Soccer: Analysis & Prediction
MIRI - Machine Learning (ML)
20th June, 2018

Martin Galajda
Florian König
Aitor Ortiz de Latierro

1 Description

Soccer is the most popular sport in the world and, not surprisingly, we have always been interested in that. Aside from the sport beauty many interesting questions quickly arise when you talk about it like: "*Will my team win today?*" or "*Is my favorite player good?*".

In this project we try to answer some of these questions by creating many different machine learning models using a public dataset.

1.1 Available Data

We use the **European Soccer Database** from the popular data science platform Kaggle [1]. It is a very complete dataset containing data from many European leagues with:

- 25000+ matches data from 2008-2016 seasons
- 10000+ players data
- betting odds from 10 houses
- FIFA players and teams statistics

Following the given initial feedback we have constrained the project to only this dataset in order to avoid spending all time on feature preprocessing.

1.2 Goals

We set 3 main objectives for the project, each with a different modeling:

1. result prediction → **multiclass classification**, home team wins, ties or loses?
2. goals prediction ⟶ **regression**, how many goals will a team score?
3. player analysis ⟶ **unsupervised**, what can we say about players?

With this *common-sense* questions about the data we can cover many of the topics studied in the course as well as satisfy our soccer-knowledge desire.

2 Previous Work

Sports outcomes prediction has been a popular topic since ever. Aside from its intrinsic difficulty and its entertainment setting developing good models has really strong economic implications due the popularity of betting houses [4]. Moreover, with the current *Big Data* trend many teams are trying to improve its tactics with machine learning [3].

A recent example is `Prediction of the FIFA World Cup 2018` by Andreas Groll [2] that combines random forests, ranking methods and Poisson regression models.

Furthermore, the author of the dataset claims that his best model (SVM) achieves 53% accuracy for predicting results (W/T/L) that are used by betting houses.

3 Data Exploration

In this section we will explain all the necessary data preparation and cleaning as well as feature selection and extraction. Although it may be one of the least interesting parts in a machine learning pipeline it is fundamental and often underestimated.

In our project it's even more important for two reasons, our raw data is very far from having predictive features and one of our objectives was *exploring* players data!

3.1 Database Conversion

Data that we acquired for building our models was available in form of dump from a `SQLite` database. In order to be able to work with data effectively in the R environment, we had to transform it to a more convenient format. We chose to save it as a CSV file.

3.2 Feature Extraction

One of the biggest and well-known difficulties in our project has been the lack of good features. The original dataset is just a database with match results and statistics, it's not designed for predicting results or numbers of goals right off the bat.

Our available raw data available for a given match can be summarized as:

- **results** → how has the home/away team performed in the past
- **goals** ⟶ how many goals it has received and scored
- **events** → shots, cards, possession and secondary metrics

In the end we couldn't find use of the secondary statistics, they didn't make a difference in our initial models, but we were creative with the two other feature categories.

3.2.1 Attack Strength

For every match and playing team we have computed its composite ***attack strength***.

First of all we will define ***attack scores*** (S^{team}_{attack}). For a given match we compute the average number of goals scored by the home team in previous home matches and in previous matches that has played in general and relate them using Equation 1a. Similarly using Equation 1b we define attack score of the team playing away.

$$S^{homeTeam}_{attack} = \frac{GS^{homeTeam}_{home}}{GS^{overall}_{home}} \quad , \quad S^{awayTeam}_{attack} = \frac{GS^{awayTeam}_{away}}{GS^{overall}_{away}} \tag{1}$$

where:

$GS^{homeTeam}_{home}$ = average number of goals scored by team playing home when playing home

$GS^{overall}_{home}$ = average number of goals scored by all teams when playing home

$GS^{awayTeam}_{away}$ = average number of goals scored by team playing away when playing away

$GS^{overall}_{away}$ = average number of goals scored by all teams when playing away

We can define ***defense scores*** ($S^{team}_{defense}$) analogously by replacing number of scored goals with number of conceded goals.

$$S^{homeTeam}_{defense} = \frac{GC^{homeTeam}_{home}}{GC^{overall}_{home}} \quad , \quad S^{awayTeam}_{defense} = \frac{GC^{awayTeam}_{away}}{GC^{overall}_{away}} \tag{2}$$

where

$GC^{homeTeam}_{home}$ = average number of goals conceded by team playing home when playing home

$GC^{overall}_{home}$ = average number of goals conceded by all teams when playing home

$GC^{awayTeam}_{away}$ = average number of goals conceded by team playing away when playing away

$GC^{overall}_{away}$ = average number of goals conceded by all teams when playing away

Finally, we can use the previously defined scores to compute attack strengths ($AStr^{team}$) for all matches as:

$$AStr^{homeTeam} = S^{awayTeam}_{defense} * S^{homeTeam}_{attack} * GS^{home}_{average} \tag{3}$$

$$AStr^{awayTeam} = S^{homeTeam}_{defense} * S^{awayTeam}_{attack} * GS^{away}_{average} \tag{4}$$

where

$GS^{home}_{average}$ = Average number of goals scored by home team when playing home

$GS^{away}_{average}$ = Average number of goals scored by away team when playing away

3.2.2 Win Ratio

Another features that were extracted from data were related to winning ratio - ratio of matches won computed based on $\boldsymbol{k}$ previous matches. Specifically, we chose to extract information about:

- overall winning, tie ratio of home team
- overall winning, tie ratio of away team
- winning ratio of home team playing home, away
- winning ratio of away team playing home, away
- tie ratio of home team when playing home, away
- tie ratio of away team when playing home, away

We also tried to use different strategies to compute different types of winning ratios:

- taking tie as half win and not computing tie ratios separately
- computing tie ratios separately and using them as separate features

During evaluation of our models, we were using different k for computing ratios and we chose the best one by performing cross-validation.

3.3 Player Analysis

Independently from the matches we created a CSV table with features of the different players. By crossing different databases from the `SQLite` dump we obtained the following information for every player/season available:

- **personal info** → name, age, height and weight
- **FIFA info** ⟶ points given to the players in attack, defense, speed...
- **matches stats** → total number of appearances and goals

There are many anecdotal facts that can be quickly answered with this information like how tall is the tallest player (Kristof van Hout, 208cm), how many goals has scored the top scorer (Messi, 295)... We will focus though on some other non-trivial results.

Correlations

Thanks to the FIFA valuations We have many information in different soccer areas like attack (accuracy, dribbling, shot power...), defense (tackling, strength, marking...) and others (vision, goalkeeping, aggression). We think it may be very interesting to find relationships between them as well as with the other features.

We can obtain many interesting details from Figure 1 like that the most influential feature to a player value is its reaction time (an important feature for all players) or how constitution players compensate fully agility and technical qualities with pure strength.

This kind of facts can be useful players, trainers and even for predicting models.

Clustering

Another interesting analysis would be to see what natural clusters emerge from the data. We have used the simple **k-means** algorithm repeating it $N = 100$ times to find good local optimums and deciding the best k with the studied Calinski-Harabasz index.

In Figure 2 we see that $k^* = 3$. Looking at the cluster centers we can distinguish:

- **strength-focus** → tall and heavy players
- **agility-focus** ⟶ small and agile players
- **good strikers** ⟶ many games and goals scored, well-valuated

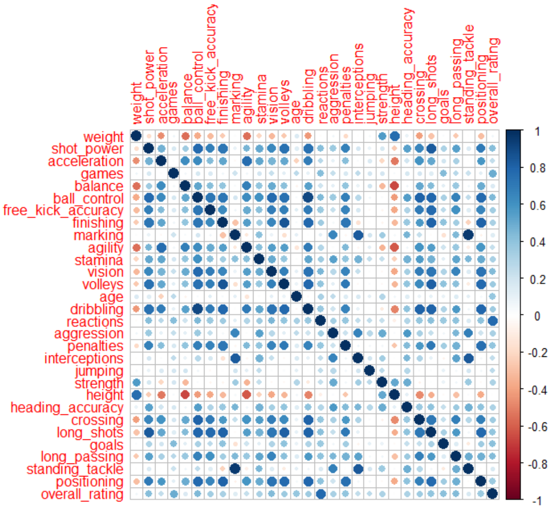

Figure 1: attributes correlation matrix

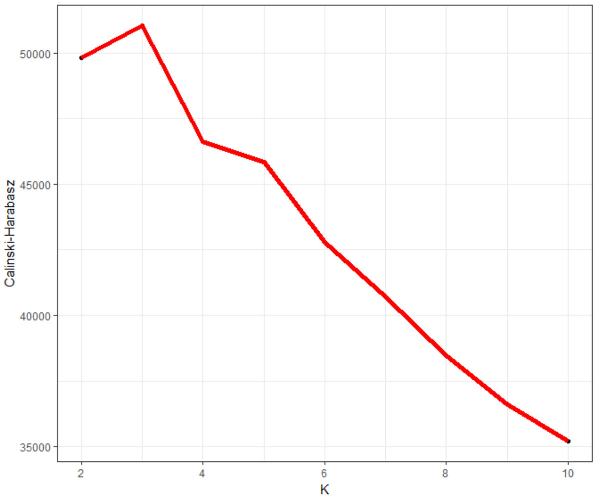

Figure 2: Calinski-Harabasz index

4 Modeling & Validation

In this section we will explain what models we have used for match result and # of goals prediction as well as which methods have we used for selecting features and parameters.

4.1 Result Models

For predicting match results we have considered and compared multiple models:

- Linear Discriminant Analysis (LDA)
- Quadratic Discriminant Analysis (QDA)
- K-Nearest Neighbors (KNN)
- Multinomial Logistic Regression (LR)
- Multiclass Support Vector Machines (SVM)

4.2 Goals Models

For predicting number of goals we have considered and compared multiple models:

- Generalized Linear Model - Normal & Poisson (GLM)
- Random Forests (RF)
- Nearest Neighbors Algorithm (KNN)

4.3 Feature Selection

For determining best subset of features for given models we have used 10-fold cross-validation technique. Specifically, we have implemented function which tries all possible combinations of features for building model and computes cross-validation error for every combination. This was very exhaustive search which allowed us to determine best subset of features. However, it was very computationally intensive and for models like SVM, it was very impractical. Because of that, we have developed second technique which starts with all features and greedily tries to remove one feature from prediction model and then it observes change in cross-validation error. We greedily remove features from the model as long as CV-error reduces (we use the biggest reduction of CV-error in every iteration). Using this techniques, we were able to determine best subset of features for predictions for each model.

We tried to compute CV-errors using different models to evaluate how many previous matches for the match to use for computing win ratios (to extract features for predicting match result). Specifically, we compared winning ratios from previous 3 and 10 matches. The better results were achieved by using previous 3 matches and that is what we ended up using. However, more exhaustive search for the right number of previous matches to use (trying all numbers between 3 and 10) could improve our models, but the time required for computations was out of the scope of our project.

For logistic regression we have performed comparison using exhaustive and also greedy cross-validation approach for different subsets of features. The results obtained from exhaustive cross-validation can be observed in Figure 3. We obtained lowest CV error using different subset of features than what we chose to use in the end. The reason for that is that common sense told us that the features that we chose to use might be more appropriate and the differences between our chosen subset and subset which achieved lowest CV-error were very small. Moreover, 10-fold cross validation choses partitions randomly and we didn't have enough resources to run computations multiple times to reduce effect of randomness, which could introduce small bias in determining best subset of features. That led us to believe that the differences in errors between using these two subsets of features are negligible and using the subset which makes more sense for us would be better option. The exhaustive search confirmed that best subset of features was achieved by:

- attack strength of home team

- attack strength of away team
- overall win ratio for home team
- overall win ratio for away team
- win ratio for home team when playing home
- win ratio for away team when playing away

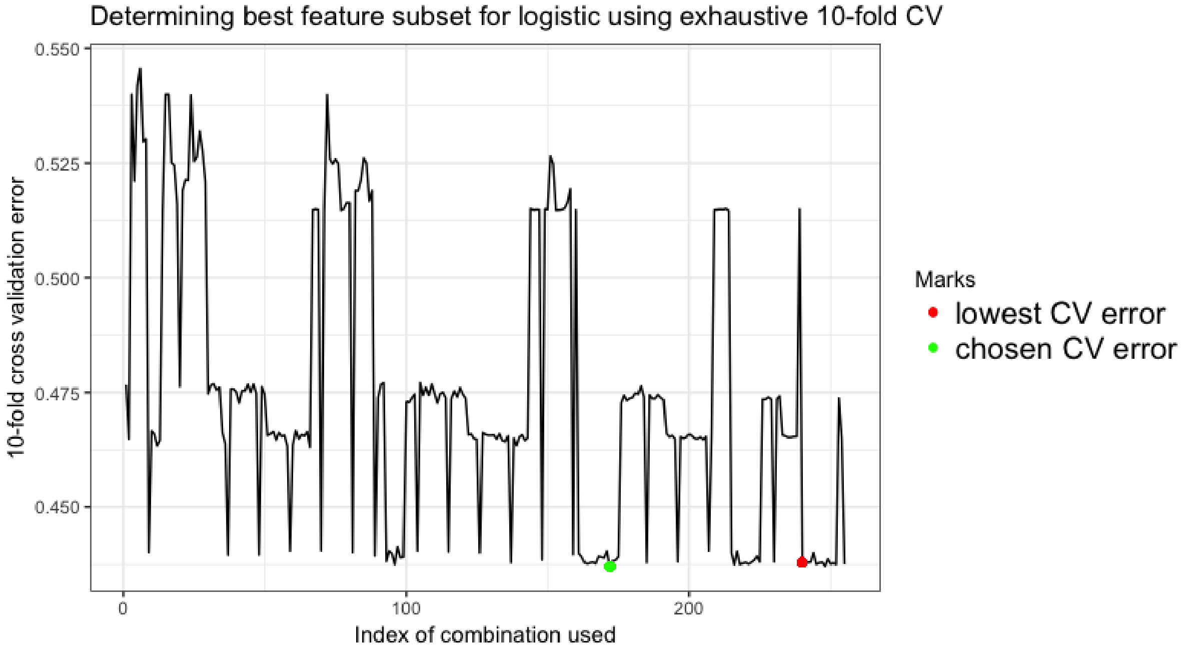

Figure 3: Exhaustive Cross-validation for logistic regression

The greedy cross-validation of different subset of features achieved very similar results. Moreover, for LDA, QDA and KNN, the results from cross-validating best subset of features for predicting were very similar and we ended up using the same subset. However, for SVM, using greedy approach with cross-validation, we determined different subset of features which logically also made sense and included smaller number of features (which could potentially reduce chance of overfitting):

- attack strength of home team
- attack strength of away team
- overall win ratio for home team playing home
- overall win ratio for away team playing away

The results from greedy cross-validation for SVM can be observed in Figure 4.

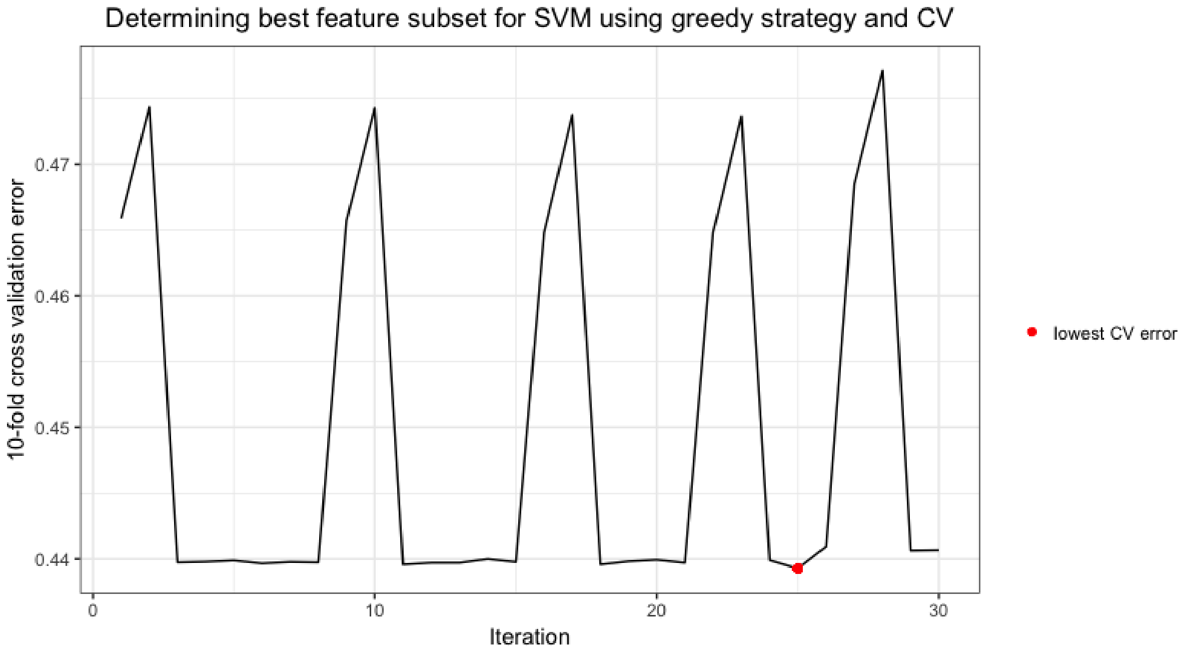

Figure 4: Greedy cross-validation of features for SVM

4.4 Tuning parameters for SVM

For tuning parameters for SVM, we have used 10-fold cross validation. We have compared cross-validation errors for different values of C: 1, 51, 101, 151, 201, 501 and 1001. Moreover, we have tried 3 different kernels: rbf, polynomial and vanilla kernel. The results obtained can be observed in Figure 5. We can see that SVM with rbf kernel and C set to 1 achieved lowest CV error.

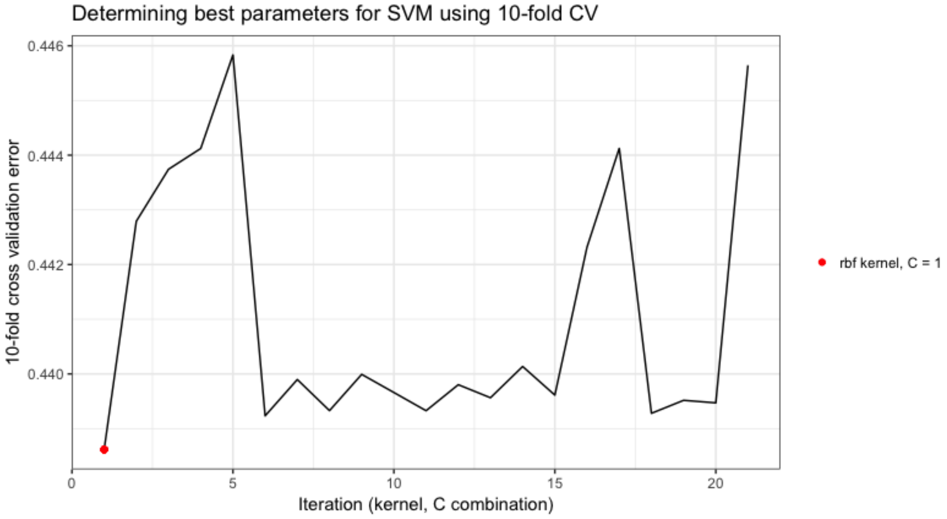

Figure 5: Tuning parameters for SVM

4.5 Tuning parameters for KNN

For tuning k parameter for KNN, we also used 10-fold cross validation. We compared cross-validation errors for different values of k, starting from $k = 1$ and ending with $k = 401$, with a step size equal to 5. The results obtained are shown in Figure 6.

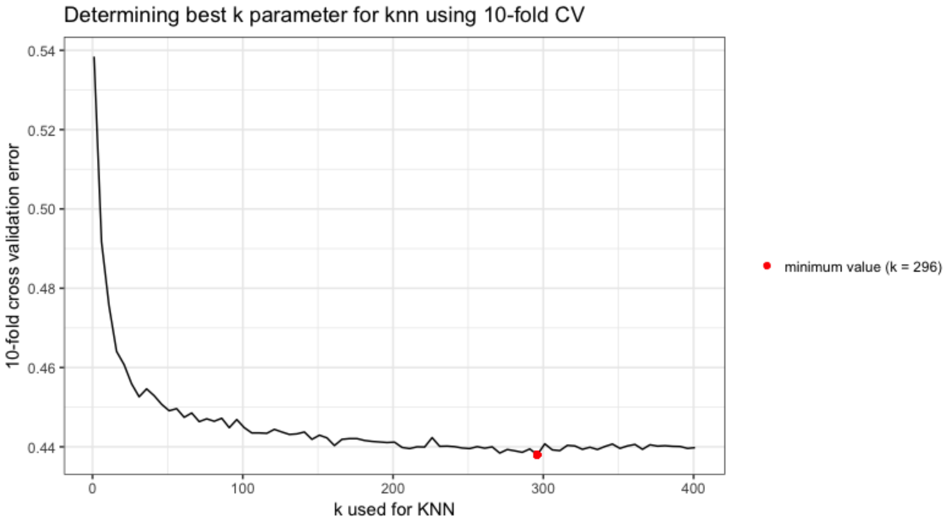

Figure 6: Tuning k parameter for KNN

4.6 Computing generalization error

For computing generalization error we have chosen two approaches. The first approach consisted of randomly choosing 20 % of data for computing generalization error and the rest for training the model. However, this approach could introduce a small data leakage as we are computing attack strength ratios for teams using all matches (which means that we are using information from future). For that reason, we also compute generalization error by using all matches from last season in dataset for determining test error and all other matches for training the model. From these two errors, we choose the higher error to determine the generalization error for our predictions.

5 Results

In this section we will compare the models accuracy and estimate its generalization error.

5.1 Match Results Prediction

In Figure 7 we show the cross-validation errors for each model. We can see that they all have close performance with QDA standing out as the worst with 45% error and multinomial LR as the best performer with 43.8% error.

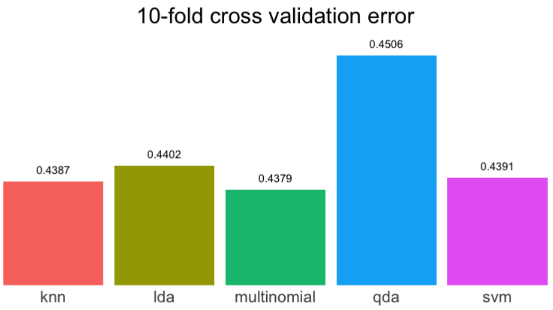

Figure 7: Determining best model

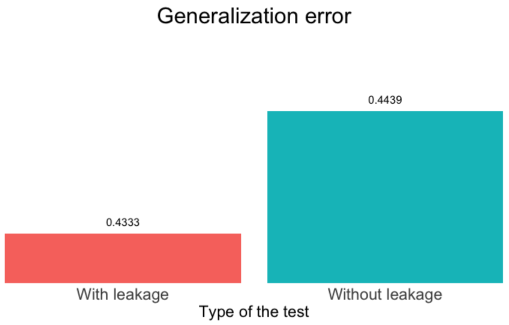

Figure 8: Determining generalization error

In Figure 8 we can see the performance of the best model in the two test sets approaches. As expected the model has worse (+ more real) accuracy using the bigger test set with leakage than with the other approach. With a 55.6% accuracy of our model we can state that our model performs better than the model of the author of dataset, SVM with 53% accuracy.

5.2 Goals Prediction

We show in Table 1 the **generalization without leakage** metrics of the different tested models a including *best-prior* baseline for reference. We can see that the best result is obtain with a simple linear regression! Non-linear models are less robust in the end.

Model	MSE	NRMSE	Acc. (%)
BL	1.54	1.00	33.4
GLM-N	**1.17**	**0.88**	**37.6**
GLM-P	1.21	0.90	36.5
RF	1.22	0.90	35.2
KNN	1.19	0.89	34.8

Table 1: # goals generalization metrics

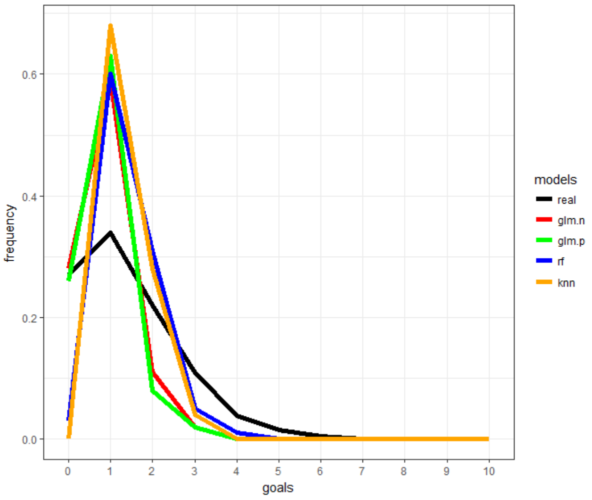

Figure 9: predicted goals statistics

In Figure 9 we also show the # goals probability distribution for all models. We can see that models are *cautious* over-representing 1 goal. An interesting insight is how different RF and KNN are, almost never predicting 0 goals and still having good metrics.

6 Future Work

We're aware that this project is just a stepping stone in the huge statistical field of sport outcomes prediction. There are many parts that could have been extended and improved with more **computation**, **data** or **expertise** out of the course scope.

Some of the models feature selection could have been made more exhaustively avoiding greedy approximations or feature reusing. Extending some of the model hyperparameters searches and creating ensembles would also allow a performance boost.

Because of project constraints we only used a single dataset to perform all modeling. By using multiple data sources, e.g. international tournaments, more robust and complete prediction systems can surely be developed (like the ones used in real betting houses).

The last but not least factor would be use more domain knowledge about players and teams to combine it smartly with matches data (e.g. use Section 3.3 data better).

7 Conclusions

Our best result predicting match results has been 55.6% accuracy with the multinomial logistic regression model. This is a very good and encouraging result that surpasses even the dataset author claimed best result of 53%.

Our best model predicting number of goals has been a simple linear regression with 0.88 NRMSE and 37.6% accuracy. As we expected this was also a very difficult problem.

Finally we have also done a brief exploration of the players data obtaining some non-trivial and curious facts about how the different skills and biometrics are interrelated.

References

[1] Hugo Mathien https://www.kaggle.com/hugomathien/soccer Kaggle

[2] A. Groll, C. Ley, G. Schauberger, and H. Van Eetvelde. Prediction of the FIFA World Cup 2018 - A random forest approach with an emphasis on estimated team ability parameters. *ArXiv e-prints*, June 2018.

[3] L. Pappalardo and P. Cintia. Quantifying the relation between performance and success in soccer. *ArXiv e-prints*, May 2017.

[4] Stefan Luckner, Jan Schröder, and Christian Slamka. On the forecast accuracy of sports prediction markets. In Henner Gimpel, Nicholas R. Jennings, Gregory E. Kersten, Axel Ockenfels, and Christof Weinhardt, editors, *Negotiation, Auctions, and Market Engineering*, pages 227–234, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.