Analyzing and predicting NCAA volleyball match outcome using machine learning techniques

Dhvanil Sanghvi, Priya Deshpande, Suhas Shanbhogue and Vishwa Shah

BITS Pilani, India

Abstract

In this paper, we perform a thorough match prediction analysis of our newly mined NCAA (National Collegiate Athletic Association) volleyball data set. We also investigate the comparative power of two distinct yet comparable models, namely team aggregates and player aggregates, to predict the outcome of an NCAA volleyball match. The dependent variables for both models are mainly hitting rates of serves, recepts, attacks, and assists. The output variable is the winning team. Apart from the features specific to volleyball, we also incorporated a few general match statistics. Among the multitude of Machine Learning models available for classification, the study finalizes on three primary ones viz Logistic Regression, Decision trees, and Neural Networks. Results show that Decision trees and Neural Networks perform considerably well in both the team and player models on the ROC metric and accuracy, with Neural Networks giving marginally better results. Logistic regression on team aggregates performs only slightly better than randomized outcomes, whereas, for the player model, it performs way better. In terms of model structure, player aggregates give much better classification than team aggregates with a maximum ROC of 0.98. This shows that volleyball, despite being a team sport, is intrinsically more impacted by players who make the team than the team as a whole. Our model accuracy suggests that this model can be successfully used to predict the outcome of a NCAA volleyball match.

Keywords

NCAA, Data Mining, Volleyball, Machine Learning

1. Introduction

One of the most common task in Supervised Machine Learning is the Classification task [1]. This is mainly due to the use-case and entailment it has in sports predictions . Sports prediction is a part of an enormous market and forms the crux of a team's analyst. In some sports getting the strategy and team build right can make the difference of winning and losing the whole tournament . Stakeholders of a team like owners , coaches and analysts rely on computer simulations and models to predict the team performance with respect to strategies and tactics. Also large monetary rewards in betting further elucidate the necessity of good accurate models to predict sport matches. [2], states that betting markets are highly volatile and are subject to

CEUR Workshop Proceedings (CEUR-WS.org)

negative returns in the long run. In most literature historical stats, player performance stats and opposition information have been traditionally used as features [1]. [3], in their paper 'Using Bookmaker Odds to Predict the Final Result of Football Matches' state that bookmakers' odds correlate significantly with match predictions and can be used for predicting matches. [4], look into numeric predictions where the authors have dwelled into winning margins in college football. [5], use a unique ranking method to predict and model the English Premier League. [4],also make a keen observation that treating the prediction as a classification problem rather than a regression based classification gave higher accuracy.

Our paper focuses on match prediction as a classification problem of win and loss. While most existing literature focus on mainstream sports, we sought to draw our attention to another popular sport Volleyball and fill the void and gap in the current literature. Additionally we also compare how using holistic features of team stack up against amalgamated individual player features.

In recent times, Volleyball has gained immense popularity in the world of both professional sports and recreational leagues. The sport is played at the Olympics and also has many European and American leagues associated with it.Volleyball is played both on turf and the beach. It is important to note that these are entirely different sports and our focus in this paper is turf volleyball of the popular NCAA (National Collegiate Athletic Association) league.

1.1. Volleyball:Rules and Regulations

Before we dive deep into Machine Learning Research aspects, lets try to learn more about volleyball to get an intuition of the game. We briefly explain the structure and terminologies of Volleyball. The volleyball rules as stated in the NCAA Women's Volleyball Handbook are as follows:
A typical volleyball game consists of 6 members on each side of the court. The team however consists of 10-12 players with rotating substitutions. Every time a particular side serves the ball, there is a rotation among the 6 players so that no player serves twice continuously. The sport can be played indoors as well as outdoors. [6]

Most of the basic rules remain although the size of the court and position of boundaries changes slightly. The rules for volleyball are the same for women and men with the only exception that the official height of the net is shorter for women. The two ends of the net must be at the same height and it cant exceed the official height by more than 2cm. Some basic rules of volleyball include:

- A team cannot touch the ball more than 3 times before it crosses the net.
- A particular player cannot touch the ball more than once.
- The ball may not be lifted, held or carried.

Two forms of scoring are observed in volleyball. A serving team loses serve if it makes a mistake, a point is only given to the serve team if the non serving team makes an error, in the other case it leads to a service change. In the formally adopted scoring mechanism, each serve results in a point to either team, its sort of a rallying mechanism. Matches are played up to 25 points and 3 games. In order to win a game a team must have a 2 point score lead. Else the score keeps accumulating even beyond 25 until either team wins. The match ends when a team wins the

majority of the games i.e. 2. For this research we have used historical data for features of both teams and players to predict the winning team using the features mentioned in the following sections. Important terms and definitions used in this text can be found in the appendix section.

2. Review of existing literature

In [7] two models were developed for forecasting point spread for Women's Volleyball, one for predicting point spread using a regression model and a second model to predict the probability of winning using a logistic regression model. Difference between the averages of the in-game statistics was calculated between the two teams and placed in the model. The score margin model had an accuracy of 68% when the differences in the averages of the in-game statistics were used. [8] uses Logistic Regression for predicting Football results from Barclay's Premier League and sofifa.com. They highlight the most significant features used by previous researchers which include Home Offense, Home Defense, Away Offense, and Away Defense. The paper gives additional insight into the coefficients obtained Logistic regression, concluding that the most significant variables as Home Defense and Away Defense.

In [9] they discuss using a machine learning approach, ANN, to predict the outcomes of one week, specifically applied to the Iran Pro League (IPL) 2013-2014 football matches. The data obtained from the past matches in the seven last leagues are used to make better predictions for the future matches. Some unique features that have been used as input to ANNs involve Quality of Opponents in last matches, Condition of Teams in Recent Matches, Condition of Teams in the Overall League. [10] gives a good insight into our dataset and features of the box statistics of a volleyball match. The paper used the 1994 NCAA Women's volleyball tournament and calculated mean and standard deviation across a division along with the correlation coefficients. The authors use multiple regression to predict a match using the aforementioned features. The paper found that attack coefficient correlates the best to a team's success. Blocking stats were next important for division I and II while serve was next most important stat for division III. This simple regression method predicted 60% of the variance of the team's success across the divisions.

[11] is a review paper which gave us a bird's eye view of the existing research on Match predictions in the form of most popular sports which researchers choose for prediction and what is the frequency of ML algorithms used in existing literature. The paper shows that ANN are most frequently used in existing literature for predicting Team sport matches. The paper states 65% of the papers consider ANN models as part of their experiments and 23% of the papers solely use ANN in their work. But the authors say that using ANN models does not lead to high accuracy in prediction and it's unclear why historically ANN models have been so widely used. The authors say ANN models are black boxes and its very difficult for analysts and coaches to reverse engineer the outcomes of the ANN model predictions.

[12] uses multiple algorithms and even multiple sports in order to make sports analytics more accurate. Although volleyball is not a part of the sports discussed, the paper was very instrumental in giving us a direction to use individual player statistics to predict the final winning team. The paper also confirms our belief that individual player feature estimations are very much correlated with the team features.

[13] provides the framework and basis for the usage of Artificial Neural Networks to predict male volleyball professional league rankings. It also gives some insights on the features that can be used to predict the rankings such as wins, defeats, home/away etc. The paper concludes by suggesting that the best kind of ANN is one with a single hidden layer 4-neuron model which had "logsig" transfer function, "trainlm" training function, and "learngmd" adaptive learning function. [14] proposes a Bayesian hierarchical model to predict the rankings of the volleyball national teams. The model also allows the estimation of results of each match played in the league. The model consumes efficiencies in four categories - Serve, Attack, Defense and Block. The efficiencies are calculated as follows:

$$Efficiency = \frac{Perfect attempts - Total Errors}{Total attempts} \tag{1}$$

[15] adopts a Logistic Regression model based on efficiencies of the players at the different positions. The efficiency is calculated in a similar manner as Andrea Gabrio (2020). The different efficiency variables used were Libero player efficiency, Middle blocker efficiency, Setter efficiency, Middle blocker efficiency, Outside hitter efficiency and Universal hitter efficiency.

3. Dataset

3.1. Raw data

The data was obtained from National Collegiate Athletic Association's (NCAA) official website [16]. It includes data of all the Division 1 Women's Volleyball matches played from 2011 through 2015. Generally, volleyball statistics are split into the following 6 categories : Attacking, Serving, Setting, Passing, Defending and Blocking [6]. The analysis in this paper consumes statistics from the following four categories:

- Attacks
- Assists
- Serves
- Blocks

First we made a web crawler to crawl the NCAA website to extract all the match links of the statistics for individual matches. Then the data was scraped from these links of the NCAA website using BeautifulSoup, a package available in Python. The code for this scraper is available in this repository. We use a HTML parser to parse the stats page and all the different tables were extracted to a Pandas dataframe and stored into a Dictionary object. The data on the website was organized match-wise. So, first the links of all matches were extracted and then they were iterated over to obtain the data for each match. We made a dictionary to hold the matches statistics of all the matches until the current match. The dictionary was used as our database for generating prior and features for the current match prediction. The key of the dictionary was either a team name to hold team stats or a tuple of (team name, player ID , player name) to hold the player stats.

For the given match we would look into our Dictionary database and engineer the features for the current match. This ensures that there is no data leakage whatsoever into our Model

predictions. After feature engineering done, we feed the current match's data to the dictionary so that its stats can also be used as prior for next upcoming match. We discuss about the feature engineering in the next section.

3.2. Generating Priors for the Current Match

It is the generation of features from the primary dataset to make it ready to be consumed by the machine learning model. It is natural to assume that to predict $Match_i$ (Match number i of the tournament), the data that we have available to us is only till $Match_{i-1}$.

Phase 1: In phase 1, only the overall team statistics from the past matches are used to predict the outcome of the match. Therefore, a weighted average of the different statistics available was taken. A weight of one suggests a similar weight to recent matches than older matches. The decay factor was taken to be 0.9 so as to incorporate a decaying effect with age/duration of the match stat. This enables giving the performance in recent matches more weight than the performances in matches played quite some time back. This can be understood simply with the help of the following expression:
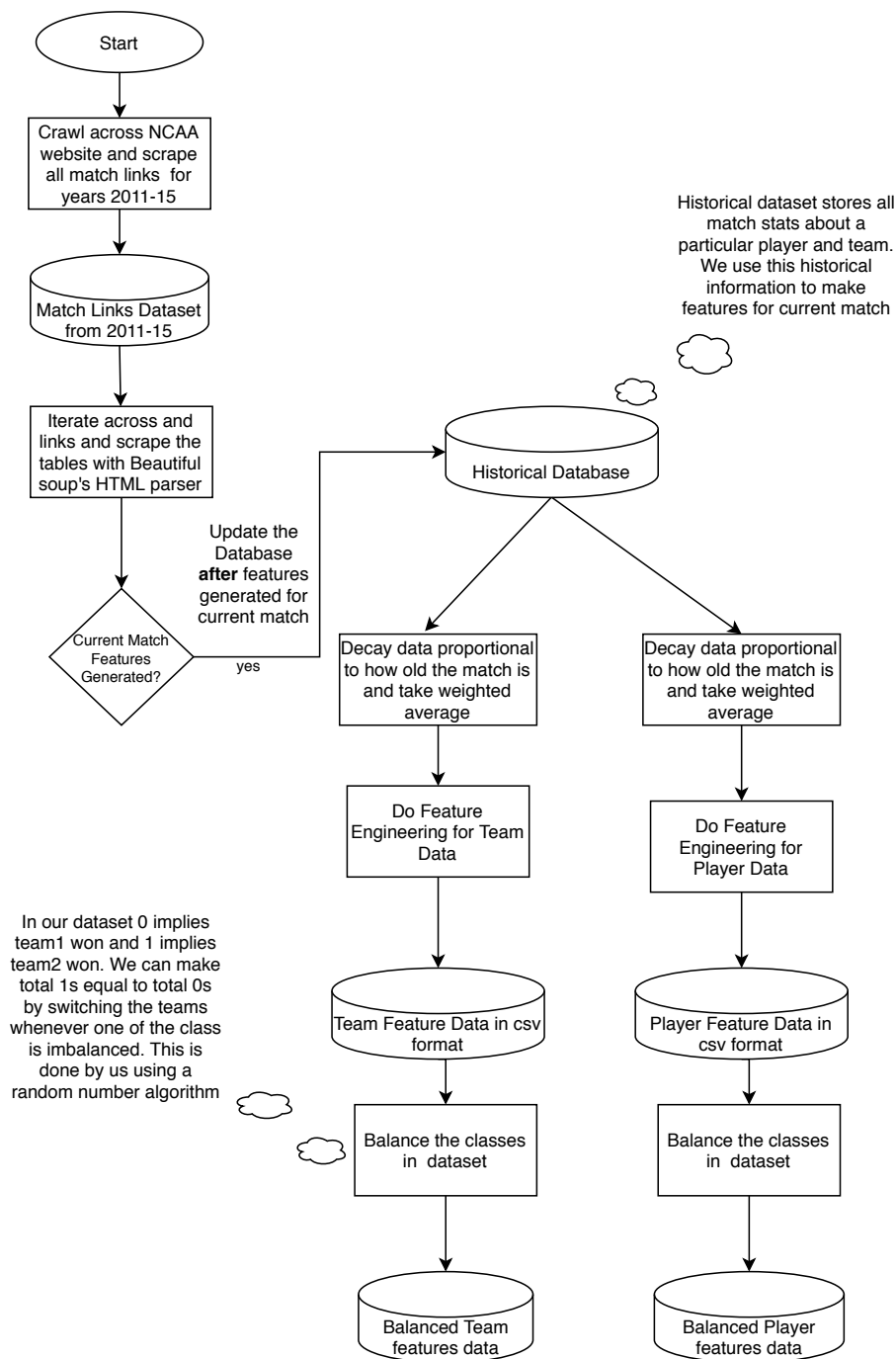
$$F_{m=i} = \frac{F_{m=i-1} + 0.9 * F_{m=i-2} + 0.9^2 * F_{m=i-3}...0.9^i * F_{m=0}}{1 + 0.9 + 0.9^2 + ... + 0.9^i} \tag{2}$$

Here, 'F' refers to a particular feature corresponding to that match and 'm' refers to the match number. Therefore, for a match number i played by a team, let's say Arizona, the features for that match would be a weighted average from the Arizona's previous match to Arizona's first match in that season.

Phase 2: In phase 2, player-wise statistics were used for the study. For each player of a particular team, the features of that player were engineered from their performance in the previous matches playing for the same team. The features were weighted in the same way as in phase 1.

4. Methodology and Feature Engineering

In order to predict the wins in a volleyball match, we decided upon two different paradigms: First, A structure that lays emphasis only on the team features and decides the match outcomes on the team statistics without giving much of a direct focus on the players. The other approach is to look at the players being the strength of the team and incorporate more directly the impact of the stronger players, rather than averages. The final features of both the approaches are calculated as historical averages. There are broadly six skills in volleyball that are considered crucial to a player/team's strength. Four of these are considered in this paper due to their relative importance: Attack, Assist, Serve, and Recept [17]. The concept of PCT (a term popularly used in volleyball to abbreviate percentage) is used to calculate both teams' and players' features [7]. Now, we are going to select the machine learning models on which we are going to train the data. Figure 1, adapted from [11] captures the frequency of usage for the different machine learning models applied to predict matches in the existing literature. We will train the same machine learning models for both the phases so that we can compare and contrast their performances

Figure 1: Flowchart of Data Scraping, Feature Engineering and Class Balancing

later. The first model that we choose is Logistic Regression. Our label is a binary variable and hence, the Logistic Regression model calculates the probability in the following way:

$$P(Y = 1|X, \alpha) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 * X_1 + ... + \beta_d * X_d)}} \tag{3}$$

Here, α is the parameter set for the model. The probability of Y belonging to label 0 can be just calculated by $P(Y = 0|X, \alpha) = 1 - P(Y = 1|X, \alpha)$.

As our second machine learning model, we choose Artificial Neural Networks also called feed-forward neural networks to be more precise. [11] suggests that it is the most widely used model for prediction of matches.

The third model is Decision Trees. A minimal cost complexity approach is used for pruning in the model. The splitting criterion used is - randomly initiate the threshold for all features and then iterative to find the best split. This helps reduce over fitting.

Overcoming class imbalance

Due to some cultural or circumstantial reasons, the NCAA data inherently had stats where the second team on the list won 90% of the matches. Upon further analysis we found no reasonable argument for the same. Also, most of the NCAA matches were played in neutral venues. To remove this bias from our data set we balance the number of 0's and 1's by randomly shuffling the ordering of the teams so that there is an equal probability of either team winning. The shuffling has lead to no loss of generality and the data set ends up being balanced.

4.1. Phase 1

In phase 1, we focus on team aggregates as a whole to predict wins. Individual match data was collected from the NCAA website. The train set consisted of six features which are calculated using the match statistics.

The data is cumulative in nature, in the sense that every following year contains the information cumulatively up to, but not including that year.

The following features are used:

- Attack PCT : Attack pct measures the average attacking power of the team. Attacks are the most important way for an offensive team to win points. Attack pct is directly proportional to the win rate [18]

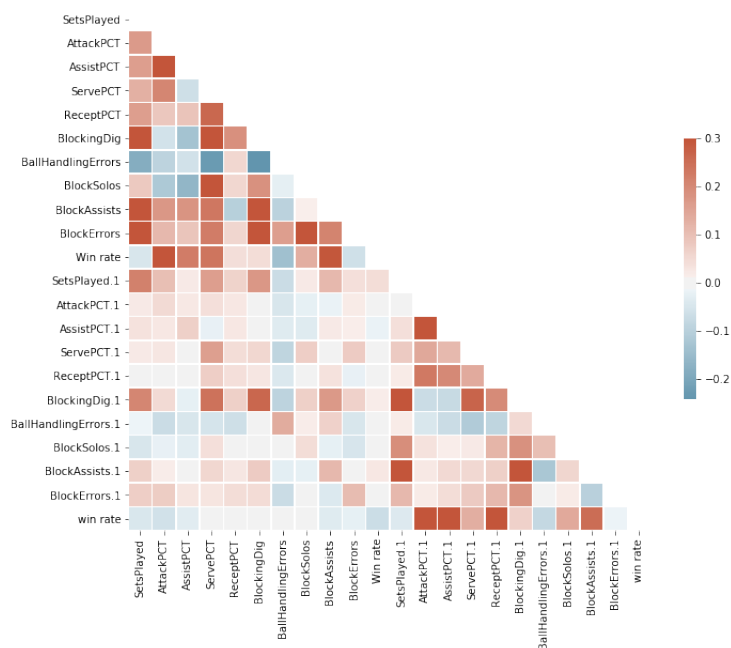$$AttackPCT = \frac{TotalKills - TotalErrors}{TotalAttempts} \tag{4}$$

- Serve PCT : Serve PCT is a measure of the number of serve aces by a team.

$$ServePCT = \frac{TotalServiceAces}{TotalSets} \tag{5}$$

- Assist PCT : A player is awarded an assist if he/she passes the ball to a teammate who then closes in on a kill or attack.

$$AssistPCT = \frac{TotalAssists - TotalErrors}{TotalAttempts} \tag{6}$$

Figure 2: Correlation Heatmap for NCAA Volleyball match features.

- Recept PCT :How well a team handles a potential serve ace is measured by the recept PCT.

$$ReceptPCT = \frac{TotalServeRecepts}{Totalsets} \tag{7}$$

- Set Win Ratio : It is the ratio of total set wins by all sets played.

$$SetWinRatio = \frac{TotalSetWins}{TotalSetsPlayed} \tag{8}$$

One important thing to note is that the features mentioned above show a high level of correlation with each other as shown in Figure 2 . This can be attributed to the fact that a proficient player might possess more than one skill to a reasonable extent [19]. A similar reasoning applies to team features as well. Therefore, we have considered these as individual features and maintain that they would individually add soundness and robustness to the model [20].

4.2. Phase 2

In phase 2, player-wise statistics were used for the study.To represent the efficiency and strength of the player we have used AttackPCT, AssistPCT, ServePCT. These features are measured similar to what we did for teams. It is considered that the squad playing a given match is not known a priori. For each match played, we use the 10 players of each team. For each player of

a particular team, the features of that player were engineered from their performance in the previous matches playing for the same team. The features were generated with a decay for the past performances (similar to Phase 1).

Similar to [21] we have defined Block Efficiency of a player as a ball that has been touched by blockers and then played by the defence.

$$BlockEfficiency_{ij} = \frac{BD_{ij} + BS_{ij} + BAij - BE_{ij} - BHEij}{BD_{ij} + BS_{ij} + BAij + BE_{ij} + BHEij} \quad (9)$$

where BD is Blocking digs, BS is Block solos, BA is Block Assists, BE is Block Errors, BHE is Ball handling errors. X_{ij} refers to the player j of a particular team i.

Ace is a serve which lands in the opponent's court without being touched, or is touched, but unable to be kept in play by one or more receiving team players, resulting in a point for the team serving. Since Ace is a special kind of serve, this ratio is a representative of the player's serving skills and hence used as a feature.

$$AceStrength_{ij} = \frac{ServeAces_{ij}}{TotalServes_{ij}} \quad (10)$$

4.2.1. Overcoming high dimensional data

The authors collected data of 299 matches spread across five years. For a particular year, there are fifty to sixty matches. But, using player-wise data leads to an explosion in the number of features because there are five features each for the twenty players who are going to play in that match (Ten per team) summing up to a total of 100 features. This is a severe problem [22] because the number of data points in a given year are much lesser than the number of features. Hence, it is imperative that measures are taken to reduce the dimension of the data [23].

It can be observed that all the five features for the players are efficiency ratios. Therefore, they are already normalized between zero to one. To reduce the number of features for prediction and to not lose important information, a single metric is allotted to every player. The metric (player-Score), acts as a proxy of the player strength and provides information on how valuable is the player to the team. It is calculated in the following way:

$$playerScore_{ij} = \frac{AttackPCT_{ij} + AssistPCT_{ij} + ServePCT_{ij} + AceStrength_{ij} + BlockEfficiency_{ij}}{5} \quad (11)$$

Here, similar to conventions followed above, j represents a particular player of a team i. Hence, it is just a mean of all those five features. These scores are generated for every player who is (predicted to) play in that match. After this is done, the data set is transformed to a new lower-dimensional data set. The new data set has the five featured mentioned above (of which the mean is taken) for three players of each team. The three players are said to be the representatives of the team for that match.

The first representative player's stats are an average of the best three players of that team on the basis of the calculated playerScore. Similarly, the second representative is an average of the three players who have medium strength - The fourth, fifth and the sixth best players for that team. And the third representative is an average of the poor performing teammates, having the least four playerScores. These three representatives are calculated in the same manner for the

second team. Now, for each match we have a total of 30 features containing the features of 3 representatives for each team.

This would allow us to capture important relationships between the match winner and the 3 representatives. Using the coefficients and the importance of features in case of logistic regression and decision tree respectively, we can empirically conclude relations between the match winner and the best players, the average players and the worst players.

4.2.2. Model construction

In this section we give details about the architecture of our models. These are the three models that were constructed for training on data set for our prediction. Each model was tuned for the best hyperparamters.

1. Logistic Regression model
2. Decision Tree Classifier
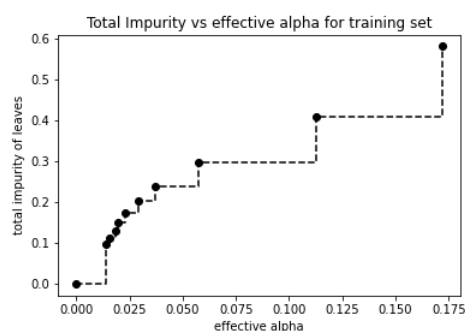3. Artificial Neural Network

Logistic Regression A tuned Logistic regression was used as a baseline for model training. A limit on the max iterations was set to 100000. The optimizer is set to limited memory BFGS(lbfgs) for our model.

Decision Tree Classifier Decision Tree Classifier from the sklearn library was used to implement the decision tree classifier. A critical factor is that with such limited data points and without any hyperparameter tuning, the tree over-fits the training set completely. We set a particular value for ccp_alpha, which is the hyperparameter for cost complexity pruning in Decision Tree provided by the sklearn library. To observe the variation in impurities in the leaves with the changing ccp_alpha, we plot the following graph.

We set ccp_alpha at 0.06 in our final model so that the model does not overfit and is robust. The model uses entropy as the criteria to judge the quality of split. The class weight is set to "balanced" so as to learn both the class labels equally.

Artificial Neural Networks

After firm experimentation on regularization in our ANN models, we zeroed in on the architectures which have been mentioned in the results. We use Adam optimizer and disable

Figure 3: Impurities in leaf nodes VS ccp_alpha

shuffling to prevent data leakage as our data is time series in nature.

5. Results

Since this is a classification task we have taken the ROC-AUC metric to analyze our model. Second metric we have used is F1-score, F1 Score is the harmonic mean of precision and recall. F1 score conveys the balance between the precision and the recall. We have used data spanning from 2011 to 2015 and since our data is dependent on time and the sequence in which matches were played, we have used the following train test split:

Table 1
Year-wise Train Test split

Split	Train	Test
Split1	2011	2012
Split2	2011, 2012	2013
Split3	2011, 2012, 2013	2014
Split4	2011, 2012, 2013, 2014	2015

Eg: here, data from matches played in 2011, 2012 would be used to predict for matches in 2013. We take the average of ROC-AUC score obtained in each of the above 4 cases. We have taken care that within the same season also the match data should not be shuffled and is ordered according to the dates when they took place.

5.1. Phase 1 - Team Data

In this section we compare the results of the models developed using Team Data and the features mentioned in section 4.1.

5.1.1. Logistic Regression

We develop a Logistic Regression Model for Binary Classification. We set the max iterations as 100 and set the class weight parameter as "balanced" to automatically adjust weights inversely proportional to class frequencies in the input data. The value of C has been chosen by tuning is across various values to get appropriate regularization(200 values on logarithmic scale). The output labels would be 1 or 0 depending on which team is predicted to win(0: team1 wins, 1: team2 wins)

5.1.2. Decision Tree

We use the Decision Tree Classifier and set the splitting criterion as gini and set the splitter as 'random', so that all features are sampled randomly according to the feature importance. As we have limited data, we want to ensure that the decision tree does not overfit. To ensure this we have tuned the max_depth and min_samples_leaf as 10 and 20 respectively. The ccp_alpha parameter here has been chosen as 0.01.

Table 2
Split wise ROC-AUC and F1 score using Logistic Regression using Team Data

Split	ROC AUC Score	F1 Score
Split1	0.57856	0.59701
Split2	0.59833	0.51064
Split3	0.72945	0.64407
Split4	0.66181	0.61290
Mean	0.64204	0.59116

Table 3
Split wise ROC-AUC and F1 score using Decision Tree using Team Data

Split	ROC AUC Score	F1 Score
Split1	0.5	0.66667
Split2	0.65917	0.51429
Split3	0.89022	0.89655
Split4	0.97607	0.91228
Mean	0.75636	0.74744

5.1.3. Artificial Neural Networks

We use Sequential Model provided by **Keras** Library for the Artificial Neural Network. The 5x2 features from both teams i.e. 10 units form the input layer. We use the ReLU (Rectified Linear Unit) activation function at both stages to learn a non-linear mapping for classification task. The final output is passed through a Sigmoid function- which will give an output in the range (0,1) denoting the probability of the team winning for our binary classification task. We train the Neural Network for 100 epochs. We set shuffle = FALSE as we want prevent data leakage as our data is time series in nature.

Input		**Dense**		**Dense**		**Output**
10 units	→	15 units,Actn:ReLU	→	25 units,Actn:ReLU	→	1 unit,Actn:Sigmoid

Table 4
Split wise ROC-AUC and F1 score using Artificial Neural Networks using Team Data

Split	ROC AUC Score	F1 Score
Split1	0.82258	0.81356
Split2	0.81667	0.81633
Split3	0.85483	0.84746
Split4	0.90323	0.9
Mean	0.84933	0.84434

In the above 3 models, there is a similar trend in the ROC/F1-score vs the Split being trained on. As the training data increases, the metrics of the model improve as the model has learned

on more data.

5.2. Phase 2 - Player-wise Data

In this section, we document the results obtained from the different models trained on player-wise statistics for a given Volleyball match.

5.2.1. Logistic Regression

Similar to Phase 1 Logistic regression on Team data, we tuned and trained a Logistic Regression Model for Players data to set a baseline . We set the class weight parameter as "balanced" to automatically adjust weights inversely proportional to class frequencies in the input data. We tuned across multiple C values (2000 of them divided on a logarithmic scale) to get the best regularization .

Table 5
Split wise ROC-AUC and F1 score using Logistic Regression using Player Data

Split	ROC AUC Score	F1 Score
Split1	0.98439	0.93103
Split2	0.95833	0.84615
Split3	0.98387	0.98412
Split4	1.0	1.0
Mean	0.98568	0.94032

5.2.2. Decision Tree

The Decision Tree is trained with ccp_alpha at 0.06. The class weights are set to balanced so that both the classes are learned equally. The criterion to decide the quality of a split is taken to be entropy.
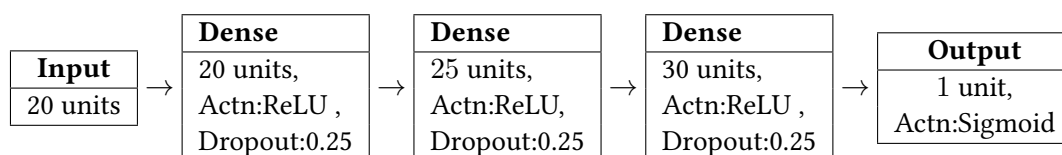
Table 6
Split wise ROC-AUC and F1 score using Decision Tree using Player Data

Split	ROC AUC Score	F1 Score
Split1	0.81842	0.76667
Split2	0.93583	0.87719
Split3	0.93184	0.95385
Split4	0.99844	0.98413
Mean	0.92113	0.89546

Table 6 clearly suggests that the decision tree model starts performing better as the number of training data instances are increased. From a F1 score of 0.76667 for 2012 data, it goes to an F1 score of 0.89546 for 2015. We observe an approximately linear growth in the ROC AUC score and F1 score across the years.

5.2.3. Artificial Neural Networks

We used player vectors of 30 features as input to the Neural Network and set shuffle = FALSE as we want prevent data leakage as our data is time series in nature. We trained for 10 epochs with Adam optimizer. Below is the architecture which gave the best results after tuning.

Input 20 units	→	**Dense** 20 units, Actn:ReLU , Dropout:0.25	→	**Dense** 25 units, Actn:ReLU, Dropout:0.25	→	**Dense** 30 units, Actn:ReLU , Dropout:0.25	→	**Output** 1 unit, Actn:Sigmoid

Table 7
Split wise ROC-AUC and F1 score using Artificial Neural Networks

Split	ROC AUC Score	F1 Score
Split1	0.96357	0.85714
Split2	0.97166	0.88000
Split3	1.0	1.0
Split4	1.0	1.0
Mean	0.98381	0.93428

In the above three models there is a clear trend of better metrics with more and more data with every passing year.

6. Discussion and Conclusion

The results that volleyball despite being a team sport , results are intrinsically more impacted by players who make the team than the team as a whole. In other words , models trained on Player data contain finer statistics than model trained on team data. This shows that in sports prediction bifurcated chunks of features which make the entity which is team gives better information to the model and is a better predictor.

Our classification results achieved better performance than strict guessing in all cases, with prediction ROC's ranging from 0.74 to almost 0.98 in some cases. In phase 1, the roc scores are 0.64, 0.75, 0.84 for LR, DT and NN's. Similarly for Phase 2 the corresponding scores are 0.98, 0.95, 0.98. Neural Networks perform better in both phases. The differences in accuracy are primarily due to the different approaches that we have chosen for this classification task. The aggregate player approach seems to have picked up the key causes that result in a team win as it incorporates significantly more information in its features. Although our model parameters were finalized through a series of experiments, we are aware that more specialized models could result in higher accuracy. Some general features such as average age of players, average age of the team, number of new players etc can be used as well. We were unable to use the same due to a lack of relevant data.

The metric for accuracy used was ROC AUC, as it is not biased towards the size of test or evaluation data. While accuracy is measured on predicted classes, roc auc is measured on

predicted scores which makes roc scores and f1 scores better for classification tasks. A high accuracy moreover could be due to over-fitting.

In phase 1, the mean ROC is highest for Artificial Neural Networks as they have the ability to learn and model non-linear and complex relationships between inputs and outputs. We observe Decision Tree performs much better in split4 when it is supplied the maximum test data.

Further extension of this model could use the extensive NCAA data available to make the current model more robust and versatile. Moreover, it could also incorporate a home and away team feature. In the current literature, we were unable to do so because the NCAA games are not necessarily conducted in the home or away grounds. We believe this is a crucial factor that must be used for sports predictions. We can also experiment with Recurrent Neural Network (RNN) architectures to learn from the temporal property of the data.

6.1. Usage of priors

An essential point of consideration for any probabilistic model is the inclusion of *prior probabilities* for all possible outcomes. One way to measure priors as mentioned in [24] is to use historical data to come up with some reasonable value. The paper uses the previous output to determine the new probabilities for the current year. This approach, however, poses several questions for time series data. How many years of data to use?, what if a new team/player joins the tournament?, with only five years of data and a few hundred matches, will the priors be biased to the train data? These questions require extensive research and are beyond the scope of this literature. Moreover, ncaa data pertains to university/college level matches. This implies that the players in a particular team may change considerably over time. Thus using priors on teams would be rendered useless. On players as well, priors would have to take into consideration their increased experience and skill level, the data to which we did not have access. After much discussion and deliberation, we come to a conclusion to use equally probable priors, i.e., we assume initially that both the teams are equally likely to win, and the only factors that affect the match outcome are the posteriors. An enhancement of the model could take into consideration these factors and estimate the relevant priors to further improve the classification accuracy.

6.2. Generating a network of players

The model that we have trained do not capture the synergistic relations between the different players. Although, Artificial Neural Networks might capture these relations implicitly, no inferences can be made from the trained neural network on these synergistic relations between the players. [25] uses an edge-centric multi-view network analysis to predict the performances of a given Basketball lineup in the NBA. The nodes of the networks are the players and the weights on the edges of the graphs represent the performance inhibitors/boosters due to the other players in the match. Using this technique could significantly understand the game of volleyball. For instance, if we find that the setter and outside hitter have a significant impact on each other, the team manager could choose to not replace these players in the current lineup. Similarly, calculating the centralities and eigen vectors of the network could help us get insights into the impact of the players in the team performance.

[26], in their paper propose the analysis of these player-interactions via the social network theory. They re-conceptualize the sports team as a social network and hence the relations between the nodes capture the interactions between these players. For eg., if the sport concerned was of Basketball, the network could be a ball-passing network.

These network-based approaches move away from the conventional machine learning methods, and include richer information into the model. Useful features can be generated after network analysis which can be helped generate robust models for the same conventional machine learning algorithms.

6.3. Using K means for Clustering and Merging Similar Players

While we used an algorithm of sorting and merging players in sections of 3, 3, 4 K means could have been employed to cluster similar players together and replacing the vectors of clustered Player with one single vector. [27], in their paper merge different players into relevant clusters to find a beta player which increased their model accuracy. The issue with this method is though its difficult to predict the number of players in each cluster. Many clusters may have more players and need to be weighted differently than clusters with lesser players to prevent bias. Also finding the optimal K value is another difficult task.

6.4. Decay Factor

We have chosen a decay factor of 0.9 with respect to the previous matches. But, technically the selection of the decay factor is in itself a search problem. Further research can be done to understand how the decay factor should vary within earlier matches of same season and matches from previous season. Here we have considered a geometrical decay factor. There are other questions like should this decay factors be varying for the past years? For example, should matches from 2 years back have the same decay factor as the matches from the past year. Most teams in NCAA play only for about 1-2 matches in a season. Hence, impacts of past 2-3 years can be seen in the data if we just use a simple decay factor without analyzing these things. The decay factors can also be adjusted on the basis of the volatility of players' skill. There are a lot of possible future directions this paper can be extended to and it will be interesting to see the impacts of them on the predictability.

6.5. Concluding remarks

In conclusion, we believe taking into consideration our limiting factors such as limited data availability, class imbalance, and inability to use cross-validation or shuffling as we were constrained by time-series nature of the data, the aggregative player model implements a sound classification task of predicting a volleyball win and can be used successfully for the given task.

References

[1] R. P. Bunker, F. Thabtah, A machine learning framework for sport result prediction, Applied Computing and Informatics 15 (2019) 27 – 33. URL: http://www.sciencedirect.com/

science/article/pii/S2210832717301485. doi:`https://doi.org/10.1016/j.aci.2017.09.005`.

[2] S. Wilkens, Sports prediction and betting models in the machine learning age: The case of tennis, SSRN Electronic Journal (2019). doi:`10.2139/ssrn.3506302`.

[3] K. Odachowski, J. Grekow, Using bookmaker odds to predict the final result of football matches, volume 7828, 2012, pp. 196–205. doi:`10.1007/978-3-642-37343-5_20`.

[4] D. C. D. Delen, N. Kasap, A comparative analysis of data mining methods in predicting ncaa bowl outcomes, 2012.

[5] R. Baboota, H. Kaur, Predictive analysis and modelling football results using machine learning approach for english premier league, 2018. doi:`10.1016/j.ijforecast.2018.01.003`.

[6] NCAA, Women's volleyball rules of the game, 2020. URL: http://www.ncaa.org/playing-rules/womens-volleyball-rules-game.

[7] D. Zhang, Forecasting point spread for women's volleyball, 2016.

[8] D. Prasetio, D. Harlili, Predicting football match results with logistic regression, 2016 International Conference On Advanced Informatics : Concepts, Theory And Application (ICAICTA) (2016). doi:`10.1109/ICAICTA.2016.7803111`.

[9] S. M. Arabzad, M. Araghi, S.-N. Soheil, N. Ghofrani, Football match results prediction using artificial neural networks; the case of iran pro league, International Journal of Applied Research on Industrial Engineering 1 (2014) 159–179.

[10] N. L. Estabrook, The relationship between ncaa volleyball statistics and team performance in women's intercollegiate volleyball, Kinesiology, Sport Studies, and Physical Education Master's Theses. (1996).

[11] R. Bunker, T. Susnjak, The application of machine learning techniques for predicting results in team sport: A review, 2019. `arXiv:1912.11762`.

[12] K. Apostolou, C. Tjortjis, Sports analytics algorithms for performance prediction, in: 2019 10th International Conference on Information, Intelligence, Systems and Applications (IISA), 2019, pp. 1–4. doi:`10.1109/IISA.2019.8900754`.

[13] A. E. Tümer, S. Koçer, Prediction of team league's rankings in volleyball by artificial neural network method, International Journal of Performance Analysis in Sport 17 (2017) 202–211. URL: https://doi.org/10.1080/24748668.2017.1331570. doi:`10.1080/24748668.2017.1331570`. `arXiv:https://doi.org/10.1080/24748668.2017.1331570`.

[14] A. Gabrio, Bayesian hierarchical models for the prediction of volleyball results, Journal of Applied Statistics 0 (2020) 1–21. URL: https://doi.org/10.1080/02664763.2020.1723506. doi:`10.1080/02664763.2020.1723506`. `arXiv:https://doi.org/10.1080/02664763.2020.1723506`.

[15] C. Akarçeşme, Is it possible to estimate match result in volleyball: A new prediction model, Central European Journal of Sport Sciences and Medicine 19 (2017). doi:`10.18276/cej.2017.3-01`.

[16] ncaa.org, Women's volleyball statistics, 2020. URL: http://www.ncaa.org/championships/statistics/womens-volleyball-statistics.

[17] A. Papageorgiou, 6 basic skills in volleyball, 2020. URL: https://www.strength-and-power-for-volleyball.com/basic-volleyball-skills.html.

[18] L. M. e. American Volleyball Coaches Association, Bonnie Johnson, 2020 women's volleyball

statisticians' manual, 2020. URL: http://fs.ncaa.org/Docs/stats/Stats_Manuals/VB/2020.pdf.

[19] G. B. e. a. Ferraz R, Pacing behaviour of players in team sports: Influence of match status manipulation and task duration knowledge, PLoS One 13 (2018). URL: https://doi.org/10.1371/journal.pone.0192399.

[20] S. Senthilnathan, Usefulness of correlation analysis, SSRN Electronic Journal (2019). doi:10.2139/ssrn.3416918.

[21] A. Gabrio, Bayesian hierarchical models for the prediction of volleyball results, Journal of Applied Statistics (2020). URL: https://doi.org/10.1080/02664763.2020.1723506. doi:https://doi.org/10.1080/02664763.2020.1723506.

[22] L. Yu, Feature selection for high-dimensional data: A fast correlation-based filter solution, Proceedings of the 20th international conference on machine learning (2003).

[23] L. Yu, Feature selection for high-dimensional data, 2015.

[24] T. I. M. Laura Hervert-Escobar, Neil Hernandez-Gress and, Bayesian based approach learning for outcome prediction of soccer matches, 2018.

[25] M. Ahmadalinezhad, M. Makrehchi, Basketball lineup performance prediction using edge-centric multi-view network analysis, Social Network Analysis and Mining 10 (2020) 72. URL: https://doi.org/10.1007/s13278-020-00677-0. doi:10.1007/s13278-020-00677-0.

[26] J. Ribeiro, P. Silva, R. Duarte, K. Davids, J. Garganta, Team sports performance analysed through the lens of social network theory: Implications for research and practice, Sports Medicine 47 (2017) 1689–1696. URL: https://doi.org/10.1007/s40279-017-0695-1. doi:10.1007/s40279-017-0695-1.

[27] R. Kumar, Z. Liu, W. Zamri, Sports competition stressors based on k-means algorithm, Malaysian Sports Journal (2019) 04–07. doi:10.26480/msj.01.2019.04.07.