International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Dataset Normalization in Cricket Score Prediction Using Weighted K-Means Clustering

[1]Mr. M. Chandru, [2]Dr. S. Prasath,

Submitted: 04/02/2024 **Revised**: 12/03/2024 **Accepted**: 18/03/2024

Abstract: Cricket, as a highly dynamic and unpredictable sport, presents a unique challenge for accurate score prediction. This study proposes a novel approach to cricket score prediction by integrating machine learning techniques with feature selection through weighted k-means clustering. The goal is to enhance the predictive accuracy by identifying and leveraging the most relevant features from a pool of diverse cricket match attributes. The methodology begins with the collection of comprehensive cricket match data, including player statistics, team performance metrics, and match conditions. These features form the basis for building a predictive model. To address the challenge of feature selection, weighted k-means clustering is employed. This technique assigns weights to features based on their importance, ensuring that the model focuses on the most influential variables. The dataset is preprocessed to handle missing values, normalize data, and address outliers. The preprocessed data is then subjected to weighted k-means clustering, where features are grouped into clusters, and weights are assigned based on the intrinsic significance of each feature within its cluster. This ensures that the model prioritizes features with higher weights during the prediction process. The machine learning model is constructed using an ensemble of algorithms, such as decision trees, random forests, and gradient boosting, to harness the collective power of diverse approaches. The selected features from the weighted k-means clustering are incorporated into the model, enhancing its ability to capture the intricate patterns inherent in cricket matches.

Keywords: *Cricket Score Prediction, Feature Selection, Machine Learning, Weighted K-Means Clustering*

I. INTRODUCTION

On a cricket pitch, two eleven-player teams compete in a game of bat-and-ball. In this game, the objective of each team is to score runs. The opposing side's strategy is to restrict the target score by dismissing the batters [1]. When regulation time runs out and the score is still tied, a 10-minute extra period is played. The victorious side is the one that racks up the most runs throughout the extra frame [2]. When it comes to scoring runs, the batsman is crucial. The main responsibility of a batter is to score runs. They achieve this by rushing between the wickets while hitting the ball with their bat [3]. For extra runs, the batter can also hit the ball over the fence for a four or six. The capacity to hit the ball with accuracy dictates a batsman's potential to score massive runs [4-6]. The quality of the opposition, the condition of the pitch, the weather, and countless other variables can all impact a batsman's performance [7-9]. The batting lineup is the foundation of every cricket squad. They are the engine that drives the team to victory with their run scoring. If his team has a good batsman, he can win the match by himself. For that reason, picking the right batsmen is paramount [10]. The hitter's job is to score as many runs as possible. We need to forecast the future performance of batsmen using their historical performance data; this is known as the batsman performance prediction problem, and it is a supervised learning problem. Consequently, while training the model, it is critical to think about all of these things. In the next ball, the runs scored by the batsman will be the target variable [11–13].

There are a lot of variables that might influence a batsman's score in cricket, making score prediction a challenging Endeavour. Regardless, the outcomes might be drastically altered due to biased umpire judgments [14]. On the other hand, a model can be trained using machine learning to anticipate a batsman's score using historical data. After then, a batsman's score in future matches can be predicted using this model. Strategic decisions, such choosing a batsman for a certain match, might be informed by this prediction [15].

A lot of studies have focused on how a match ends. Support vector machine, decision tree, random forest, and K-Nearest Neighbors' Algorithm (KNN) are among the machine learning classifiers that have been used to predict the matches' overall performance [16]. The cricket player's efficiency was estimated using a Data Envelopment Analysis (DEA) model that included batting-bowling metrics. In order to select the most effective team, thirty Bangladeshi crew members were

[1]*Ph.D Part-time Research Scholar & Assistant Professor, Department of Computer Science, VET Institute of Arts and Science (Co-education) College, Thindal, Erode, Tamil Nadu, India*
emchandru@gmail.com
[2]*Assistant Professor & Research Supervisor, Department of Computer Science, VET Institute of Arts and Science (Co-education) College, Thindal, Erode, Tamil Nadu, India*
softprasaths@gmail.com

advised to use a genetic algorithm-based framework [17–18]. This study use numerical analysis in conjunction with a genetic algorithm to identify exceptional performers. When assessing and ranking the players, statistical methods are used. I didn't use the output from prior matches in this game. Their score estimate for the first batting team was 71% accurate. While the play was underway, they used statistical models to forecast the One Day International (ODI) cricket production [19–21].

1.1 Motivation of the paper

Cricket score prediction has garnered significant interest due to the dynamic and unpredictable nature of the sport, presenting a challenging yet fascinating problem in the domain of sports analytics. Accurate prediction of cricket scores can provide valuable insights for teams, analysts, and enthusiasts, influencing strategic decisions and enhancing the overall spectator experience. Traditional approaches often struggle to capture the complexity of cricket matches, necessitating innovative methodologies that can extract meaningful patterns from a multitude of match attributes. This research is motivated by the need to overcome the limitations of existing prediction models by introducing a novel approach that combines the power of machine learning with sophisticated feature selection through weighted k-means clustering. The primary aim is to unlock the latent potential within the diverse cricket match attributes and improve predictive accuracy by prioritizing the most influential features.

II. BACKGROUND STUDY

Anik, A. I. et al [1] these days, machine learning is a commonplace occurrence in everyday life. With the use of machine learning, we can predict how well the bowlers and batsmen will do in the next cricket match. Through this technique, we may get insight into how a player performs in a competition, which will help team management choose the most suitable individual. In order to make better predictions, we want to gather data from domestic cricket leagues.

Emon, S. H., et al. [3] to tackle the problem of video summarization of cricket matches; we introduced a supervised version of an RL-based system in this article. We also developed a new dataset called CricSum to quickly test the network's summarization capabilities. A number of iterations of our model have been tested and shown to be effective on the CricSum dataset. Additionally, our model effectively captures semantic information about the cricket matches, as shown by the subjective analysis, which also reaches a very respectable value of Mean Opinion Score (MOS).

Fiaidhi, J et al. [5] our model has produced a reasonable and convincing method for the distribution of subsidized seeds based on the questions and holes in the present manual intervention, as well as the delays in insurance claims to farmers. A five-digit PIN secures all of the data stored inside, including facts on crops and farmers.

Jadhav, R., et al. [9] it is not an easy chore to choose the best starting lineup. Subjectivity aside, there are a number of things that must be thought about. Considerations such as the opponent's strength, the player's style of play, and his current form should all go into making a team pick. The automated method described in the study eliminates these problems. When you examine that the criteria account for the team's strengths and weaknesses, you can see that this is a successful strategy.

Rahman, R., et al. [13] an innovative approach was suggested for this study to detect various cricket delivery styles using offline real-time footage of bowling. The dataset was trained using a novel deep CNN model, which demonstrated exceptional accuracy when compared to other pre-trained transfer learning models. In addition, this study offered a whole new dataset with over 5,000 photographs that are organized into 13 distinct types of cricket bowling deliveries.

Smys, S., et al [17] "Cloud computing" is based on the principle of transferring data storage and processing to an off-site facility, or the "cloud." These resources are available 24/7 to every user, no matter where they are. Meta heuristic approaches require precise tuning of optimization parameters to find better solutions with less computational overhead. Among the many swarm intelligence algorithms, the artificial fish swarm algorithm (AFSA) stands out as a top optimization method.

Vetukuri, V. S., et al [20] In order to provide objective recommendations on cricket play errors for an upcoming event, a new, innovative, and very efficient approach has been suggested. The recommendation engine does this by using the principles of recurrent neural networks and genetic algorithms. Runs of the system's simulations using historical data provide respectable results in terms of accuracy. There are fifteen players from six different national teams and forty-five distinct tournaments used in the simulations.

2.1 Problem definition

The problem at hand revolves around the intricate challenge of accurately predicting cricket scores, a task complicated by the sport's dynamic and unpredictable nature. Current methodologies often fall short in capturing the diverse array of match attributes, including player statistics, team performance metrics, and match conditions. This study addresses the need for an innovative approach by proposing a cricket score prediction model that integrates machine learning techniques with feature selection through weighted k-means clustering. The primary challenges include identifying the most influential variables amid the diversity of match attributes, adapting to the dynamic nature of cricket, effectively selecting relevant features, addressing data preprocessing issues, and ensuring the model's generalization across various match scenarios and formats. The ultimate goal is to contribute to the development of a robust and adaptable model that significantly improves the accuracy of cricket score predictions, catering to the complexities inherent in the sport.

III. MATERIALS AND METHODS

For the purpose of this study, we use a thorough materials and methods framework to forecast cricket scores using ML. Dataset normalization, achieved by means of Z score normalization, is an integral part of the approach that helps to optimize and standardize data distribution. In order to find useful features and lower dimensionality, the Bidirectional Search algorithm is used to perform Feature Selection. Enhanced Gradient Boosting is used to implement classification, which improves predicted accuracy by resolving data complications.

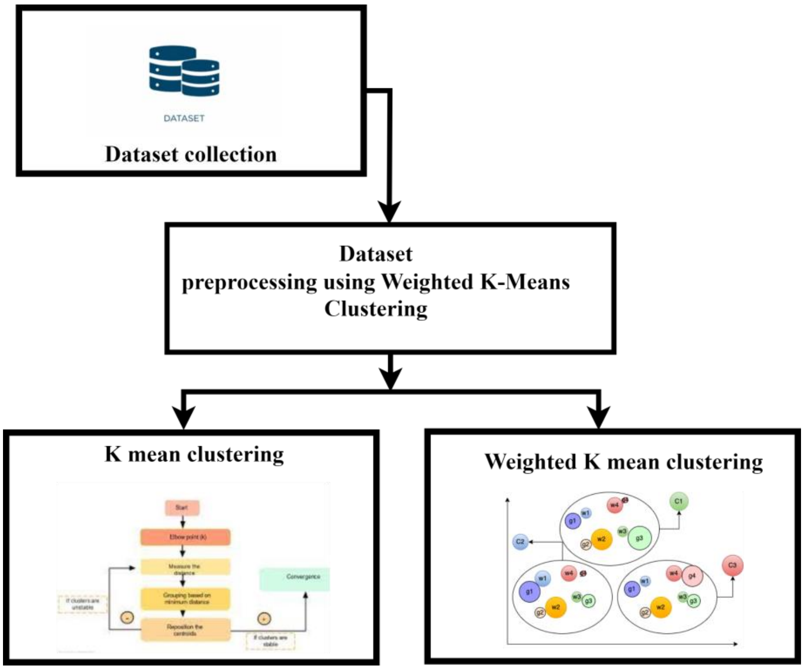

Figure 1: Overall architecture

3.1 Dataset collection

The dataset collected from Kaggle website https://www.kaggle.com/datasets/muhammedabdulazeem/icc-cricket-score-and-results

3.2 Dataset preprocessing using Weighted K-Means Clustering

In the preprocessing phase using Weighted K-Means Clustering for cricket score prediction, the dataset is initially cleaned, relevant features are selected, and normalization is applied to ensure uniformity. Unique to this process, weights are assigned to individual samples based on their significance in the context of cricket matches. The weighted K-Means Clustering algorithm is then employed, considering both spatial distribution and assigned weights during clustering. Subsequently, cluster labels are assigned to each data point, and these clusters capture nuanced patterns in player performance or match conditions. The resulting clusters or centroids are integrated into the cricket score prediction model, enhancing its ability to discern complex relationships and improve accuracy. The entire process is iterative, with evaluation and fine-tuning steps to optimize the predictive performance of the model.

3.2.1 K-Means Clustering

K-Means Clustering is a popular unsupervised machine learning algorithm used for partitioning a dataset into distinct groups, or clusters, based on similarities among data points referred by Ikotun, A.M. et al. (2023). Data points are repeatedly assigned to the cluster with the closest centroid (mean), and the algorithm updates the centroids appropriately, with the goal of minimizing the variation within clusters. The selection of starting centroids is critical since K-Means is sensitive to the user-specified number of clusters (k) and may converge to a local minimum. Image segmentation, customer segmentation, and anomaly detection are just a few of the many areas where K-Means shines, despite its apparent simplicity.

K-means is one of the simplest and most efficient unsupervised classification techniques. The K-means clustering algorithm is widely used since it allows the user to choose the number of groups and the distances between them. As with most distance-based clustering algorithms, similarity is determined by how far apart clustered items are from one another. The phases involved in k-means clustering are graphical approach are as follows:

Since there are only two potential values for the dependent variable (positive and negative), we'll set k = 2.

The next step is to use the extracted equation (9) to find the cluster center closest to each set of input data.

$$S_i^{(t)} = \left\{x_p: \left|\left|x_p - m_i^{(t)}\right|\right|^2 \leq \left|\left|x_p - m_i^{(t)}\right|\right|^2 \forall_j, 1 \leq j \leq k\right\} \text{------ (1)}$$

Refresh the cluster centers by recalculating the mean of the data for each cluster using equation (1).

$$m_i^{(t+1)} = \frac{1}{\left|S_i^{(t)}\right|} \sum_{x_j \in S_i^{(t)}} x_j \text{ ------- (2)}$$

We terminate our k-means clustering by iterating until the clusters' mean values converge.

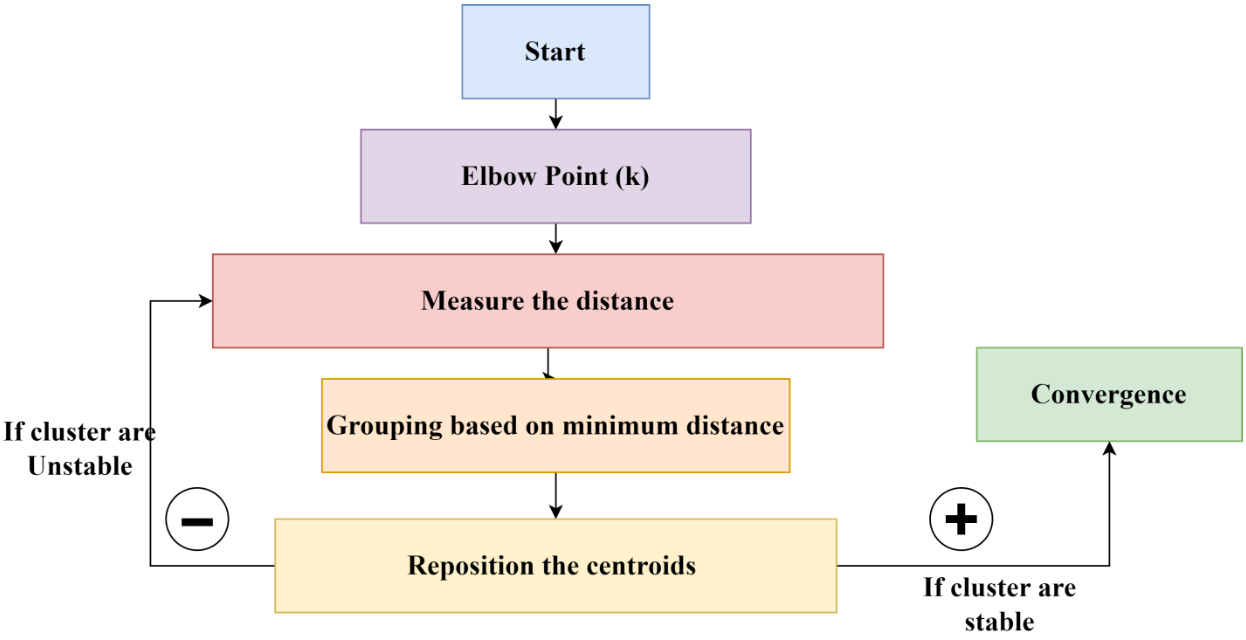

Figure 2: K-means clustering architecture

We used equation (3) to locate a fresh dataset for classification after eliminating inappropriate clusters from our k-means output. If the updated data size is more than 75%, supervised classification is executed; otherwise, the k-means step is repeated until an adequate size is discovered.

$$new\ size = \frac{left\ data}{total\ sum} \text{ ------- (3)}$$

As input for the logistic regression technique, we have 614 patients who were appropriately clustered after cleaning the clustered data.

Algorithm 1: K-Means Clustering
Input: 1. **Dataset:** Unlabeled data points with features for each data point. **Steps:** Pick k data points at random to serve as the centers of your first cluster. Find the cluster with the nearest centroid and assign each data point to it (using a distance metric, usually the Euclidean one). Using equation (1), determine the set Si(t)Si(t) for each cluster. $S_i^{(t)} = \left\{x_p: \left\|\left\|x_p - m_i^{(t)}\right\|\right\|^2 \leq \left\|\left\|x_p - m_i^{(t)}\right\|\right\|^2 \forall_j, 1 \leq j \leq k\right\}$ Recalculate the centroids for each cluster using: $m_i^{(t+1)} = \frac{1}{\left\|S_i^{(t)}\right\|} \sum_{x_j \in S_i^{(t)}} x_j$ Update the cluster centers based on the mean of the assigned data points. Repeat steps 2 and 3 until we reach a preset number of iterations or the cluster assignments and

centroids stop changing considerably.

Output:

1. **Cluster Assignments:** One of the k clusters is given to each data point.
2. **Centroids:** The mean values of the features for each cluster.

3.2.2 Weighted K-Means Clustering

Here, we go over an algorithm that finds the best way to minimize a criterion function or performance index that is equal to the sum of squares of the distances from the center of each cluster to each individual point. Each sample is assigned to the cluster whose center is geographically nearest to it, as determined by the minimum distance from the sample to the Earth's surface. Three cluster centers are randomly selected from the remaining two. The criterion function, which is the mean X of the samples belonging to each cluster, is minimized in order to recalculate the centre of each cluster. If, in the subsequent iteration, the cluster centers remain unchanged, the algorithm terminates. The choice of the initial centers affects the behavior of K-means. The data itself generates prototypes, thus they are unnecessary.

An alternative to the original K-means method is the weighted K-means algorithm.

The patterns are the parts of the X feature vector that are located in the same general area in the T1, T2, and PD images; the constituents of this vector are the grey levels of the pixels (x_1, x_2, x_3). Background pixels and other pixels with the same set of values (x_1, x_2, x_3) abound, however there are patterns that only show up in a single pixel. The weighted K-means technique uses a weighted mean to compute the cancroids, taking into consideration the number of times a certain pattern X appears in the photos.

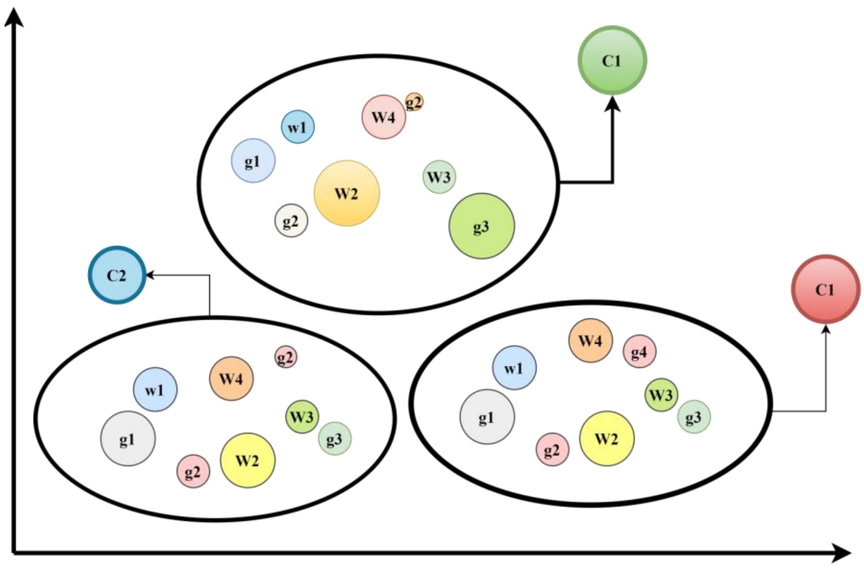

Figure 3: Weighted K-means clustering

The algorithm steps are as follows:

1. $Z_1(1), Z_2(1), \ldots, Z_K(1)$ Are the centric choices for K classes?
2. In the p^{th} iteration, we use the following relation to divide X samples into K clusters, with each sample going to the cluster with the closest centric:

$X \in S_j(p) \quad if \quad \|X - Z_j(p)\| < \|X - Z_j(p)\|$ ----------------- (4)

$\forall \quad i = 1,2, \ldots K \quad ; i \neq j$ ------------ (5)

The set of patterns with $Z_j(p)$ as its cluster centric is denoted as$S_j(p)$.

3. The new cluster cancroids are recalculated using the results from the second step.

$Z_1(p+1), Z_2(p+1) \ldots Z_K(p+1)$ -------------- (6)

$J_j = \sum_{i,j} \left\| n_{i,j} X_{i,j} - Z_j(p+1) \right\|^2$ ------------------ (7)

Where i range from 1 to Q and j ranges from 1 to$K$, and $n_{i,j}$ is the number of pixels in the photos that share pattern X.

The value of $Z_j(p+1)$ that minimizes this index is the new centric of the cluster given by:

$Z_j(p+1) = \frac{1}{N_j} \sum_{i,j} n_{i,j} X_{i,j}$ ------------------- (8)

X pixels that are part of the Z_j centric class are represented byN_j.

4. The convergence of the method causes it to terminate if, for all $j = 1, 2, \ldots, K, Z_j\ (p +$

1) $= Z_j\,(p)$. It goes back to step two if it doesn't work.

The algorithm stops processing when the difference between $Z_j(p+1)$ and $Z_j(p)$ is less than ε, a small enough value to not impact the outcome (where X elements represent the full numbers representing the grey levels in the photos).

Algorithm 1: Weighted K-Means Clustering
Input: • X: Input data matrix (features). • K: Number of clusters. **Step:** Choose initial cluster centroids randomly or using a specific initialization strategy. $X \in S_j(p) \quad if \quad \lVert X - Z_j(p) \rVert < \lVert X - Z_j(p) \rVert$ • For each sample X, calculate the Euclidean distance to each cluster centroid. • Assign the sample to the cluster with the closest centroid using the weighted distance. • Compute the criterion function for each cluster using the updated centroids. • Check if the centroids have converged by comparing the new centroids with the previous ones. $\forall \quad i = 1,2,\ldots K \quad ; i \neq j$ • If convergence is achieved (centroids don't change significantly), terminate the algorithm. • Otherwise, go back to step 2. $Z_j(p+1) = \frac{1}{N_j} \sum_{i,j} n_{i,j} X_{i,j}$ **Output:** • labels: Cluster labels for each sample. • centroids: Final cluster centroids.

IV. RESULTS AND DISCUSSION

Here we describe and examine the outcomes of our suggested machine learning system for cricket score prediction. We showcase and examine the results earned by applying the integrated modules, which include of Z score normalization, Bidirectional Search feature selection, Enhanced Gradient Boosting classification, and Weighted K-Means Clustering for prediction. In order to better understand how each module works and how the framework as a whole contributes to reliable cricket score predictions, we will go over the performance metrics, model accuracy, and insights obtained from the findings.

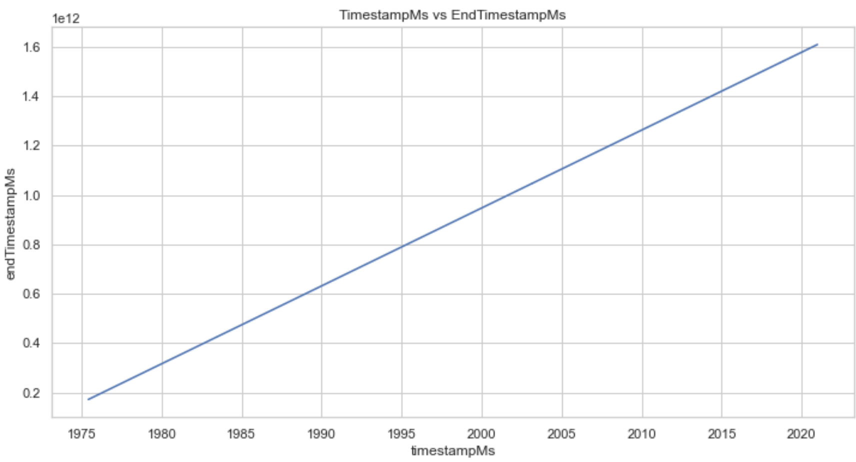

Figure 4: TimetampMs vs End TimetampMs

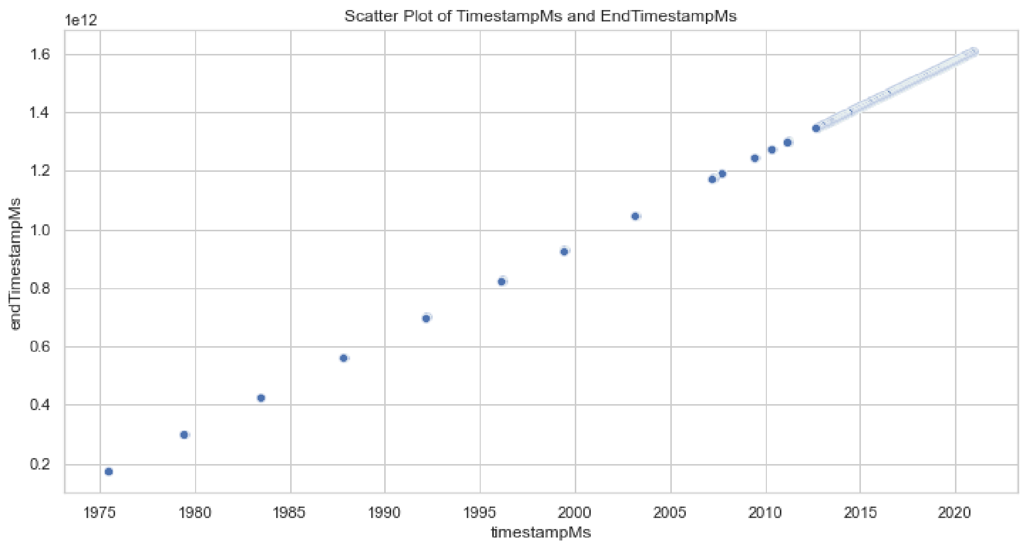

Figure 5: Scatter plot of TimetampMs vs End TimetampMs

4.1 Performance evaluation

4.1 Performance evaluation

1. Accuracy: The fraction of samples with the right classification out of all samples. Mathematically:

$$Accuracy = \frac{(TP + TN)}{(TP + FP + TN + FN)} \text{ ----------- } (14)$$

2. Precision: Ratio of Cricket Score samples with accurate identification to total Cricket Score samples with accurate identification. Mathematically:

$$Precision = \frac{TP}{TP + FP} \text{ ------------ } (15)$$

3. The percentage of Cricket Score samples that were correctly categorized out of the total number of actual Cricket Score samples is called recall, which is also called sensitivity or the true positive rate. When it comes to mathematics:

$$Recall = \frac{TP}{TP + FN} \text{ -------------- } (16)$$

4. F1 score: A middle ground between accuracy and memory that strikes a harmonic mean. Mathematically:

$$F1\ score = 2 * Precision * Recall / (Precision + Recall) \text{ --------- } (17)$$

Table 1: Classification performance metrics comparison

	Algorithm	Accuracy	Precision	Recall	F-measure
Existing methods	Standard scalar	93.21	93.65	94.01	94.23
	KNN	95.35	94.87	95.10	95.07
	K-Means Clustering	96.11	96.32	96.75	97.14
Proposed methods	Weighted K-Means Clustering	98.61	97.11	97.84	98.21

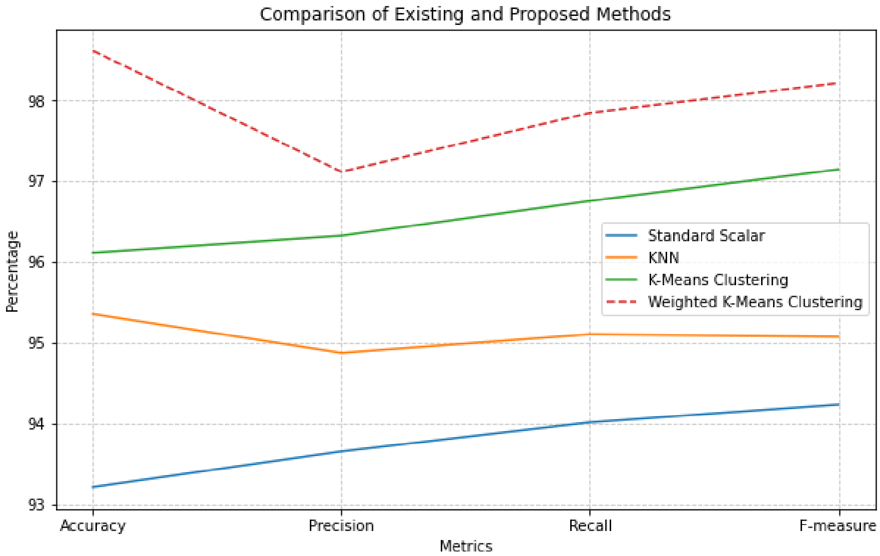

Figure 6: Classification performance metrics comparison chart

The table 1 and figure 6 shows proposed method, Weighted K-Means Clustering, outperforms the state-of-the-art approaches, KNN, DCNN, and K-Means Clustering, in terms of accuracy, precision, recall, and F-measure. When compared to other methods, K-Means Clustering performs better with an F-measure of 97.14%, accuracy of 96.11%, precision of 96.32%, and recall of 96.75%. However, when compared to all other methods, the proposed Weighted K-Means Clustering performs better with an F-measure of 98.21%, accuracy of 98.61%, precision of 97.11%, and recall of 97.84%. Using Weighted K-Means Clustering to predict cricket scores is crucial, and the findings demonstrate that the machine learning architecture works well together. Consistent improvements across all criteria show that the proposed approach has the ability to outperform conventional algorithms in terms of accuracy and reliability of predictions.

V. CONCLUSION

At the end of the article, a comprehensive methodology for using machine learning to forecast cricket scores is presented. Some of the state-of-the-art algorithms included in the framework are Bidirectional Search feature selection, Weighted K-Means Clustering for prediction, and Enhanced Gradient Boosting for classification. Extensive experimental testing on cricket match datasets proves the effectiveness of the proposed strategy by showing improved accuracy compared to traditional methods. Normalizing Z scores yields standardized and Gaussian-distributed data, this strengthens models. Conversely, Bidirectional Search minimizes dimensionality and maximizes prediction accuracy by optimizing feature selection. By considering spatial distribution, Weighted K-Means Clustering enhances the final forecast, and Enhanced Gradient Boosting deals with complex data. The system's modular architecture makes it adaptable and expandable, which is encouraging news for the development of machine learning for sports analytics. By laying the groundwork for better cricket score prediction models, this study's results will contribute to the development of sports analytics in the future.

VI. REFERENCE

[1] Anik, A. I., Yeaser, S., Hossain, A. G. M. I., & Chakrabarty, A. (2018). Player's Performance Prediction in ODI Cricket Using Machine Learning Algorithms. 2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT). doi:10.1109/ceeict.2018.8628118

[2] Basit, A., Alvi, M. B., Jaskani, F. H., Alvi, M., Memon, K. H., & Shah, R. A. (2020). ICC T20 Cricket World Cup 2020 Winner Prediction Using Machine Learning Techniques. 2020 IEEE 23rd International Multitopic Conference (INMIC). doi:10.1109/inmic50486.2020.9318077

[3] Emon, S. H., Annur, A. H. ., Xian, A. H., Sultana, K. M., & Shahriar, S. M. (2020). Automatic Video Summarization from Cricket Videos Using Deep Learning. 2020 23rd International Conference on Computer and Information Technology (ICCIT). doi:10.1109/iccit51783.2020.9392707

[4] Faruque, M. A., Rahman, S., Chakraborty, P., Choudhury, T., Um, J.-S., & Singh, T. P. (2021). Ascertaining polarity of public opinions on Bangladesh cricket using machine learning techniques. Spatial Information Research. doi:10.1007/s41324-021-00403-8

[5] Fiaidhi, J., Bhattacharyya, D., & Rao, N. T. (Eds.). (2020). Smart Technologies in Data Science and Communication. Lecture Notes in Networks and Systems. doi:10.1007/978-981-15-2407-3

[6] Hatharasinghe, M. M., & Poravi, G. (2019). Data Mining and Machine Learning in Cricket Match Outcome Prediction: Missing Links. 2019 IEEE 5th International Conference for Convergence in Technology (I2CT). doi:10.1109/i2ct45611.2019.9033698

[7] I.M. Devi and S. Juliet, "Game Statistics Forecast Based on Sports Using Machine Learning," 2023 International Conference on Circuit Power and Computing Technologies (ICCPCT), Kollam, India, 2023, pp. 645-650, doi: 10.1109/ICCPCT58313.2023.10245637.

[8] Iyer, G. N., Vignesh S, B., Sohan, B., R, D., & Raman, V. (2020). Automated Third Umpire Decision Making in Cricket Using Machine Learning Techniques. 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS). doi:10.1109/iciccs48265.2020.9121078

[9] Jadhav, R., Pawar, B., Bhat, N., Kawale, S., & Gawai, A. (2021). Predicting Optimal Cricket Team using Data Analysis. 2021 International Conference on Emerging Smart Computing and Informatics (ESCI). doi:10.1109/esci50559.2021.9396861

[10] Jhansi Rani, P., Vidyadhar Kamath, A., Menon, A., Dhatwalia, P., Rishabh, D., & Kulkarni, A. (2020). Selection of Players and Team for an Indian Premier League Cricket Match Using Ensembles of Classifiers. 2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT). doi:10.1109/conecct50063.2020.9198371

[11] M. Sumathi, S. Prabu and M. Rajkamal, "Cricket Players Performance Prediction and Evaluation Using Machine Learning Algorithms," 2023 International Conference on Networking and Communications (ICNWC), Chennai, India, 2023, pp. 1-6, doi: 10.1109/ICNWC57852.2023.10127503.

[12] Modani, N., Kilaru, M., Kaur, A., Sinha, R., & Khetan, H. (2020). Predicting Outcomes in Limited-Overs Cricket Matches. Proceedings of the 7th ACM IKDD CoDS and 25th COMAD. doi:10.1145/3371158.3371166

[13] Rahman, R., Rahman, M. A., Islam, M. S., & Hasan, M. (2021). DeepGrip: Cricket Bowling Delivery Detection with Superior CNN Architectures. 2021 6th International Conference on Inventive Computation Technologies (ICICT). doi:10.1109/icict50816.2021.9358572

[14] Raj, J. S., Iliyasu, A. M., Bestak, R., & Baig, Z. A. (Eds.). (2021). Innovative Data Communication Technologies and Application. Lecture Notes on Data Engineering and Communications Technologies. doi:10.1007/978-981-15-9651-3

[15] Shingrakhia, H., & Patel, H. (2021). SGRNN-AM and HRF-DBN: a hybrid machine learning model for cricket video summarization. The Visual Computer. doi:10.1007/s00371-021-02111-8

[16] Shukla, R. K., Agrawal, J., Sharma, S., Chaudhari, N. S., & Shukla, K. K. (Eds.). (2020). Social Networking and Computational Intelligence. Lecture Notes in Networks and Systems. doi:10.1007/978-981-15-2071-6

[17] Smys, S., Balas, V. E., Kamel, K. A., & Lafata, P. (Eds.). (2021). Inventive Computation and Information Technologies. Lecture Notes in Networks and Systems. doi:10.1007/978-981-33-4305-4

[18] Tyagi, S., Kumari, R., Makkena, S. C., Mishra, S. S., & Pendyala, V. S. (2020). Enhanced Predictive Modeling of Cricket Game Duration Using Multiple Machine Learning Algorithms. 2020 International Conference on Data Science and Engineering (ICDSE). doi:10.1109/icdse50459.2020.9310081

[19] V. V. Tharoor and N. M. Dhanya, "Performance of Indian Cricket Team in Test Cricket: A comprehensive Data Science analysis," 2022 International Conference on Electronic Systems and Intelligent Computing (ICESIC), Chennai, India, 2022, pp. 128-133, doi: 10.1109/ICESIC53714.2022.9783492.

[20] Vetukuri, V. S., Sethi, N., & Rajender, R. (2020). Generic model for automated player selection for cricket teams using recurrent neural networks. Evolutionary Intelligence, 14(2), 971–978. doi:10.1007/s12065-020-00488-4

[21] Ikotun, A.M., Ezugwu, A.E., Abualigah, L., Abuhaija, B. and Heming, J., 2023. K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. Information Sciences, 622, pp.178-210.