Emilio Padrón Molina

# Regression Report

For this report, I decided to work with the FIFA 19 dataset to generate regression models to predict the wage of a soccer player based on their age, position, preferred foot, and other attributes. The dataset was obtained from Kaggle (https://www.kaggle.com/karangadiya/fifa19), and contains information of all of the active professional soccer players in the world. Aside from their names, age, nationality and club, there are other specific attributes such as their overall score, potential, stamina, and specific abilities. This set has 18,207 rows (corresponding to specific players), and 89 columns (representing their attributes). It worth noting that hese attributes are both numerical and categorical

After importing the data and dropping the useless columns, it is necessary to deal with missing values. For this specific set, the 'Preferred Foot', 'Position', 'Height', 'Weight' columns don’t have many missing values, so one can just delete the observations that don't contain this info. Additionally, for the columns that represent specific abilities, the missing values could just be replaced by the mean of the column; nonetheless, this wasn’t necessary given that these observations coincided with the observations that were deleted previously.

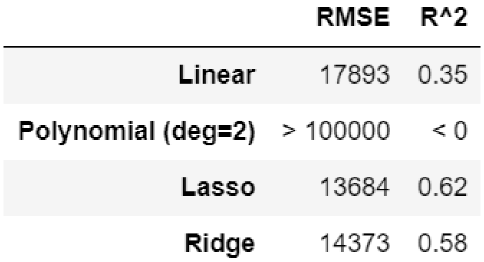
Furthermore, it is relevant to check the data types of each column. In this specific case, it can be seen that the 'Wage', 'Height' and 'Weight' columns are considered as objects rather than numbers; therefore, it is necessary to change them to a numerical type. Similarly, for the attributes regarding abilities, given that their values correspond to integers, it is required to switch from a float64 type to an int64 type.

Before starting to do the final feature engineering process, it is necessary to drop the rows of player that don’t have a wage. This is mainly because the are outliers with respect to the rest of the data. After doing this, the shape of the set can be analyzed once again; this results in 17912 rows/observations and 42 columns that represent different features and targets (most of them relating to specific soccer attributes such as ball control or defending). This new dataset only contains two categorical columns: “Preferred Foot” and “Position”. Both of these features need to be encoded; for this, the OneHotEncoding object from Scikit-learn can be put into use. As a result, the two previous columns were transformed in 27 new ones (already considering the deletion of one column per category to avoid collinearity).

Next in line, the data can be divided into a training-testing split of 70-30 respectively to be used for building the regression models. Even though the target variable doesn’t necessarily need to be normally distributed, it sometimes helps in the accuracy of the model; therefore, the y_train variable can be checked for normality. Given that its distribution was not normal, several transformations (log, boxcox, square root, inverse) can be applied to it as an attempt to improve the quality of the target. These transformations, however, appear to be unsuccessful in normalizing y_train, so one should just proceed.

At this point, the models can start being implemented with the help of pipelines. For the first model, the StandardScaler and the LinearRegression objects can be introduced in the pipline to fit the X_train and y_train data. Furthermore, for the second model, one can copy the first model and add polynomial features of degree two (given the number of columns, memory runs out for higher degrees) in the middle. Then, for the third model, it is worth to use a regularized model, in this case Lasso, and implement the GridSearchCV object to find the best alpha parameter (in this case alpha=400). Moreover, a similar algorithm can be coded using Ridge to have a fourth model. For this last model, the regularization parameter is alpha=4600.

For each of the four models, the "predict" method can be implemented using the X_test data to compare these results to y_test, and obtain the R^2 score and Root Mean Squared Error (RMSE). These results were the following:



| | RMSE | R^2 |
|---|---|---|
| **Linear** | 17893 | 0.35 |
| **Polynomial (deg=2)** | > 100000 | < 0 |
| **Lasso** | 13684 | 0.62 |
| **Ridge** | 14373 | 0.58 |

This table shows that the best regression model for predicting the Wage of a soccer player based on their age, height, weight, and specific attributes, is the Lasso regression with a regularization parameter of 400. Nevertheless, it is worth noting that the Ridge regression with a regularization parameter of 4600 is a close second. This can be noted given that they coincide in having the lowest RMSE and highest R^2 score. I believe that the prediction power favors the Lasso model because of its tendency to do feature selection; i.e., the probability of nullifying the coefficients in the regression.

Even though the results of the two last models are satisfactory, there are also actions one can take to further improve them. For starters, I believe that it would be interesting to see the impact of transforming the age variable into bins and treat them as categorical variables rather than numerical ones. Additionally, one might be able to get better models after achieving to normalize the y_train data. Finally, one could improve the prediction power by taking advantage of the Recursive Feature

Elimination approach to force the model in only using the most important features of the set.