International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-9 Issue-4, February 2020

Prediction of Soccer Matches using Machine Learning

Rohini H. Joshi

Abstract: ***The paper is concerned with predicting the result of a League and creating Strategies from gathered data using Machine Learning and Artificial Intelligence algorithms. Here we are taking data set from the real life game stats and from the massive multiplayer game series FIFA and PES. We will start by creating a web crawler to collect data-set and compare both the real world data and virtual data (Online game data) to predict the outcome of the match using supervised and unsupervised learning. Using K means clustering to segregate between different types of players such as offensive players, defensive players and goalkeepers using game data, then normalizing the virtual features to predict team strategies. We will use different models like Gaussian Naive Bayes, Hidden Markov model, Linear SVM etc. to reduce the error rates and increasing our accuracy to predict matches. This implementation can help all the teams to devise strategies for opposing team by knowing their strategies. This can also help to predict the winner of the league outcome. This prediction model can also be used in predicting Stock Market, Coaching improvements, journalism etc.***

Keywords: Artificial Intelligence, Gaussian Naive Bayes, Hidden Markov model, Linear SVM.

I. INTRODUCTION

Sports prediction is one of the latest research works which can predict the condition of winning or losing the game. For this all the data should be gathered and analyze properly, so artificial intelligence and machine learning algorithms are used and for creating a custom web scraper to scrape our data from different websites, also Support Vector Machine (SVM) is used. In SVM is used for classification as well as for regression challenges. In this, each data item is plotted with each feature and then classification is done.

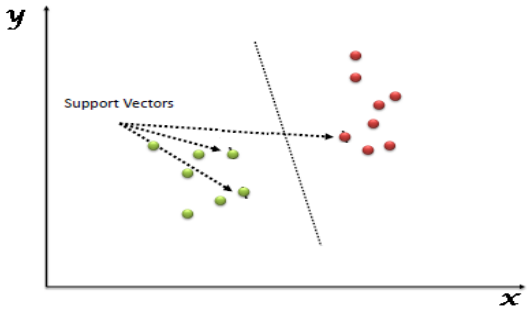

Fig. 1. Support Vectors

Revised Manuscript Received on February 28, 2020.
***** Correspondence Author
Rohini H. Joshi*, Assistant Professor, Department of Information Technology at Shri Ramdeobaba College of Engineering and Management, Nagpur, India.

The clustering algorithm is an unsupervised machine learning approach and that is used in statistical data analysis and in many fields. The following algorithms are used for clustering:

1. Connectivity models
2. Centroid models
3. Distribution models
4. Density Models.

In this research-Means Clustering is used to segregate between different types of players, so in this algorithm first we assign specify the desired number of clusters,then randomly assign data point to cluster, after that we compute cluster centroid and reassign to the closest centroid.After clustering, linear regression is used as predictive modeling technique.

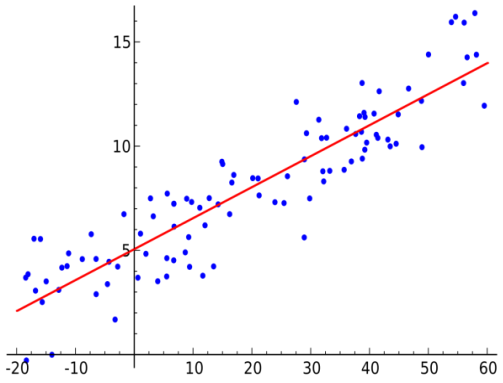

Fig.2. Simple Linear Regression

This figure concerns with two-dimensional points as independent variable and dependent variable, also it finds a linear function as accurately as possible and it predicts the dependent variable values as a function of the independent variables.

The virtual data is collected from video game to prediction and analysis of soccer match. Our objective is that our results must be comparable in way a way that it will suggest the strongest team who can win the match weak team who may lose the game.

II. LITERATURE REVIEW

The main work done by different authors on this field is described below:

Read the Beautiful Soup 4.0.0 documentation by Leonard Richardson [1] in this documentation the author describes Beautiful Soup a Python library which is used for extracting data from HTML and XML files.

Martin Spann, Bernd Skiera [2] Sports Forecasting: A Comparison of the Forecast Accuracy of Prediction Markets, Betting Odds and Tipsters, the author compares the accuracy with help in prediction market by analyzing and comparing the values for better accuracy.

Alan McCabe, Jarrod Trevathan [4], the authors describes the analysis of the literature in ML, which makes use of Artificial Neural Network (ANN) application to sport results prediction.

A. Joseph, N.E. Fenton, M. Neil [6], the author describes machine learning techniques for predicting the conditions of winning and losing of matches.

III. PROPOSED METHODOLOGY

In proposed methodology, first we collect the dataset from the web, the analyze the data and decide the attributes which are necessary for making prediction engine. We scaled and analyze the dataset by checking the scatter-plot using Matplotlib.Splitting of dataset into Train set and test set is done using Scikit-Learn then we trained the Train set using Machine learning models and the check the accuracy using Test set. The following figure shows the diagrammatic representation of research work:

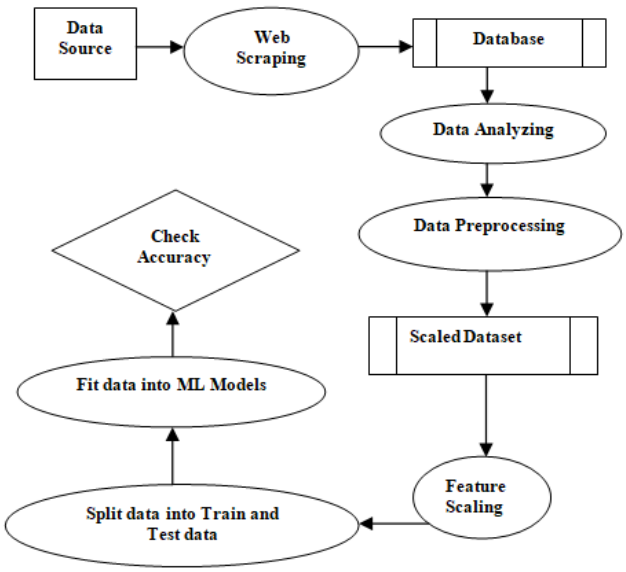

Fig.3. Diagram of Prediction Engine

The steps involved in this research work are as follows:

1. Apply some preprocessing steps to prepare the data.
2. Perform a descriptive analysis of the data to better understand the main characteristics.
3. Train different machine learning models using scikit-learn.
4. Iterate and evaluate the learned models by using unseen data.
5. Once we have chosen the candidate model, we will use it to perform predictive and to create a simple output that consumes this predictive model.

After scraping process, the dataset is prepared and analyzed and the prediction is done, so for analyzing the dataset the following properties of dataset is used:

1. The number of rows
2. Average values
3. Minimums and Maximums
4. Percentile's values and standard deviation

The overall column refers to a player's current rating and since it is the variable that we will use a measure of how good and bad player is, then we can get some initial interpretations of our data from above statistics like as follows:

1. We have 17611 observations (Players)
2. The average player's overall value is about 66
3. The worst player has an overall value of46
4. The best player has an overall value 94
5. Only one quarter of the players have an overall value greater than 7

After prediction analysis is done, we use Jupter notebook to implement this research work. The first dependency is used as pandas which is most popular library. The second dependency is XGBoost which is machine learning model that is used to form a prediction model based on an ensemble decision tree.

```
#data preprocessing
import pandas as pd
#produces a prediction model in the form of an ensemble of weak prediction models, typically decision tree
import xgboost as xgb
#the outcome (dependent variable) has only a limited number of possible values.
#Logistic Regression is used when response variable is categorical in nature.
from sklearn.linear_model import LogisticRegression
#A random forest is a meta estimator that fits a number of decision tree classifiers
#on various sub-samples of the dataset and use averaging to improve the predictive
#accuracy and control over-fitting.
from sklearn.ensemble import RandomForestClassifier
#a discriminative classifier formally defined by a separating hyperplane.
from sklearn.svm import SVC
#displayd data
from IPython.display import display
```

Fig.4 The dependencies

Here we see that about 46% is the win rate

```
Total number of matches: 5600
Number of features: 12
Number of matches won by home team: 2603
Win rate of home team: 46.48%
```

Fig.5. Output showing win rate of home team

After that we will try to visualize the distribution of the data, so we use panda which is a great tool called as scatter matrix, which basically shows how one variable affects the other and the correlation between the features.

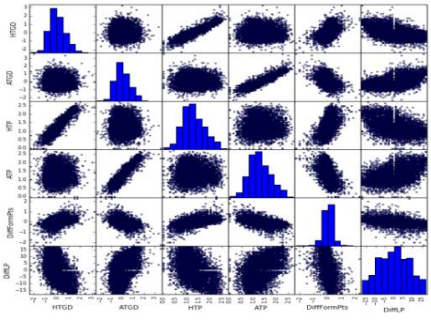

Fig.6. Scatter Matrix

The tools and technologies used in this research work are as follows:

1. Fixture Results

This source defines the results of the played matches and corresponds to the output variable in the prediction algorithm.

2. Real Data

Real data represents statistics about performance of team from the match history such as goals, yellow cards.

3. Virtual Data

In this source the data is collected and represents different features for each player set. In this research following processes are used:

A. Preprocessing Data for analytics- In this step we will preprocess the data by

- Saving it in .csv format
- Data preparation
- Tokenizing
- Cleaning the data
- Dimensionality Reduction

B. Using Machine Learning algorithms -

- Unsupervised Learning for strategic analysis
- Supervised Learning for predictor

C. Data Visualization We will visualize the data using Tensorboard and seaborn and using various techniques with it like PCA, T-SNE etc.

The technologies used in this research work includes Machine Learning, Web scraping and Data Analysis. The most important one is Tensorflow library. Detailed description of technologies used is given below:

1. BeautifulSoup

Beautiful Soup is a Python library which is used to extract HTML and XML data.

2. Python

Python is a high-level programming language which compiles and runs on every major platform, so we use python to implement this research work.

3. Anaconda

To implement this research work, we use open source version of Anaconda which is powered by Python..

4. Jupyter and Jupyter Notebook

It is a web application that allows creating and sharing of documents that contain live code, equations, visualizations and explanatory text.

5. Libraries

The libraries that are used for implementing this research work are Gensim, Matplotlib, NLTK, Numpy, Pandas. Scikit, Scipy Seaborn, Urllib.

6. Tensorboard and Embedding Projector

In this research work, a visualization tool Tensorboard is used to debug and optimize the complex programs.

IV. RESULTS

In this research, we have completed the dataset of 17981 players with 74 attributes After that the redundancies are removed and describe each with less properties i.e. 34 attributes as below:

	Acceleration	Aggression	Agility	Balance	Ball control	Composure	Crossing	Curve	Dribbling	Finishing	...	Reactions	Short passing	Shot power	Slidi tack
4549	83	63	87	83	68	70	58	70	78	66	...	69	68	72	43
13830	44	37	57	42	15	23	15	13	14	10	...	59	28	25	14
4164	77	51	81	83	75	76	70	65	72	58-1	...	66-1	76	68	64
3427	43	80	36	33	54	54	30	25	46	30	...	65	57	50	68
12506	89	54	92	62	64	70	49	37	62	56	...	67	61	61	28

5 rows × 34 columns

Fig.7. 34 attribute Dataset

The approach to implement is manually label 50 players, going one by one and writes down their positions, the train a logistics regression on that data and used it to predict the positions of all 17,981 players as shown below which will give 87.74% accuracy.

	Name	Preferred Positions	CB	ST	GK	CM	CDM	RM	LM	LB	RB	CAM	RW	LW	CF	LWB	RWB
0	Cristiano Ronaldo	ST LW	0	1	0	0	0	0	0	0	0	0	0	1	0	0	0
13	A. Sánchez	RM LW ST LM	0	1	0	0	0	1	1	0	0	0	0	1	0	0	0
14	L. Modrić	CDM CM	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0
15	G. Bale	RW	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
35	P. Pogba	CDM CM	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0

Fig.8. A predictor for preferred position

By using PCA, we can reduce the dimensionality of the data to just 2 dimensions, which will allow us to visualize the data.

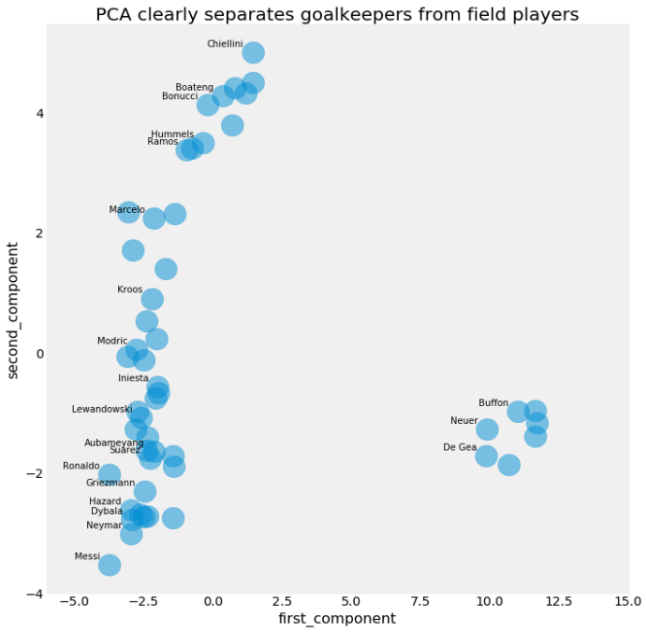

Fig.9. PCA Visualization

The first component is basically separating the goalkeepers from the field players, while the second component is differentiating within the field players themselves. The following figure shows PCA visualization excluding goalkeepers.

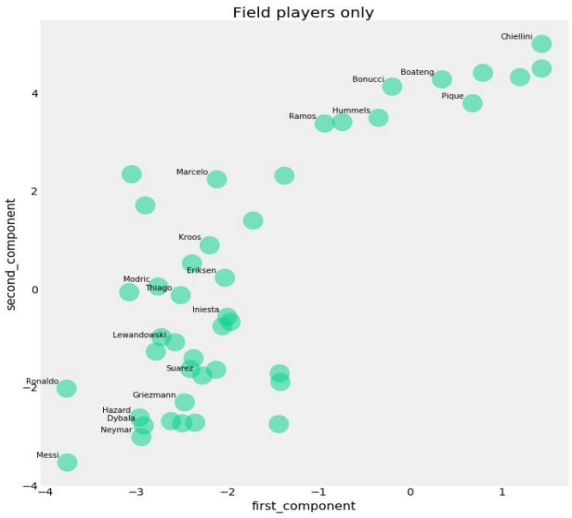

Fig.10. PCA Visualization excluding goalkeepers

The following figure shows the spaces between players, which helps to see the differences more clearly that is the defenders are in the right-upper quadrant-putting them closer to goalkeepers.

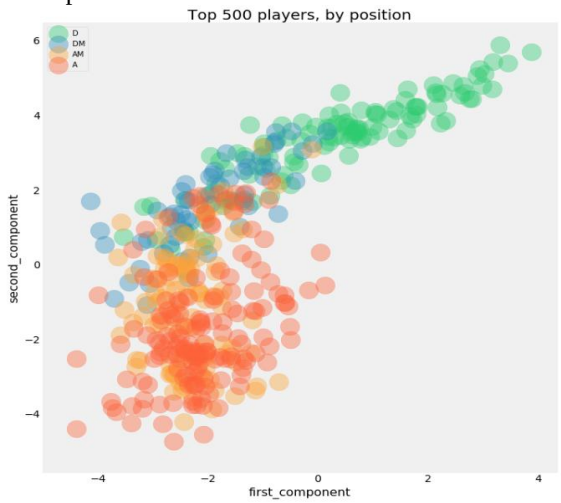

Fig.11. PCA Visualization with different colors

The above figure clearly shows that the defenders are still at the top, followed by the defensive midfielders, attacking midfielders and attacking players at the bottom. Now further proceed to the predictor, the result are in the form of accuracy of three classifiers that is logistic regression, SVM, XGBoost.

```
Training a LogisticRegression using a training set size of 5550. . .
Trained model in 0.2450 seconds
Made predictions in 0.0380 seconds.
0.621561035256 0.665405405405
F1 score and accuracy score for training set: 0.6216 , 0.6654.
Made predictions in 0.0000 seconds.
F1 score and accuracy score for test set: 0.6957 , 0.7200.

Training a SVC using a training set size of 5550. . .
Trained model in 2.5040 seconds
Made predictions in 1.2430 seconds.
0.620453572957 0.68036036036
F1 score and accuracy score for training set: 0.6205 , 0.6804.
Made predictions in 0.0250 seconds.
F1 score and accuracy score for test set: 0.6818 , 0.7200.

Training a XGBClassifier using a training set size of 5550. . .
Trained model in 0.4470 seconds
Made predictions in 0.0160 seconds.
0.652147113211 0.694954954955
F1 score and accuracy score for training set: 0.6521 , 0.6950.
Made predictions in 0.0020 seconds.
F1 score and accuracy score for test set: 0.7451 , 0.7400.
```

Fig: Output showing accuracy rate for different classifiers

We got 74% accuracy for XGBoost, this can be optimized using hyper-parameters as, shown:

```
parameters = { 'learning_rate' : [0.1],
               'n_estimators' : [40],
               'max_depth': [3],
               'min_child_weight': [3],
               'gamma':[0.4],
               'subsample' : [0.8],
               'colsample_bytree' : [0.8],
               'scale_pos_weight' : [1],
               'reg_alpha':[1e-5]
             }
```

Fig.12.The Hyper-parameters

After optimization we got more accurate results with 80% of accuracy as shown below:

```
XGBClassifier(base_score=0.5, colsample_bylevel=1, colsample_bytree=0.8,
       gamma=0.4, learning_rate=0.1, max_delta_step=0, max_depth=3,
       min_child_weight=3, missing=None, n_estimators=40, nthread=-1,
       objective='binary:logistic', reg_alpha=1e-05, reg_lambda=1,
       scale_pos_weight=1, seed=2, silent=True, subsample=0.8)
Made predictions in 0.0150 seconds.
F1 score and accuracy score for training set: 0.6365 , 0.6827.
Made predictions in 0.0000 seconds.
F1 score and accuracy score for test set: 0.7826 , 0.8000.
```

Fig.13. Output showing 80%accuracy

V. CONCLUSION

In this research work, we will decide which attributes to use as hyper-parameters in our Machine Learning models, hence implementing this algorithms we are able to find the strategic analysis and perdition of soccer match. We have achieved 80% accuracy using XGBoost. We took data of 15,670 players and their virtual ratings. We also did a PCA to cluster different type of players also we did prediction with 80% of accuracy. This research work can be used to make prediction in other different sectors such as Stock Market prediction, Weather prediction, Disease prediction etc.

REFERENCES

1. Beautiful Soup 4.0.0 documentation by Leonard Richardson
2. Martin Spann and Bernd Skiera, "Sports Forecasting: A Comparison of the Forecast Accuracy of Prediction Markets, Betting Odds and Tipsters" Journal of Forecasting, Wiley Online Library,2009
3. S. Luckner, J. Schroder, and C. Slamka, *" On the forecast accuracy of sports ̈prediction markets."* In Negotiation, Auctions, and Market Engineering,pages 227–234. Springer,2008
4. A. McCabe and J. Trevathan.*" Artificial intelligence in sports prediction",* pages 1194–1197. IEEE, 2008"The World's Most Watched League." Web. 11 Dec. 2014.
5. A. Joseph, A. E. Fenton, & M. Neil, Predicting football results using Bayesian nets and other machine learning techniques. Knowledge-Based Systems 19.7 2006: 544-553.
6. H. Rue and O. Salvesen, Prediction and retrospective analysis of soccer matches in a league. Journal of the Royal Statistical Society: Series D (The Statistician) 49.3 2000: 399-418
7. "Mark Lawrenson vs. Pinnacle Sports." Web. 11 Dec. 2014.
8. H. Kopka and P. W. Daly, A Guide to LATEX, 3rd ed. Harlow, England: Addison-Wesley, 1999.
9. Machine Learning [Online]. https://en.wikipedia.org/wiki/Machine_learningh Premier League Match Outcomes, 2013.

10. Ben Ulmer and Matthew Fernandez; Predicting Soccer Match results in the English Premier League, cs229, 2014.
11. Data mining [Online]. Available: https://en.wikipedia.org/wiki/Data_mining

AUTHOR PROFILE

Rohini H. Joshi, is working as Assistant Professor in the Department of Information Technology at Shri Ramdeobaba College of Engineering and Management, Nagpur, India. She has 4.5 years of teaching experience. She received her M.Tech degree from Nagpur University in 2015.She has published 5 research papers in reputed international journals and conferences. Her area of interest includes Image Processing, Web Designing and Machine Learning.