# Can we use video games data for real-life data analysis?

Nishchal Dethe
UNI : nd2506
Abhishek Jindal
UNI : aj2708

# Contents

# 1 Introduction

Videogames are now a ubiquitously accepted source of entertainment in the society. Some popular games like EA Sports' FIFA have managed to create an amazingly accurate simulation of an actual football match. They have a whole bunch of data being utilized in the game engine, ranging from player attributes to team formations and tactics. Although, all this data serves to make the game more enthralling for end-users, can this data be utilized outside the game? Is this data useful in data analysis in the real world? Does this data have any correlation with how matches between two teams actually pan out? And above all what skills have the most impact on the outcome of the game? In our work, we try to answer these questions and evaluate the validity of the FIFA data in real life predictive tasks. We utilize some match data of actual football matches between teams in the popular European soccer leagues till the year 2016 and some data on player attributes borrowed from the FIFA game. We then try to answer the above questions by trying to find the correlation between the match outcomes and the player attributes of the teams on the pitch. By utilizing the player attributes of the playing teams, we are able to predict the outcome of the match with nearly 70% accuracy. We therefore conclude that the player attributes provided by the FIFA game are highly correlated with the outcome of real-life matches and can be useful predictors for games and perhaps even intra-game events.

# 2 Dataset and Pre-Processing

## 2.1 Dataset

We utilize the European Soccer Database provided on Kaggle for free public use. The data consists of statistics scraped from the FIFA(the world governing body for soccer) website and the FIFA video-game(created by EA Sports).

The dataset is more than 1GB in its uncompressed format (SQLite) and has separate tables for player attributed (extracted from the game) and match statistics (extracted from actual matches). The player attributes consist of the rows listed in Table 1.

The matches table has information from around 31,000 matches played in popular European Leagues like the English Premier League, Spains La Liga etc. It contains fields like the x-y coordinates of the starting formation of the 11 players from the home team, the x-y coordinates of the starting formation of the 11 players from the away team, the ids of the 11 players from the home team, the ids of the 11 players from the away team and match events like fouls, goals, penalties, corners, yellow cards, red cards etc. We didnt utilize the match events as it gave us too much information about the match and didnt help us in evaluating the usefulness of the player attributes alone in predicting the outcome of the match.

They are more tables for players, having the players name, the players date of birth, and some other personal data. There is also some more information about the European leagues ( we have not utilized this information in our current work). There was also some information

about the teams themselves, like some inferred variables about the style of play, defense, attack and some other categorical variable denoting a teams tactics under different match situations. Considering this data is of dynamic nature ie. it contains some temporally-varying tactical information linked with match-events, we kept this data out of the scope for our current work as we just want to utilize the game attributes as priors on our prediction of the match. However, in the future, it is possible to perhaps build team prior vectors as well and we will like to extend our work to explore this extended problem.

### 2.2 Pre-processing:

1. **Basic cleaning**: We converted the entire dataset into multiple CSV files. We used various pre-processing packages in R to remove some rows containing NULL and other nonsensical values (eg. string values present in columns with numerical attributes). We also pruned some other columns from both the matches table and the player attributes table. Some categorical fields were converted to numerical fields for convenience in future processing.

2. **Feature imputing**: Some of the player attributes were NULL or empty in the original data. For numerical attributes, we replaced NULL and empty values by the mean value of the feature across all the players. For categorical attributes, we did majority polling for replacing NULL and empty values.

3. **Feature**: To logically,"join" the information between the games player attributes and the match data, we created a primary key, foreign key in the player attributes table and the match data (having player information). This join was required in feature vector construction for the players and for various kinds of exploratory data analysis we wanted to do on the data.

4. We then added an additional field in the match for the outcome of the match. We infer this information We maintain two versions of this data, one where we keep the draw outcomes and where we exclude the draw outcomes.

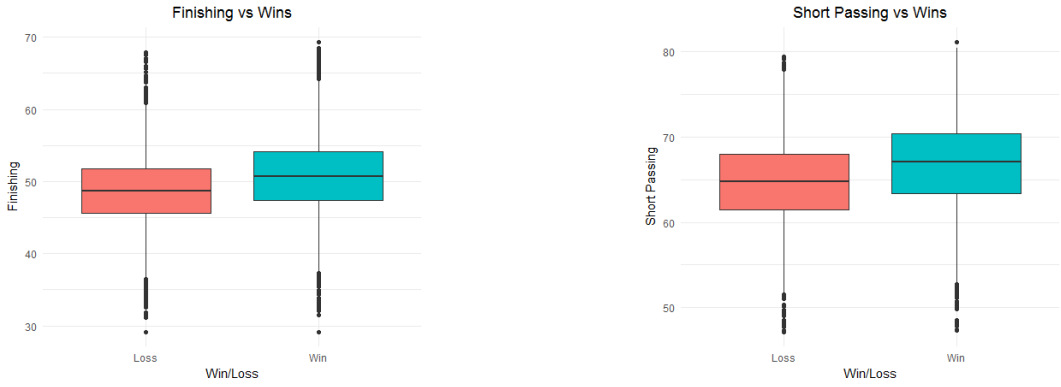
5. The amount of effort required in preprocessing, joining etc involved in the above process seemed to be equivalent to that of curating a new dataset.

## 3 Exploratory Data Analysis:

We started by looking at the relationship between winning and certain variables. The thinking behind this was that some variables might contribute more towards the outcome of a match than others. For example if the winning team had better short passing than the losing team. This intuition is reinforced by the boxplots.

Table 1: Table of Player Attributes

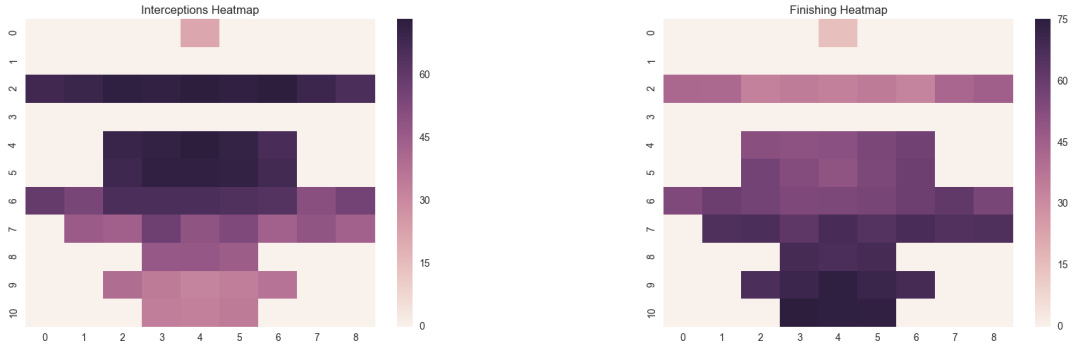
| | Skill | Description |
|---|---|---|
| 1 | Potential | 0 to 100. The ceiling on the players ability. |
| 2 | Attacking Work Rate | Low, high, medium. The players ability to be involved in attack. |
| 3 | Defensive Work Rate | Low, high, medium.Ability to be involved in defensive moves. |
| 4 | Crossing | 0 to 100. Ability to make aerial pass |
| 5 | Finishing | 0 to 100. Ability to convert a goal scoring opportunities |
| 6 | Heading Accuracy | 0 to 100. Ability to hit a header on goal. |
| 8 | Short Passing | 0 to 100. Ability to accurately pass the ball over short distances. |
| 9 | Dribbling | 0 to 100. Tricks |
| 10 | Curve | 0 -100. Ability to curve. |
| 11 | Free Kick Accuracy | 0 - 100. Measure of player's chances of scoring from free kick. |
| 12 | Long Passing | 0 to 100. Player's ability to accurately pass the ball over long distances. |
| 13 | Ball Control | 0 to 100. Measure of player's control over the ball |
| 14 | Acceleration | 0 to 100. As the name suggests. |
| 15 | Sprint Speed | 0 to 100. As the name suggests |
| 16 | Agility | 0 to 100, Measure of player's agility. |
| 17 | Reactions | 0 to 100. Higher the number faster the player's reactions. |
| 18 | Balance | 0 to 100. Ability to maintain posture |
| 19 | Strength | 0 to 100. Physical strength of the player |
| 20 | Long Shots | 0 to 100. Player's ability to take long shots on target (goal) |
| 21 | Aggression | 0 to 100. Player's aggression. |
| 22 | Interceptions | 0 to 100. Player's ability to intercept an opponent's pass. |
| 23 | Vision | 0 to 100. Player's ability to anticipate future playes. |
| 24 | Penalties | 0 to 100. Player's ability to score a penalty. |
| 25 | Standing Tackle | 0 to 100. Player's ability to tackle while on his feet. |
| 26 | Marking | 0 to 100. Player's ability to |
| 27 | Sliding tackle | 0 to 100. Player's ability to tackle while sliding. |
| 28 | Goalkeeper diving | 0 to 100. |
| 29 | Goalkeeper handling | 0 to 100. Precision in handling ball. |
| 30 | Goalkeeper kicking | 0 to 100 Accuracy of kick. |
| 31 | Goalkeeper positioning | 0 to 100 Tactical positioning of goalkeeper |
| 32 | Goalkeeper reflexes | 0 to 100. Goalkeeper's reflexes. |
| 33 | Shot_power | 0 to 100. Shot Power |
| 34 | Volleys | 0 to 100. Ability to score aerial goals. |
| 35 | Stamina | 0 to 100. Tendency not to tire. |
| 36 | Jumping | 0 to 100. Ability to jump high. |




(a) Finishing (b) Dribbling

Figure 1: Positively Correlated features

Next we carried out a heat map analysis of the data to figure out where among the teams a given feature had the largest density.




(a) Finishing (b) Finishing

Figure 2: Positively Correlated features

From the above heatmaps we can see that the defensive traits like interceptions are clustered towards the back while offensive traits are clustered towards the front. This is a motivating factor in choosing a model which encodes location over one which does not.

# 4 Technical Challenges:

1. **Data Preprocessing**: Parsing and extracting information from the SQLIte database proved to be very challenging and computationally expensive. We were initially trying to parse and process the data in SQLite using Python libraries, it was taking 4 hours for a single pass over the data. After having spent nearly 6 hours of trying to optimize the Python code, we switched to preprocessing in R and we were able to do everything blazingly fast in under an hour. After some more preprocessing and vectorizing all our operations, we got this time to under 10 minutes.

2. **Inferring player roles** : Different football teams have different team formations and the player position is not present in the dataset. For example, a team might have 3 defenders at the back, 4 midfielders and 3 strikers. Another team might have 5 defenders, 3 midfielders and 2 strikers. So, if we choose a naive way of creating feature vectors, we may end up aligning a defender with a midfielder or even worse, with a striker ! Quite clearly, creating feature vectors for this task is problematic. Therefore, we had to explicitly model the problem where we either directly or indirectly inferred the role of each player in the team, based on his x,y coordinates and his skills .

3. Handling NAs and effective feature imputing was a challenge for this task. For instance, a goalkeeper with a NA for finishing accuracy, might end up getting up a value of 50 with the naive feature imputation method which will clearly affect the performance of the model.

4. **Combining Clustered Attributes** : Another challenge was to combine average the defensive/midfield/striking attributes of the players in a team. We used sum, mean of clusters/zones.

5. **Dimensionality Reduction** : The number of feature dimensions was fairly large in our models. PCA, Recursive feature selection, Feature importance analysis etc. was required to remove insignificant noisy features like goalkeeping ability for an attacking player. In our case we used manual feature engineering to reduce the dimensions. For the defenders, midfielders and strikers we removed the goalkeeper attributes. While for goalkeepers we removed all the attributes which are pertinent to the other 3 classes. This is mainly because while other 3 roles have a soft separation between them goalkeeper is unique and hard separated from the other 3 classes. Running the model with the pruned features we achieved a better validation and test accuracy.

6. **Imbalanced dataset**: The dataset was not balanced. The dataset had 60% of wins for the home data. We tried upsampling the away win instances and we also tried out a weighted log loss function which utilized sample weights in order to overcome the majority class prediction problem.

7. **Player Evolution** : Some players change their position overtime (Eg, Cristiano Ronaldo transitioned from a winger to a forward). We chose to pick the closest(in terms of time) player attributes of the player available from the video game.

8. **Mixed-role players**: Some players have mixed roles (wingers, fullbacks, sweeper-keepers). It is hard to interpret any conclusions from model about mixed-role players.

# 5 Modeling:

## 5.1 Basic Model:

The basic model would be to predict the outcome of the based on the skill sets of the 22 players, 11 home players and 11 away players. The basic model performs very well in terms

of the usual metrics but is not very interpretable as different teams use different formations. There are 792 features, 36 for each of the players. We used a logistic classifier with the log loss objective function with LASSO regression (we wanted some of the feature vectors to got to 0, e.g. goal keeping skills on a defender doesn't make any sense). We created a holdout test set for error analysis. We used 5 fold cross validation to find the optimum value of the regularization parameter.The cross validation curves are given.

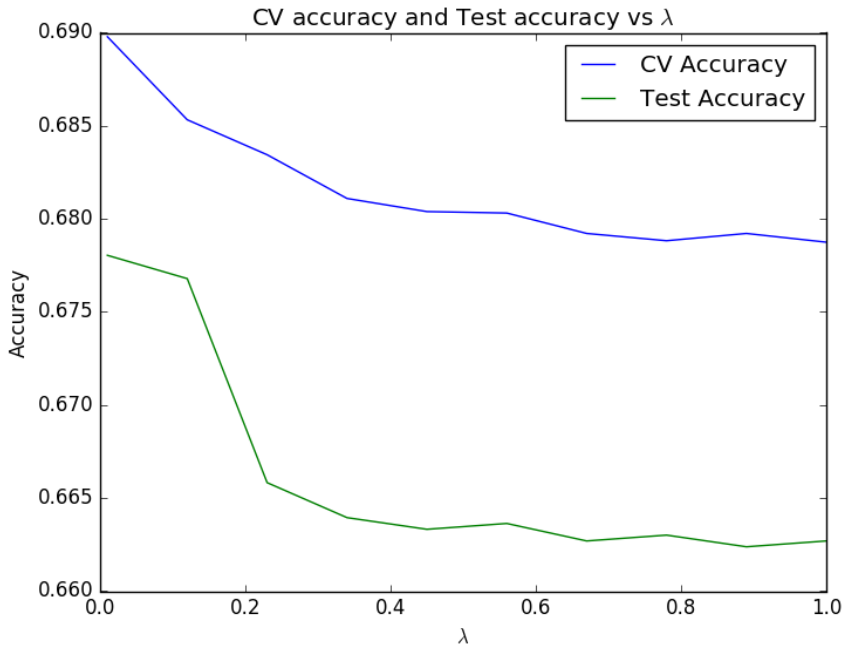



Figure 3: Cross Validation(Basic Model)

The following metrics are of use to the reader:

| | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| -1.0 | 0.56 | 0.70 | 0.62 | 1196 |
| 1.0 | 0.79 | 0.66 | 0.72 | 1997 |
| Avg/ Total | 0.70 | 0.68 | 0.68 | 3193 |

Table 2: Basic Model

The accuracy obtained on the holdout test set was 67.89%. The AUC for the model was 0.758 The ROC curve for the model is given below:

## 5.2 Role based low dimensional model:

In order to create a better interpretable model we clustered the players based on their perpendicular distance (basically their y coordinate on the pitch) from their own goal.We used k-means clustering for identifying players with similar y coordinates on the ground under the assumption that mostly players having similar y coordinates tend to have similar player

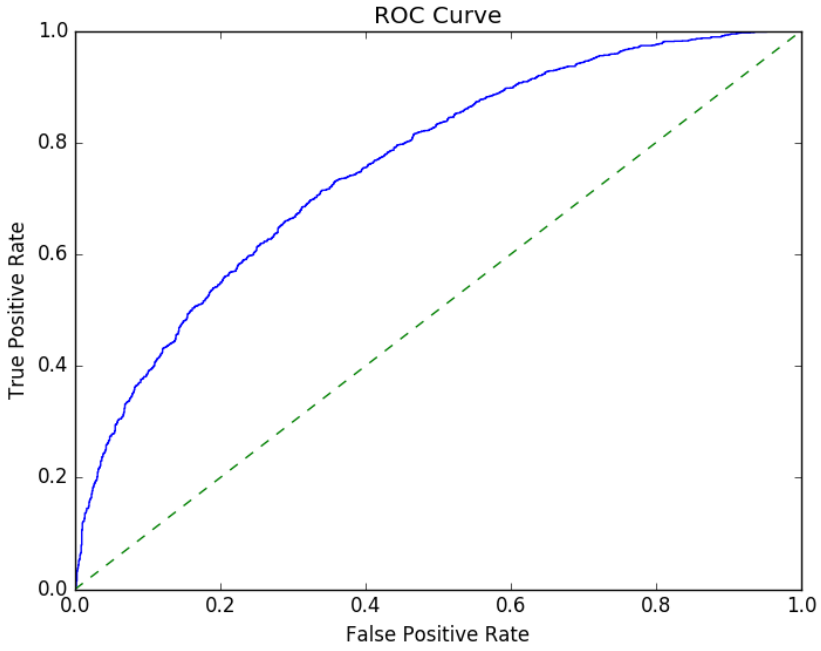



Figure 4: ROC Curve (Basic Model)

roles. For instance, players having high perpendicular distance from their own goal will tend to be attacking players or strikers, and the players having very low perpendicular distances will be defenders or defensive players. For this model we also pruned the non relevant features.

The added layer of interpretability will allow us to answer our question about how players with different roles in the team contribute to a victory and what specific skills in each of those skills contribute towards a victory. Using the k-means clustering algorithm we created 4 primary classes (Goalkeepers,defenders, midfielders and strikers) that each of the players belong to. Now instead of taking the skills of individual players we aggregated all the skills to represent the skills of a cluster. All players within a cluster contribute to the skill vector of their cluster. This aggregation is carried out in two ways:

1. Averaging the skills: The skills were averaged across the cluster to generate a unified skill set for the cluster. e.g. all the skills of defenders were averaged to give the skills of the defender cluster. Similarly for strikers and midfielders.

   Fitting a logistic classifier to these features we classify the outcome of a matchup. From this classifier we look at the weights for the variables that give the maximum boost to the probability of winning i.e. the variables with the largest weights. Using our domain knowledge of soccer we found that these features were intuitive and matched with our own understanding of how football matches are won.

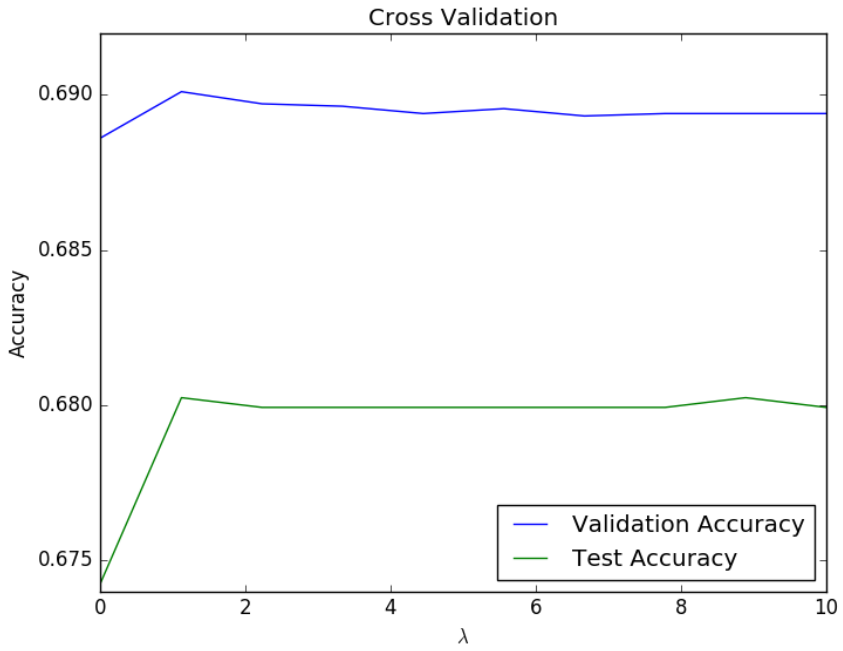



Figure 5: Cross Validation (Averaged Model)

| | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| -1.0 | 0.56 | 0.68 | 0.62 | 1196 |
| 1.0 | 0.78 | 0.68 | 0.73 | 1997 |
| Avg/ Total | 0.70 | 0.68 | 0.69 | 3193 |

Table 3: Average Model

The AUC was .7686. The accuracy corresponding to the optimum $\lambda$ was 68.05%

Table 4: Average Model

| Top 10 Features for Positive Outcome | | |
|---|---|---|
| 1 | Defender's Standing Tackle | 0.255 |
| 2 | Midfielder's Reactions | 0.221 |
| 3 | Goalkeeper's GK Kicking | 0.127 |
| 4 | Defender's Sliding Tackle | 0.120 |
| 5 | Midfielder's Ball Control | 0.117 |
| 6 | Midfielder's Penalties | 0.115 |
| 7 | Defender's Short Passing | 0.115 |
| 8 | Away Striker's Attacking Work Rate | 0.104 |
| 9 | Defender's Marking | 0.092 |
| 10 | Defender's Interceptions | 0.091 |

2. Summing up the skills: The skills were summed across the cluster to create an aggre-

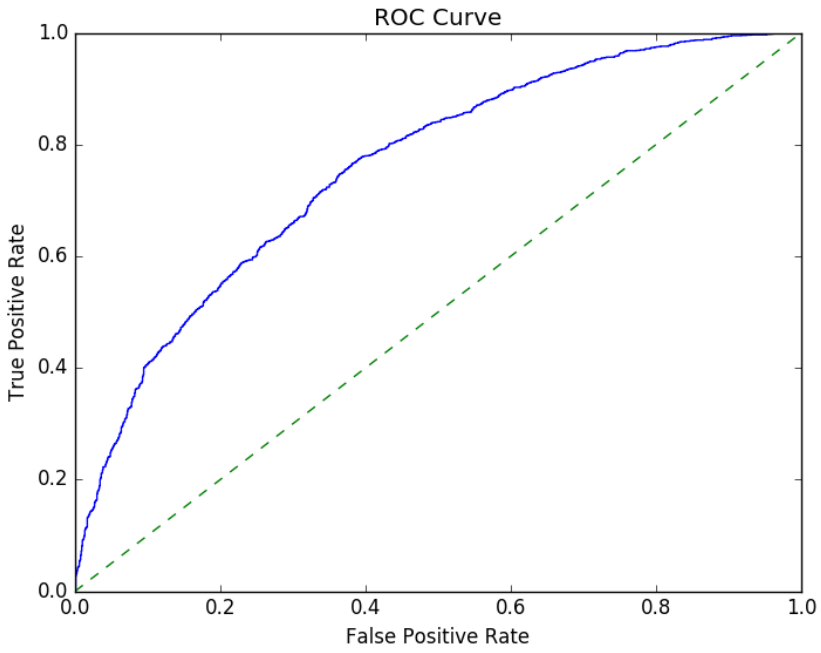



Figure 6: ROC Curve (Averaged Model)

gated skill for the cluster. This aggregation captures the number of players within the cluster and thus the relative strength. e.g. the cluster with 6 people will have better stats than a cluster with only 4 people. Both the methods produced nearly the same results both qualitatively and quantitatively.

| | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| -1.0 | 0.56 | 0.68 | 0.62 | 1196 |
| 1.0 | 0.78 | 0.68 | 0.73 | 1997 |
| Avg/ Total | 0.70 | 0.68 | 0.69 | 3193 |

Table 5: Summed Model

The accuracy of the model was 68.39%. The AUC was 0.761. The ROC curve is given:

The top 10 features for predicting positive outcome were:

## 5.3 Zonal Model

Inspired by the heat map in figure[] we broke down our field into rectangular zones of small sizes under the hypothesis that players having similar roles in the team tend to have a very strong distributional similarity. Our role-based model, although much better than our first mode, fails to capture mixed-role players like wingers who although play in defensive positions on the pitch, but are attack-minded and contribute more to attacking play rather than defensive play.

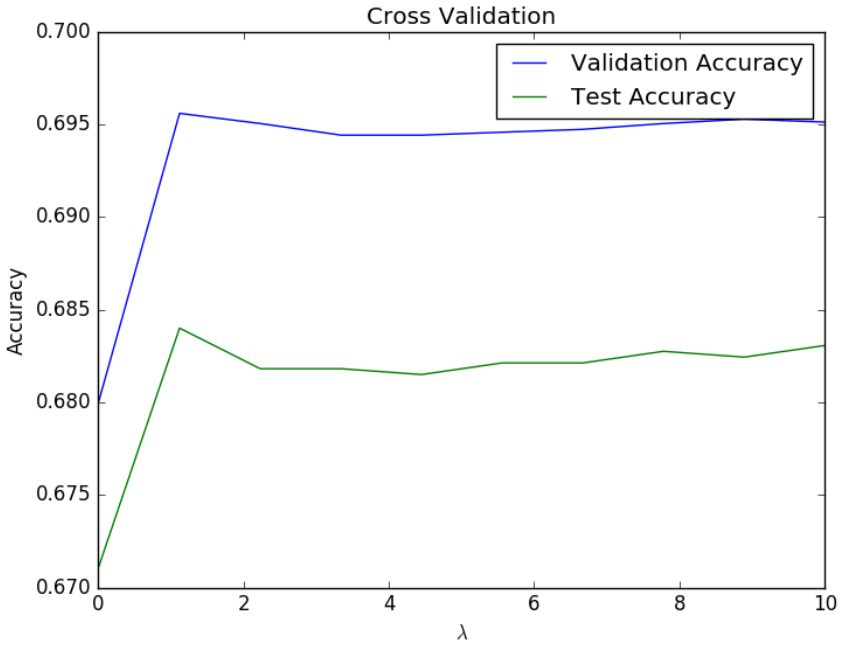



Figure 7: Cross Validation Curve (Summed Model)

Table 6: Summed Model

| Top 10 Features for Positive Outcome | | |
|---|---|---|
| 1 | Midfielder's Ball Control | 0.905 |
| 2 | Midfielder's Reactions | 0.758 |
| 3 | Striker's Vision | 0.73 |
| 4 | Striker's Reaction | 0.493 |
| 5 | Midfielder's Short Passing | 0.477 |
| 6 | Striker's Jumping | 0.37 |
| 7 | Away Midfielder's Dribbling | 0.353 |
| 8 | Striker's Positioning | 0.329 |
| 9 | Defender's Standing Tackle | 0.324 |
| 10 | Midfielder's Positioning | 0.304 |

Under this new model, players which have their x,y coordinate within the region spanned by the same zone, pool together their attributes towards the attributes of that particular zone. For example, if 2 defenders are present in the central zone present in front of the goalkeeper, we add the skills of both the defenders to calculate the aggregate attributes of that zone. In case the zone doesn't have any players in it; then the attributes or the feature vector for that zone are set 0.

Using the zonal model, the inference that one can glean is which zone needs to be strengthened by a specific team in order to increase their odds of winning. More importantly, this

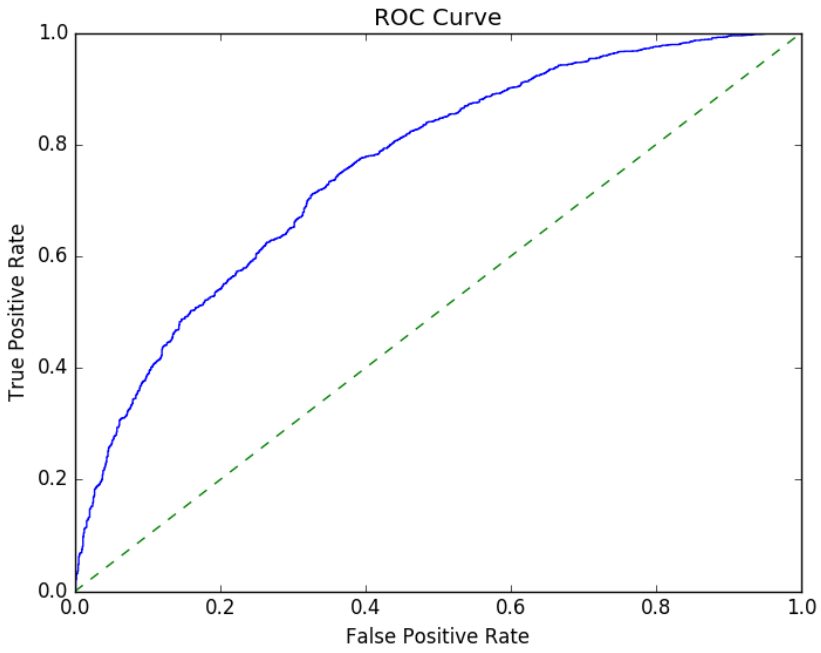



Figure 8: ROC Curve (Summed Model)

model allows us to analyze a team's structure at a much more granular level and now, we can easily evaluate importance of various skills at various zones on the pitch. For instance, our model now allows us to calculate the correlation of a team's victory with the shot accuracy attribute in the central attacking location. Similarly, we can evaluate how important is the short passing skill important in the middle zones of the pitch and how it is correlated with winning.

Another interesting inference we could draw from the model is that we could suggest teams better formations in order to increase their probability of winning, so basically we found that just by one or two rearrangements of players on the pitch, the team's chances of winning the game could change. We were able to test our hypothesis on the given data, but we didn't pursue this path further because we didn't have any gold-labeled ground truth to compare this.

We experimented with various small grid sizes. We tried out 3 x 3, 3x 4, 2x3, 4 x3 grids, and decided the best grid size using the performance on the validation set. Another factor to consider was that larger an individual zone, smaller will be the feature representation of the team and vice versa. We realized that 3 x 3 grid size gave us our best results and we therefore, focused on the feature vectors generated by this grid size in most of our experiments.

We tried out multiple classifiers in order to predict the match outcome using the zonal feature vectors created as above. We tried out Logistic Regression (with L1 norm penalty),

Support Vector Machine and XGBoost. We received the best results with Logistic Regression and XGBoost (both had similar accuracy and F-measure). We reason that L1 norm penalty here is important because there are a lot of attributes at each zone (36) and some of them are insignificant for that particular zone. Therefore, we try to enforce some sparsity in the feature weights.

The test accuracy for the above model using Logistic was 70.71%. The AUC was 0.7539.

| | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| -1.0 | 0.61 | 0.57 | 0.59 | 1169 |
| 1.0 | 0.76 | 0.79 | 0.77 | 2024 |
| Avg/ Total | 0.70 | 0.71 | 0.70 | 3193 |

Table 7: Zonal Model Logistic

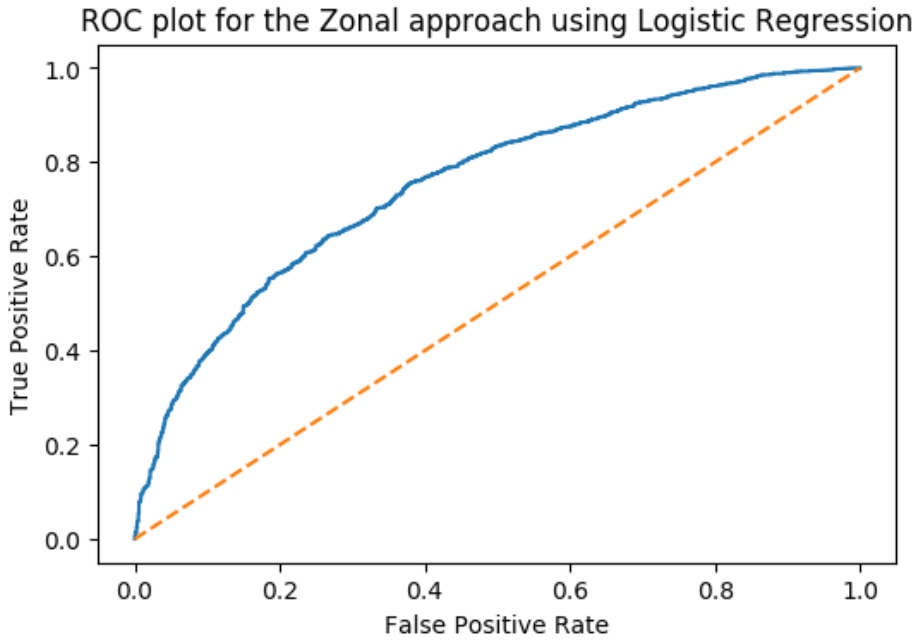



Figure 9: ROC Curve (Zonal Model Logistic)

The accuracy for the above model using XGBoost was 71.28%. The AUC was 0.7633

| | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| -1.0 | 0.64 | 0.50 | 0.56 | 1169 |
| 1.0 | 0.74 | 0.83 | 0.79 | 2024 |
| Avg/ Total | 0.70 | 0.71 | 0.70 | 3193 |

Table 8: Zonal Model XGBoost

The following figure shows the most important features for bucket 8 which is the position for the center forward/ Striker.

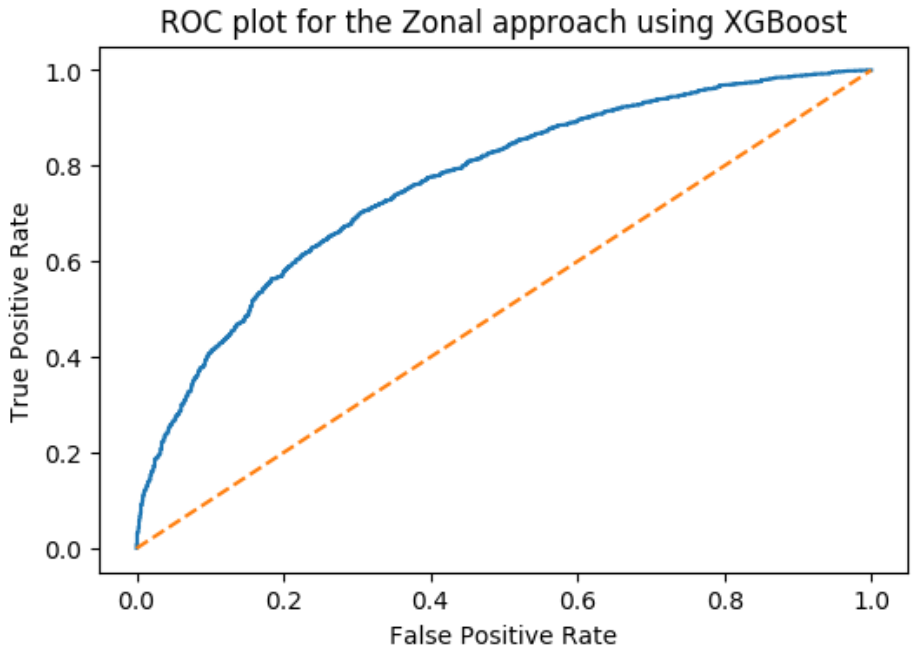



Figure 10: ROC Curve (XGBoost)

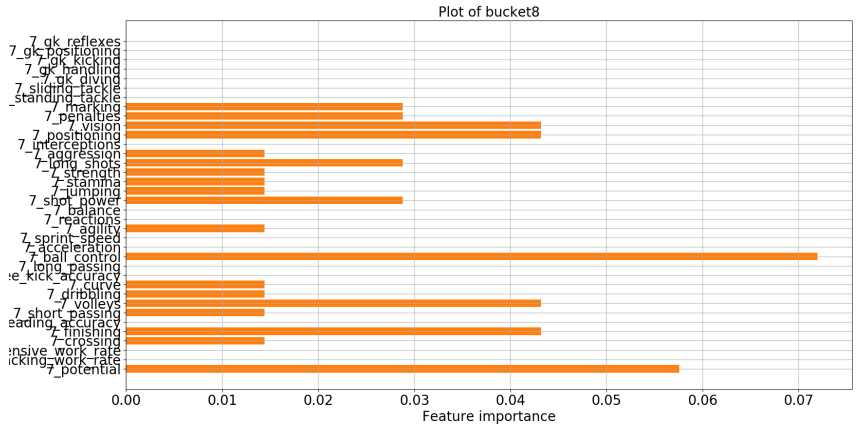



Figure 11: Most Important Features for Center Forward/Striker

## 5.4 Some other modeling considerations

In all our classification tasks above, to handle the skew in the data, we use the weighted log-loss as the objective to be minimized where weight of a particular class is inversely proportional to the ratio of that particular class.

To compare the performance of our L1 norm-based regularized penalty, we also experimented with recursive feature elimination method. We noticed that the accuracy and F1-measure we achieved with the two approaches were close to identical. Our L1 penalty for the role-based approach gave us 129 non-zero weight coefficients whereas recursive feature elimination re-

sulted in 99 features. Although, recursive feature elimination induced a greater sparsity, however, the associated computational cost was significantly than the Lasso penalty. Therefore, we decided to use the Lasso penalty for most of our experiments.

Feature scaling: We applied feature scaling before training all our models.

# 6 Failure Analysis and Future Work:

1. Soft-labeling of player roles : In our modeling approach where we try to cluster the players involved in a particular match on the basis of their y coordinate on the pitch ( basically the perpendicular distance from their own goal), we can utilize soft-labeling instead of hard labeling to get a more accurate representation of a players role in the team. For instance, a player having high passing attributes and high defensive attributes can contribute both as a defender and as a midfielder and our original approach isnt able to capture such cases.

2. Using more data : We can use the team attributes to build priors on the team. This can help us draw inferences on how different team attributes correlate with winning.

3. We wanted to try out more sophisticated models models for modeling the prediction but we felt linear models and simple classifiers are lot more directly interpretable than complex models. Therefore, in our current work, we sided with using basic (however very effective!) models.

4. Local Cluster Means for Feature Imputing: While replacing NA with mean values, we can take mean of a particular cluster instead of the entire mean. Our current method introduces some errors where a player may get assigned 50 ( mean for some of the features), when in reality, maybe that player doesn't have that skill at all.

5. Handling player evolution : Multiple records for player attributes were present in the game data for a single player (they had different reference dates). We however, handled this by taking the player attributes closest to the date of the match in our initial preprocessing. We think, it is possible to leverage the player evolution over time in the model prediction. For example, some players shift positions as they age. Some other players start losing their agility with age.

6. Better dimensionality reduction : We utilized recursive elimination, principal component analysis and some manual domain-knowledge derived feature pruning. We also tried to induce sparsity in the weight vectors by using a L1 norm penalty in our loss function. In the future, we will like to experiment with more advanced methods of dimensionality reduction.

# 7 Conclusion:

On the basis of the results we achieved, we confirm our hypothesis that FIFA player attributes are indeed useful in analyzing real life football matches. To confirm that our model indeed

performs very well, we tried to compare our performance against algorithms used by bookies which secure a $\approx 53\%$ accuracy on predicting outcome of any given football match ( where the outcome can be a win, a loss or a draw). We are able to achieve a 52.97% accuracy on the same task using the games player attributes as a prior. Our models secure a reasonable F1-score, area under the curve and a great accuracy for this difficult task, which shows that our models perform well. We were able to analyze the importance of individual features at a zonal level (which skill is more important for each zone) and therefore, the correlation of different skills with the match outcome.