Principal Component Analysis (PCA) was used for identifying patterns and clusters.

3.2 DS2

The DS2 consisted of location-specific data files (Hungary, Switzerland, VA, Cleveland). Invalid or inappropriate data types were converted to {float, missing}. The first three datasets were combined to create the training set, while the fourth dataset (Cleveland) served as the test set. Visual inspection revealed a high volume of missing data in some location files. We followed the common practice of discarding columns with more than 40 percent missing values [1]. This resulted in the removal of three attributes from the training dataset, which was then imputed using KNN. The imputed training data was plotted using scatter and histograms. Both the training and test data were standardized to reduce scale differences between attributes. EDA continued with the calculation of the correlation between columns and an examination for potential clusters. Finally, PCA was performed to understand how the training data is projected into the new space.

3.3 Insights

Our exploratory data analysis indicated that the data includes both categorical and continuous variables and no highly correlated features or visually detectable clusters. This insight guided should choice of machine learning algorithms.

3.4 Extra note

We saved two versions of train and test datasets, one with binary labels (0=no disease diagnosed, 1=disease diagnosed) used in SVM and XGBoost scripts and the other with scaled attributes from 0 to 4 used in RandomForest-Regression and MultiRidge-Regression scripts.

4 Modelling

4.1 RandomForest Regression

For this file a lot of focus was put on tuning of the model. Here it was quickly realised how long it takes to find an ”optimal” model. For the sake of time the target for the tuning algorithm was decreased such that a model could be found in a reasonable time. The lowest Root Mean Square Error(RMSE) found on the training dataset was 0.906, and on the test dataset the same model got an RMSE of 0.998. A figure of the performance over the different folds can be seen in Figure 1.

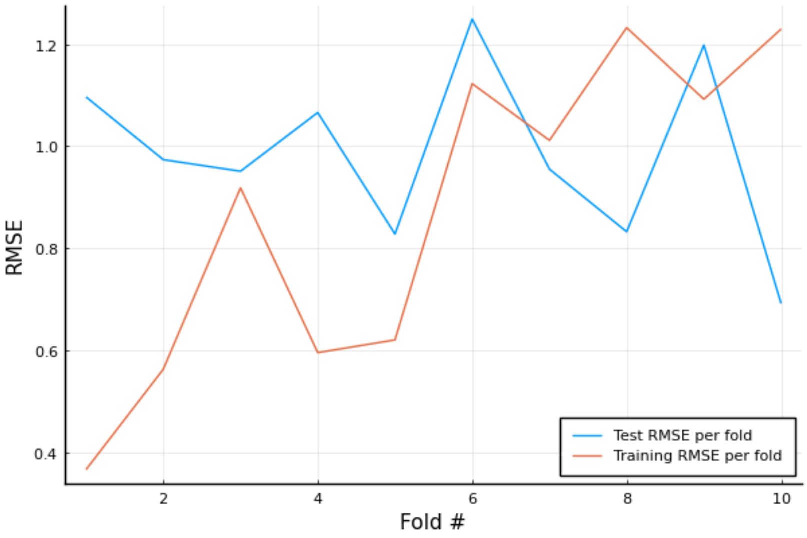

Figure 1: RMSE over folds for training and testing data

To further show the performance of different parameters a dashboard containing a contour visualization was made. Here a user can choose different hyperparameter to compare the performances of these parameter ranges. This helped further show the connection between parameters and the performance of the model. Then a dashboard was created with slider for the different hyperparameters for creating a custom model. By working with this custom model a bit, a model with similar performance to the tuned one could quickly be made, as many of the parameters are quite intuitive.

Furthermore, if the tuner was run for longer it would be harder to match it with the custom model.

4.2 Multivariate Regression

One of the regression analysis that was used was multivariate regression. For the first model, multivariate regression was employed. The model performed better on the training data compared to the test data. The root mean squared error for the training and test data were 0.8409 and 0.9957 respectively. The score difference between training and test is significant. A regularization method was utilized with the aim of enhancing the model's performance, with a focus on mitigating overfitting of the training data. Ridge regression was the regularization technique that was implemented. Ridge regression modifies the over-fitted models by adding a penalty term to the cost function. The goal is to improve the performance and stability of the regression model. Using the same training and testing data, ridge regression was applied. The training and test root mean squared error score were 0.8948 and 1.054 respectively. This demonstrates that ridge regression performed worse than multivariate regression. Another important aspect of the multivariate regression and ridge regression models are the coefficients of the features of both models. After obtaining the coefficients of each feature for both models, two line plots were created to show the magnitude of each coefficient. This is a useful tool since it allows the reader to gain some insight on what features have a stronger influence on the outcome. It's worth noting that certain features exhibited varying degrees of importance in one model when compared to the other. For instance, the fbs feature had more influence on the predictor outcome for the ridge regression model than the multivariate regression model (Figure 2). I expected for the coefficients of the features in ridge regression to be smaller than multivariate regression since the penalty term discourages large coefficient values. However, almost all the features for ridge regression had higher coefficients than multivariate regression.

4.2.1 Cross Validation

Moreover, cross validation of ten folds was utilized for multivariate regression and ridge regression. The mean of the ten root mean squared error scores of multivariate regression and ridge regression were 0.8578 and 0.8954 respectively. This demonstrates that multivariate regression is still performing better than ridge regression. A violin plot was utilized to show score comparisons for multivariate regression and ridge regression. The scores for the multivariate regression were less sparse than the ridge regression. This shows that multivariate regression is more consistent and stable. Lastly, a t-test was used to compare the both models. The t-test failed to reject the null hypothesis. This is enough evidence to prove that there is significant difference in the means between multivariate regression and ridge regression.

4.2.2 Cross Validation with Different Folds

To better understand the model for multivariate regression and ridge regression, cross validation with different number of folds was utilized. The motivation for doing this was to demonstrate the stability of the model's performance. Another motivation was to see which model consistently outperforms the other model across various fold sizes.

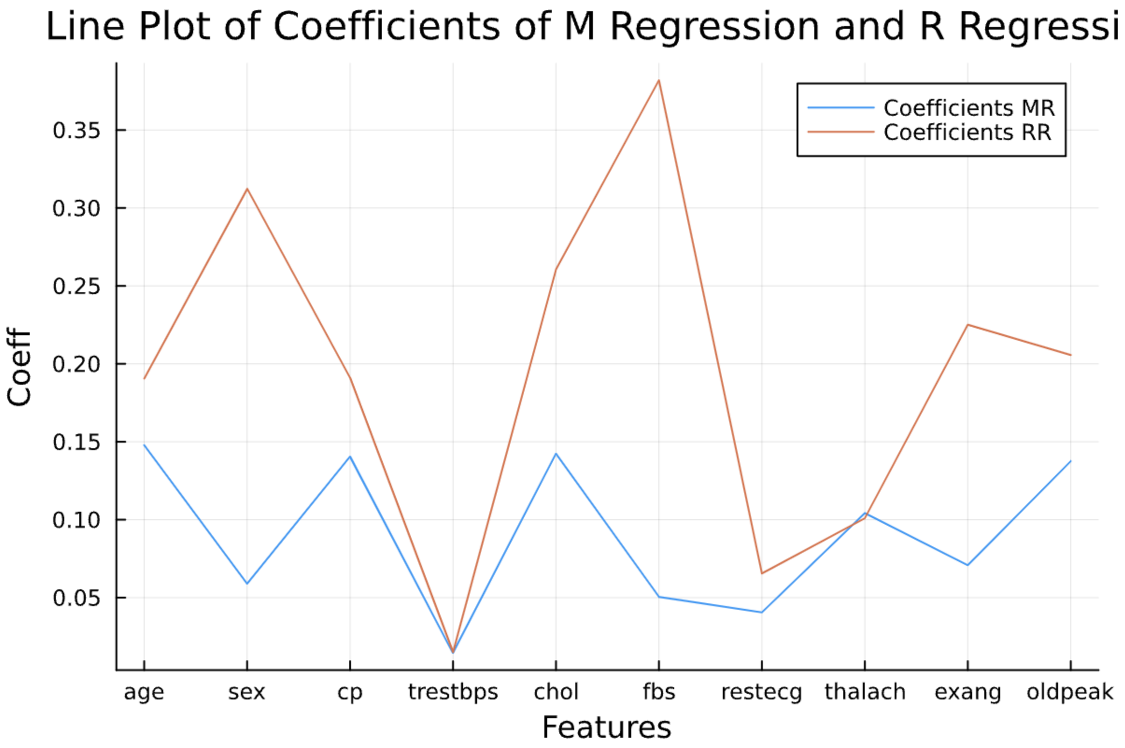

Figure 2: Coefficients of features in both Multivariate and Ridge Regression

After running both multivariate regression and ridge regression from 2 to 20 folds ten times for each fold, multivariate regression consistently outperformed ridge regression (Figure 3). The mean of the mean root mean squared error from 2 folds to folds 20 for multivariate regression and ridge regression were 0.6813 and 0.7781 respectively. This shows that multivariate regression model consistently outperformed ridge regression model.

4.3 Support Vector Model (SVM)

We initiated a primary SVM model and tested its train and test prediction performance across different values of degree and cost parameters (Figure 4). This provided insights into the general performance and a sufficient range to use in automatic tuning. The best model achieved a test accuracy of 74 and a train accuracy of 82 percent.

4.4 XGBoost

We also explored a binary logistic XGBoost algorithm. Manual searches were conducted to identify the best estimates of learning rate and maximum depth parameters, this time by using ROC curves. The model was then run with the best parameter values (Figure 5), and its performance on the test data was evaluated. Test accuracy of 72 and train accuracy of 87 percent suggest the possibility of overfitting compared to SVM results.

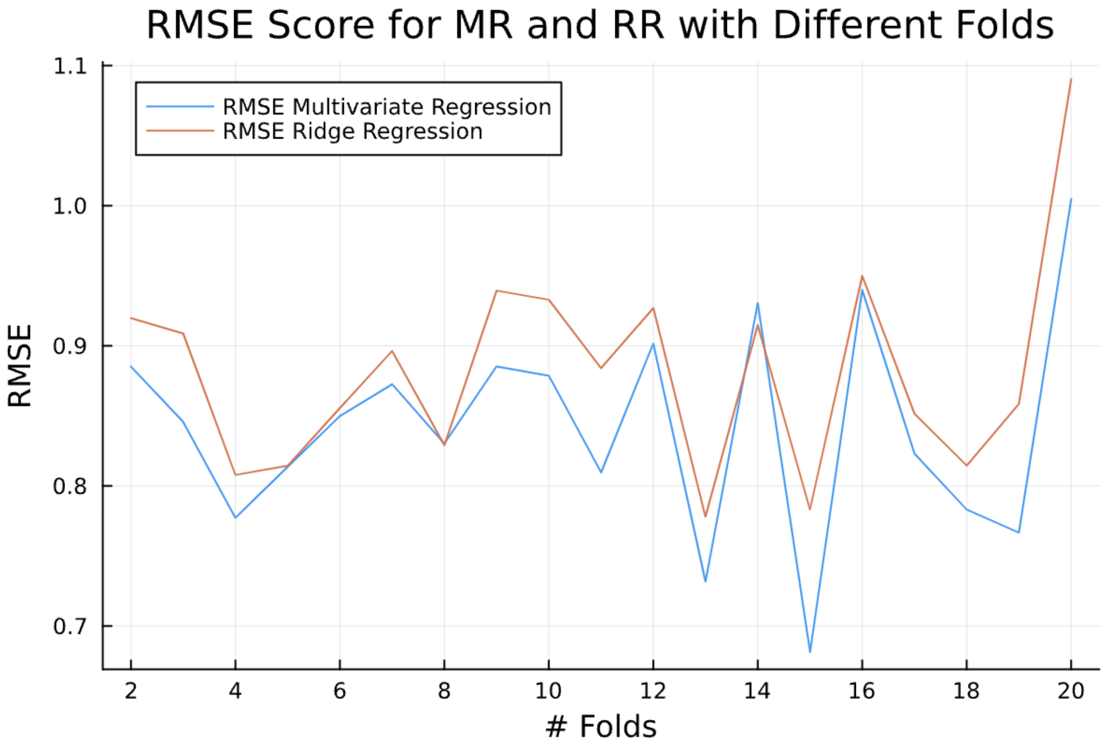

Figure 3: Multivariate Regression vs Ridge Regression across folds 2-20

4.4.1 Feature Importance

Using the XGBoost package, we assessed feature importance. Cholesterol level, age, and the type of chest pain were identified as important features, followed by activity-related heart rate and angina as shown in Figure 6. Interpret these results with caution due to the nature of features and how their importance is calculated.

5 Conclusion

In this project, we explored the data structure and took minimal pre-processing actions in data preparation. Next, we applied several models covering both binary classification and regression approaches and used different techniques to visualize and evaluate our models.

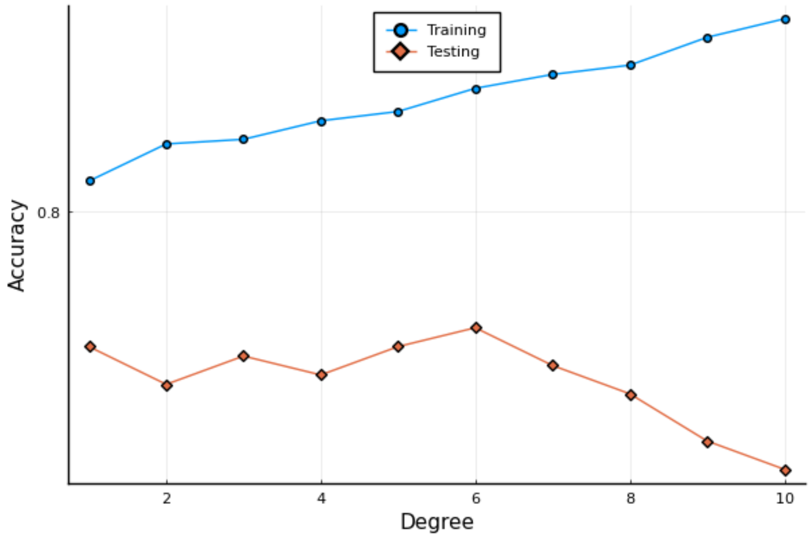

Figure 4: Accuracy over degrees for cost 0.1 for training and testing data

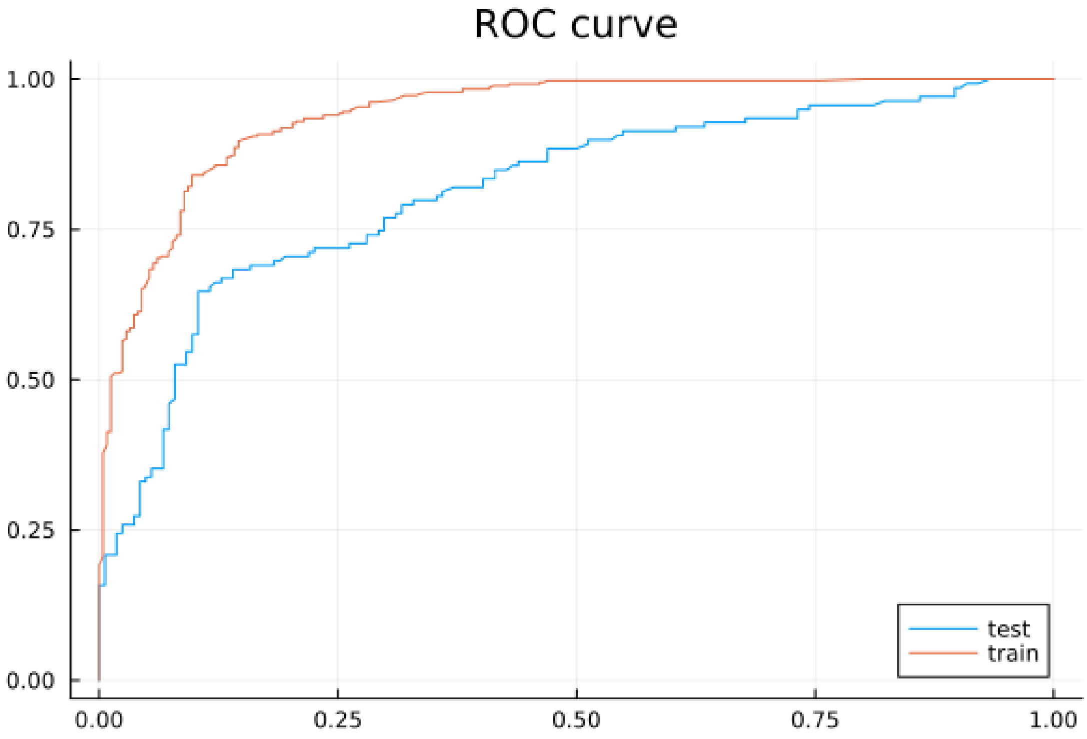

Figure 5: ROC curve for best parameters

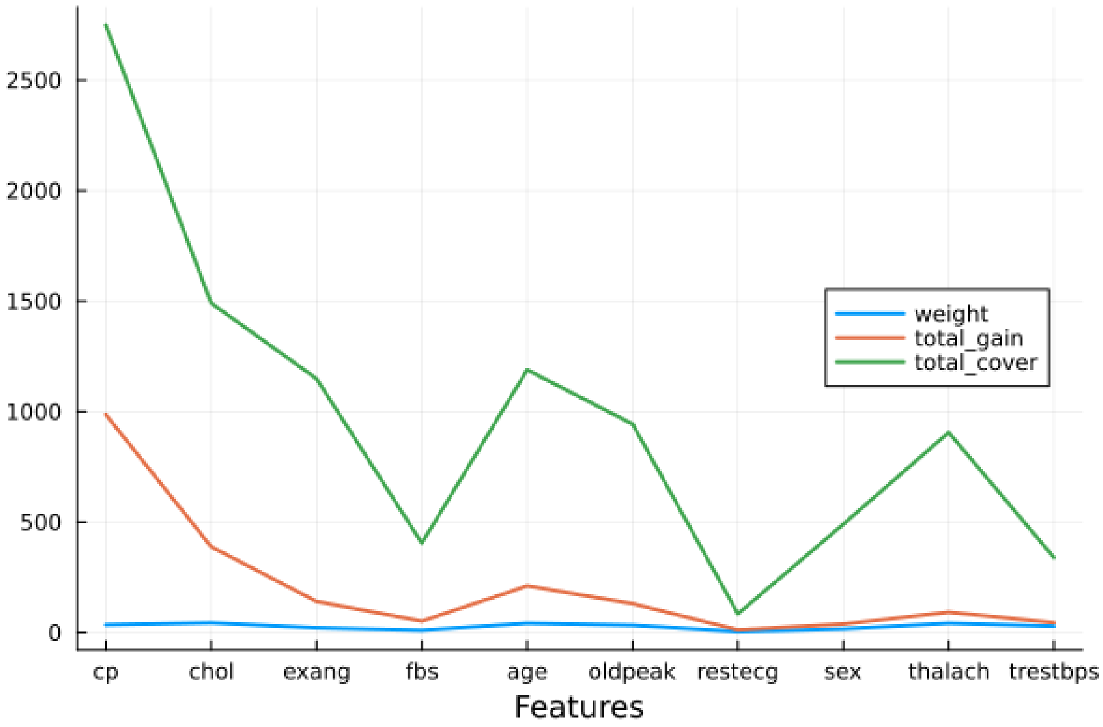

Figure 6: Feature importance metrics

6 Contributors

Although different tasks were assigned to each member, the three members actively exchanged ideas and helped with different sections of the project.

Farzane was the main contributor to EDA-Imputation, SVM-Binary and XGBoost-Binary scripts.

Morten was the main contributor to the RandomForest-Regression and Github management.

Alejandro was the main contributor to Multivariate and Ridge-Regression and report management.

References

[1] Paul Madley-Dowd, Rachael Hughes, Kate Tilling, and Jon Heron. The proportion of missing data should not be used to guide decisions on multiple imputation. *Journal of clinical epidemiology*, 110:63–73, 2019.