FIFA Player Rating

Gopalakrishnan V, Vibhav Agarwal, Varada Desikan P S

Abstract

Given a dataset consisting of all the players in FIFA 18, this algorithm thrives to predict how good a player is overall and how efficient he is at each role. Visualizations help provide a better insight about multiple relations as well.

I. Introduction

FIFA is an addictive game. Football fans all around the world are always ready to beat their peers in a friendly game of FIFA. Fan guys and girls analyze the change in data every week and each player's performance to build the perfect team and to become a perfect manager in the FIFA world. But given the details of attributes of a player, how good is he? Where does he best fit in the team?

II. Problem Statement

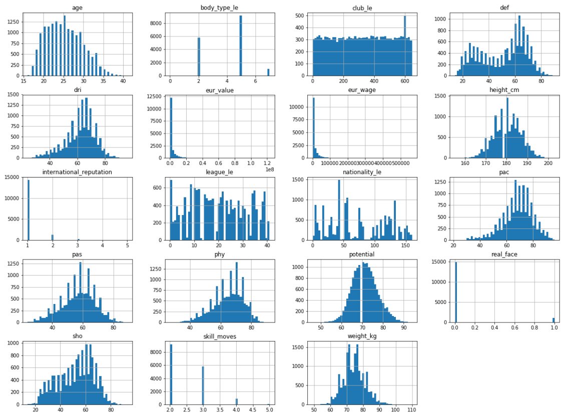
Predict the overall attribute of a footballer given the player characteristics, skills and other specialties. Apart from the final overall, provide a heuristic on how the player will play at different positions.

III. Dataset

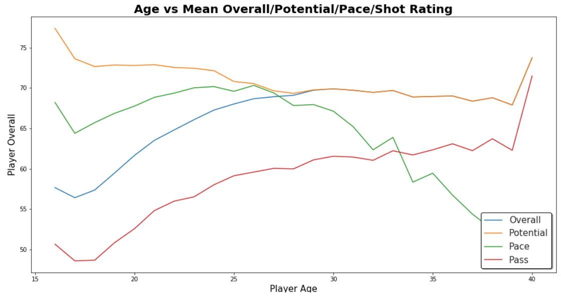
Public dataset from *Kaggle datasets* section. The dataset contains the information of FIFA 18 players consisting of their personal attributes, their current club and nationality, playing and performance attributes, preferred positions and rating at each position, their value and wage, their traits and reputation, their potential and overall rating.
The dataset consists of around 18,000 players with 185 columns. The overall ratings range from 46 to 94.

IV. Visualization

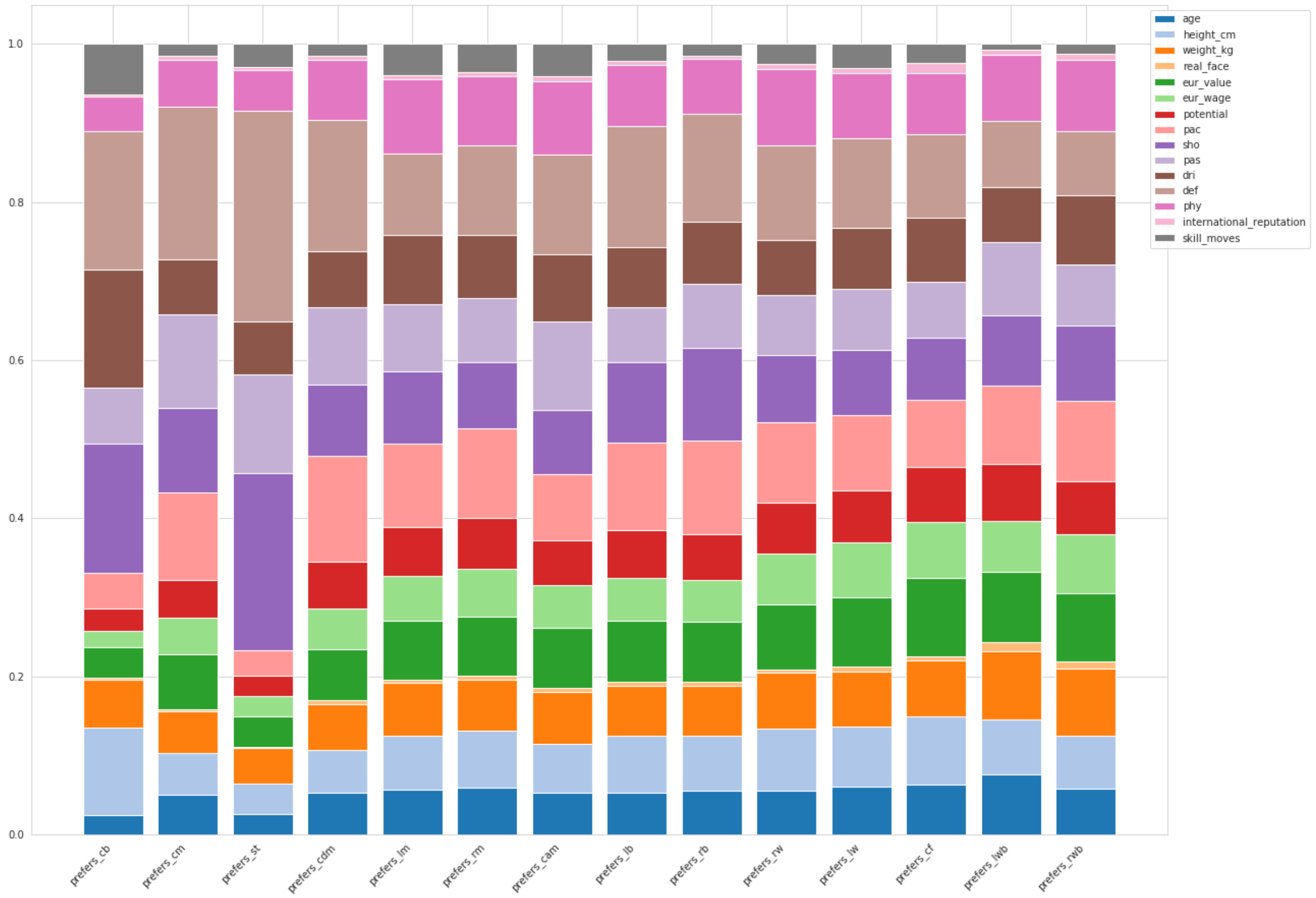
1) The importance of this histogram is that it reveals how the data is grouped together so that it can be compared and analyzed.



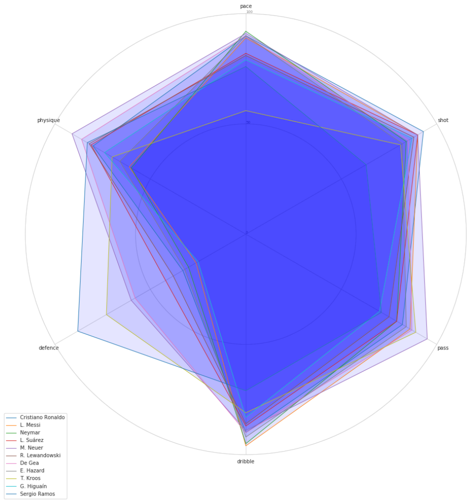
2) Compare the mean overall rating and the players age on different parameters like potential, pace, passing score. As expected, the overall increased with increase in age while the potential decreased with the increase in age. It can also be observed that the potential and the overall rating converges at some point after which we can't notice much difference between the two. Pace decreases drastically as the person ages while the stats like passing declines much slowly.



3) Analyze the feature importance for different positions.
We assumed that the defenders were dependent on defending attributes more than the others like "shot power", "dribble", etc. But, a random forest classifier showed us that our hypothesis was not accurate.
4) Perceive the skill attributes like pass, pace, dribble etc of the top 10 overall rated



players in FIFA 18. The information is provided in the form of a radar/web.



V. Preprocessing

NULL values were found in the columns 'club' and 'league'. To mend this, we set these NULL value to be "Unknown". In the position columns, it was noticed that the NULL values occurred only if the player was a goal keeper. Hence we removed all the goal keeper rows. As we don't have the rows pertaining to goal keepers, we also removed the associated column 'gk' from our dataset.

On our further analysis, it was noticed that a few outliers were noted in the 'body-type' column. To handle this, we set the 'body-type' attribute to an appropriate value (based on our domain knowledge). Considering that the evaluation metric composes of Mean Squared Error (MSE), this process would definitely help the results.

VI. Feature Engineering

As a part of dimensionality reduction, we decided to merge all the trait / speciality columns into one. To achieve this, we transformed the Boolean present in these columns into numeric counterparts and found their respective sums. But, the resulting model didn't perform to our expectations. To handle the unique and categorical string attributes a Label Encoder was used. This would allow us to handle string features, without increasing the dimensionality of the problem. To transform 'height' and 'weight' into a single column, we computed the 'bmi' value as a feature. But to our surprise, it didn't help our cause much.

Based on the correlation plots and our domain knowledge, we decided to pick the following features : club-le, real-face, age, league-le, height-cm, weight-kg, body-type-le, nationality-le, eur-value, eur-wage, potential, pac, sho, pas, dri, def, phy, international-reputation, skill-moves (where **le** stands for *label encoded*)

VII. Evaluation Criteria

R^2 is statistical measure of how close the data is fitted to the regression line. It is also known as the coefficient of determination or the coefficient of multiple determination for multiple regression. It is the percentage of the response variable variation that is explained by a linear model.

$$R^2 = \frac{\text{Explained variation}}{\text{Total Variation}} \quad (1)$$

R^2 is always between 0 and 1. In general, the higher the R^2, the better the model fits the data.

The Mean Squared Error (MSE) is a measure of the deviation of the estimates from the observed columns. MSE is always non-negative, and a value of 0 (almost never achieved in practice) would indicate a perfect fit to the data. In general, a lower MSE is better than a higher one. Note that MSE is sensitive to outliers.

VIII. Model Experiments

The set of target features : overall, rs, rw, rf, ram, rcm, rm, rdm, rcb, rb, rwb, st, lw, cf, cam, cm, lm, cdm, cb, lb, lwb, ls, lf, lam, lcm, ldm, lcb. The dataset was further split into train and test in 70:30 ratio.

1) **Multi-Output Regressor with Gradient Boosting Regressor (M1)** : Multi-target regression is a strategy of fitting one regressor per target. This is a simple strategy for extending regressors that do not natively support multi-target regression. In each state of Gradient Boosting, a regression tree is fit on the negative gradient of the loss function.
2) **Decision Tree Regressor (M2)** : Decision tree is used to fit a sine curve with additional noisy observation. Since the model is being run on the default parameters the Decision Tree tends to take the max_depth possible and therefore tends to overfit. Therefore there's a difference between the cross validation score and the predicted mean squared error.
3) **RandomForestRegressor (M3)** : A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting.
4) **PCA + Multi-Output Regressor with Gradient Boosting Regressor (M4)** : PCA is being used for dimensionality reduction on the data points and those reduced labels are then fed as an input to the Multi-Output Regressor. The player attributes are likely to be be linearly related to each other (for example agility and sprint speed), so a PCA should do a good job of compressing the data while retaining information. But to our suprise this model doesn't perform as good as the Multi-Output Regressor with our own hand picked features.

Models	**Cross Val**	**MSE**	**R2**
M1	1.5027	**1.5053**	**0.9835**
M2	5.9546	5.5473	0.9426
M3	2.2127	2.0423	0.9789
M4	1.5027	1.8596	0.9800

IX. Conclusion

The model successfully predicts the overall rating over the test data at an accuracy of **0.9835**. The future development is to implement the same algorithm over the goalkeepers as well.

X. References

1) https://www.kaggle.com/harmeggels/fifa-18-player-characteristics-per-position