International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 10 | Oct 2020 www.irjet.net p-ISSN: 2395-0072

Analysis of Football Statistics for Prediction of Results using Data Mining Techniques

Femin Dharamshi[1], Husain Ali Unia[2], Jaini Gala[3], Rohan Shah[4]

[1]*Student, Dept. of Computer Engineering, Shah & Anchor Kutchhi Engineering College, Maharashtra, India*
[2]*Student, Dept. of Information Technology, K.J. Somaiya College of Engineering, Maharashtra, India*
[3]*Student, Dept. of Electronics and Telecommunication, K.J. Somaiya College of Engineering, Maharashtra, India*
[4]*Student, Dept. of Information Technology, K.J. Somaiya College of Engineering, Maharashtra, India*

Abstract - *Both of these algorithms help in maintaining relevance to the game as it uses real life data that can be used by players to strategize, make decisions which in turn preserves the interest of the player. Future research may use interactive prediction models where a user predicts every game's outcomes to be gifted which will further help improve his Fantasy squad. Using GP feature on this data.et, we can predict the outcome of matches successfully and thus give points to user based on their predictions.*

Key Words: Apriori, Naïve Bayesian, Data Mining, Football, Football Match Result

I. Introduction

Football has always been the most popular sport to be played and perceived in most countries of Europe and South America. Nevertheless, the sport's popularity has recently begun to boom within the Indian subcontinent. India is gradually becoming a global figure in the world of football, with more and more official football events taking place, as well as major international stars taking part in the new Indian Super League. Keeping this in mind, it was obvious that such a market of increasing enthusiasts needed to be tapped into with a football concept called "Fantasy Football". While the current Fantasy Premier League is rather successful, the turnover of players over the season has seen a decrease, i.e. players lose interest, is a common problem faced by it. A game-changing strategy was conceived to solve this problem, which led to the creation of this concept. Powered by an exhaustive dataset of all football statistics from 1992, the use of Data Mining techniques to predict future statistics seemed exciting. A point-based system on the predictions accuracy, which in effect allows better players to be purchased /auctioned, brings a greater competitive feeling to the current FPL scheme. This would avoid the churning of the season's teams, as such forecasts would encourage them to earn more points and better players.

II. Literature Review

In the past two decades fantasy sports have become increasingly popular. [4] As a result, numerous businesses are aggressively pursuing business opportunities within the field of fantasy sports, and in particular through leagues of fantasy football. The main aim of this project was to help Advanced Sports Logic (ASL) check the mathematical validity of "The Unit" calculations.

[5] Their results showed that socialization plays an enormous role in a fantasy game. It also demonstrated commitment to socialization through how much a league's players reacted to posts on the message board.

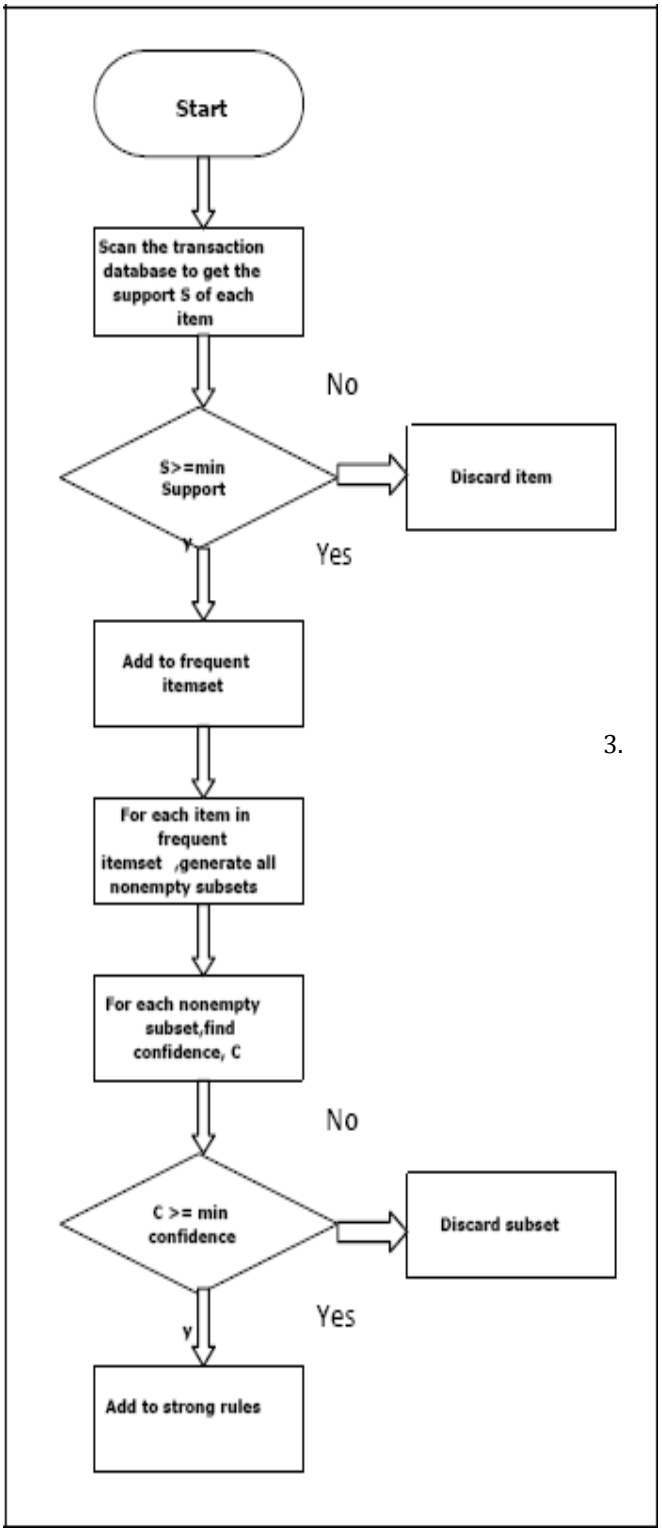
III. Proposed Methodology

The implementation of algorithms on the dataset predicted various types of outcomes for matches. The dataset was taken from the Kaggle repository which had records of 1185 groups. Each and every group had information of about 20 attributes. The attributes included id, age, gender, education, ethnicity, place, information of 14 different techniques and tricks spread over the broader categories of attributes such as numerical, binary, ordinal, interval scaled and ratio scaled. First, attributes required for Apriori algorithm and Naïve Bayesian algorithm were selected which were then cleaned. Apriori was applied using a data mining software called Rapid Miner for deriving association rules. Naïve Bayesian algorithm was applied performed using a simple Java program.

3.1. Apriori Algorithm

Using candidate set generation, the Algorithm is used in order to mine frequent itemsets and also learning association rules among them. Firstly, it recognizes regular individual items and then slowly expands them into larger sets, as long as those item sets appear in the database often enough. The collected performance can be used to illustrate general data base patterns. It has its market basket measurement applications. Following is the flowchart for the Apriori algorithm applied on the dataset.

Fig. 1: Apriori Algorithm Flowchart



Implementation of Apriori Algorithm:

It was used to mine frequent itemsets which for this dataset is players that frequently play together. Apriori algorithm uses these two main steps:

1) Join Step: This phase will create (K+1) item-set from K-itemsets by joining each item with itself.

2) Prune Step: This phase scans the data base count for each object. If the candidate element does not meet the minimum funding, it will be considered rare and will therefore be withdrawn. This step is performed to reduce the size of the candidate item-sets.

Apriori algorithm is a sequence of steps to be followed to find the most frequent itemset in the given database.

1. In the algorithm's first iteration, each item is taken as a candidate for a l -itemset. The algorithm must count individual element occurrences.

sup(item(i))

2. Let there be some minimal support come in, min-sup. The set of 1 – itemsets whose occurrence is satisfying the min sup are determined. Only candidates with more than or equal to min-sup will be taken forward for the next iteration and the others are pruned.

sup(item(i))>= min_sup

3. First, frequent items with min-sup are discovered on 2-itemset. For this, the 2-itemset is created in the join step by combing items with itself by forming a group of 2.

itemset(i,j) = {item(i), item(j)}

4. Candidates for the 2-itemset are pruned using threshold value of min-. The table will now have 2 things with just min-sup only.

sup(item(i))>= min_sup

5. The next iteration uses join and prune stage to shape 3-itemsets. Each iteration must obey property where the 3-itemsets subsets, i.e. each group's 2 –itemset subsets, collapse into min-sup. If all 2-itemset subsets are frequent then the superset is frequent pruned otherwise.

itemset(i,j,k) = {itemset(i,j), item(k)}

6. Next step will follow making 4-itemset by joining 3-itemset with itself and pruning if its subset does not meet the min_sup criteria. The algorithm is stopped when the most frequent itemset is achieved.

After implementation of above algorithm, the final frequent itemsets were found. Confidence of those itemsets was found to check if it crosses the minimum threshold. The ones that qualify the minimum threshold give the required association rules i.e. the players that frequently play together.

Confidence is given as,

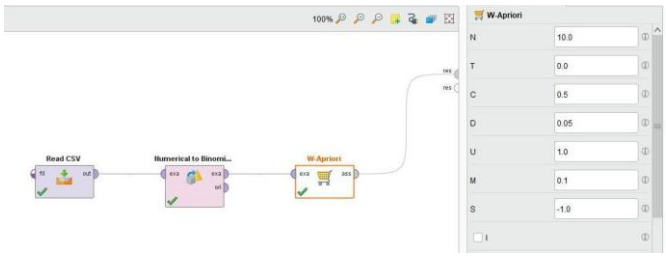
con(A -> B) =sup(AUB)/eup(B)

The ones that qualify the minimum threshold give the required association rules, i.e. the players that frequently play together.

Following portraits, the implementation done using

Rapid Miner:

Fig.2



3.2 Naïve Bayesian Algorithm

The Naïve Bayesian algorithm is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. It continues by calculating subsequent probabilities of various characteristics on the basis of which the dataset should be categorized. In case of a very la rge dataset, this algorithm proves effective and ac curate

Fig 3: Naïve Bayesian Algorithm

-Learning Phase: Given a training set **S**,

For each target value of c_i ($c_i = c_1, \cdots, c_L$)

$\hat{P}(C = c_i) \leftarrow$ estimate $P(C = c_i)$ with examples in **S**;

For every feature value x_{jk} of each feature X_j ($j = 1, \cdots, n; k = 1, \cdots, N_j$)

$\hat{P}(X_j = x_{jk} \mid C = c_i) \leftarrow$ estimate $P(X_j = x_{jk} \mid C = c_i)$ with examples in **S**;

Output: conditional probability tables; for X_j, $N_j \times L$ elements

-Test Phase: Given an unknown instance $\mathbf{X}' = (a_1', \cdots, a_n')$

Look up tables to assign the label c^* to $\mathbf{X}'$ if

$$[\hat{P}(a_1' \mid c^*) \cdots \hat{P}(a_n' \mid c^*)]\hat{P}(c^*) > [\hat{P}(a_1' \mid c) \cdots \hat{P}(a_n' \mid c)]\hat{P}(c), \quad c \neq c^*, c = c_1, \cdots, c_L$$

3.3 Implementation of Naïve Bayesian Algorithm:

This algorithm consists of two main phases:

1. Learning phase:

In this phase, probabilities of each feature for each target value are calculated.

2. Testing phase:

In this phase, using calculated probabilities and Bayes' Theorem to check probability of outcome for a particular set of features. Presence of a particular feature in a class is unrelated to the presence of any other feature. It continues by calculating subsequent probabilities of various characteristics on the basis of which the dataset should be categorized. In case of a very large dataset, this algorithm proves effective and accurate.

According to this example, Bayes theorem can be rewritten as:

$$P(y|X) = \frac{P(X|y)P(y)}{P(X)}$$

The 'y' variable is the set of outcomes of a match which is yes, no or draw. Variable X represents the parameters/features of a particular match, i.e. the attributes present in dataset.

X is given as,

$$X = (x_1, x_2, x_3, ..., x_n)$$

Here x1,x2....xn represent the attributes, i.e.they can be mapped to home team, goal keeper, etc. By substituting for X and expanding using the chain rule we get,

$$P(y|x_1, ..., x_n) = \frac{P(x_1|y)P(x_2|y)...P(x_n|y)P(y)}{P(x_1)P(x_2)...P(x_n)}$$

Now, if we obtain the values for each by looking at the dataset and substitute them into the equation. For all entries in the dataset, the denominator does not change, it remain static. Therefore, the denominator can be removed and proportionality can be introduced.

$$P(y|x_1, ..., x_n) \propto P(y) \prod_{i=1}^{n} P(x_i|y)$$

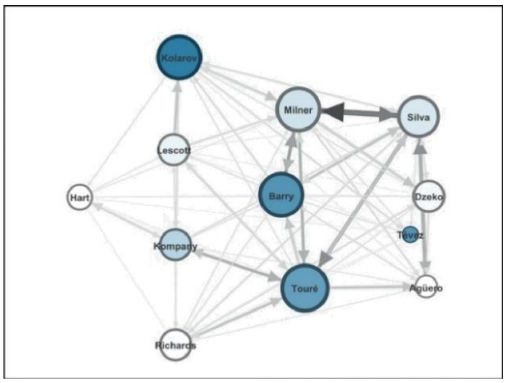
In our case, the class variable(y) has only two outcomes, yes or no. There could be cases where the classification could be multivariate. Therefore, we need to find the class y with maximum probability.

$$y = argmax_y P(y) \prod_{i=1}^{n} P(x_i|y)$$

Hence using Naïve Bayesian classifier, we can find the probability of an outcome of a particular match.

IV. Results

The Apriori algorithm was used to generate association rules on the sets of players that are often lined up together. The following results were used in to find out possible outcomes when the minimum support threshold was taken as 0.4 and minimum confidence was taken as 0.55.



The Naïve Bayesian algorithm is used to also classify the dataset based on the conditions of a match and to predict whether the outcome for a particular match. The algorithm was used to find out whether a Group 1, Forward from England will be able to score a goal or not, and the results were as follows:

```
Classifying whether a Group1, Forward from England will be able to score a goal or not on the basis of the following par
ameters using Naive Bayesian Algorithm:

Enter Group : 1

Enter Position : Forward
Enter Country : England
The Naive Bayesian probability for 'YES' is: 0.007099811
The Naive Bayesian probability for 'NO' is: 0.0033558714
YES! The Person can score
```

It says that the team can score a goal.

```
Classifying whether a Group1, Forward from England will be able to score a goal or not on the basis of the following par
ameters using Naive Bayesian Algorithm:

Enter Group : 3

Enter Position : defence
Enter Country : France
The Naive Bayesian probability for 'YES' is: 0.003944339
The Naive Bayesian probability for 'NO' is: 0.0053134626
NO! The Person cannot score
```

It says that the team cannot score a goal

V. Conclusion

Two of the many possible algorithms were implemented that help in predicting football match results. Apriori algorithm results gave us the sets of players that often are lined up to play together. This would help the players to select players for their fantasy football league teams using a more educated approach. On the other hand, the Naïve Bayesian algorithm assisted in effective classification. It helped in finding the probability of a particular outcome of match based on attributes such as players, if home team is playing or not, etc. This analysis is thus used in fantasy football by users to plan their games strategies, line ups, etc. Both of these algorithms help in maintaining relevance to the game as it uses real life data that can be used by players to strategize, make decisions which in turn preserves the interest of the player.

VI. REFERENCES

[1] Apriori Algorithm, ResearchGate

[2] NaïveBayesian, https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c

[3] https://www.researchgate.net/publication/267026589_Football_Match_Results_Prediction_Using_Artificial_Neural_Networks_The_Case_of_Iran_Pro_League.

[4] https://link.springer.com/article/10.1007/s10994-018-5703-7

[5] https://webcache.googleusercontent.com/search?q=cache:9GmbW9PAES0J:https://journals.psu.edu/ne/article/download/60898/60639/+&cd=1&hl=en&ct=clnk&gl=in&client=firefox-b-d