www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2765-2770

e-ISSN : 2583-1062
Impact Factor : 7.001

PLAYER PERFORMANCE PREDICTION IN CRICKET USING MACHINE LEARNING CLASSIFIERS

Piyush Wankar[1], Kunal Dharmadhikari[2], Ms. Dipalee Borse[3]

[1,2]PG Student H V Desai College Pune, Maharashtra, India.

[3]Assistant Professor, H V Desai College Pune, Maharashtra, India.

piyushwankar2001@gmail.com, kunalkakade2019@gmail.com, dipa.borse@gmail.com

DOI: https://www.doi.org/10.58257/IJPREMS40404

ABSTRACT

Performance analysis of players in cricket is essential for identifying strengths and weaknesses, enabling data-driven decision-making for team selection and strategy. By evaluating player performance, teams can optimize player utilization and improve overall match outcomes.This research explores the use of machine learning algorithms in predicting cricket player's performance from historical match data. We used a dataset from Kaggle which included several batting and bowling statistics to train and test the models Support Vector Machine (SVM), Random Forest, Naïve Bayes, Gradient Boosting, Multilayer Perceptron (MLP) and K-Nearest Neighbors (KNN). The goal is to construct an accurate model capable of predicting and classifying players to the correct performance tier. Random Forest, Naïve Bayes, and Gradient Boosting also performed well with 0.92 accuracy. The research demonstrates how machine learning can effectively support team selection, player development, and strategic planning in cricket

Keywords: Cricket Analytics, Machine Learning, Player Performance Prediction.

1. INTRODUCTION

Integrating strategic and physical aspects with skill, cricket stands as a profound dynamic sport. However, the subclassification of strategies within teams and even figuring out how an individual is contributing towards the performance require detailed analysis of players in respective teams. Conventional approaches have relied, to differing extents, on subjective analyses, statistical summation of predefined numbers, and cloak-and-dagger insights by so-called specialists. The emergence of data analytics and machine learning techniques is shifting the focus towards performance evaluation which is objective and empirically verifiable.

In order to accomplish this objective, a comprehensive structure for performance evaluation of cricket players is proposed by tailoring specific algorithms such as Random Forest, Support Vector Machine (SVM), MLP, Gradient Boosting, and KNN. Under this framework, batting and bowling statistics (e.g., runs, wickets, strike rate, and economy) are categorized to allow classification of performance into different levels. The scope of this research is to develop a sophisticated performance prediction model that amalgamates all relevant data and supports seasoned coaches and analysts in evidence-based decisions alongside covering the needs of the players.

This study focuses on how various statistical measures that are captured during the course of the match can be encoded to construct machine-perceivable models that subsequently classify players into distinct performance tiers (high, medium, low).

Using the ensemble learning technique of Random Forest and the classifier SVM is advantageous as they both efficiently manage diverse feature interactions, highly complex datasets, and provide accurate predictions.

This study may prove useful for cricket analytics, improving the objectivity and accuracy of player assessments. Additionally, it illustrates the impact of machine learning as a tool in sports analytics, which can inspire advanced research and applications not only in cricket, but across the spectrum of sports activities.

2. LITERATURE REVIEW

This review analyzes the use of machine learning (ML) in cricket, delving into player performance evaluation, team composition, and match result prediction. There are multiple cricket datasets that scholars apply with ML algorithms like Random Forest, Support Vector Machines (SVM), Decision Trees, Naïve Bayes, and neural networks to enhance prediction accuracy in team and player analytics. With regards to forecasting, predicting player performance is at the core of team composition studies.The authors [1] used a weighted Random Forest model and achieved 93.73 % accuracy by integrating weather conditions with player achievement data. The innovation brought forth by this work demonstrated the power of ML in improving selection decisions. Similarly, in [4] comparative study of algorithmic prediction of runs and wickets has been performed using Random Forest and SVM. The results showed that Random Forest provided the highest accuracy predicting 92.25% for wickets and greatly informing team management.

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2765-2770

e-ISSN : 2583-1062
Impact Factor : 7.001

The optimization of team selections has also been explored by studying players' weaknesses and strengths or against their opponents. A balanced team recommendation system, based on player fitness and historical performance data, that identified optimal line-ups[3]. Another model categorized players into "performers," "moderates," and "failures" when selecting players; this model has been very successful in ranking players' potential. However, the study revealed gaps in data for fielding and wicket keeping metrics, and calling for more detailed metrics to ensure an optimal opportunity to leverage ML.Outcome prediction of matches, specifically for tournaments like the IPL, is yet another emerging area of research study. One study used Decision Trees, Random Forest, and XGBoost models to enhance the prediction that measured IPL winners based on past season data. After tuning hyperparameters, predictive accuracy was highest with the XGBoost model[3].

Another study created a Cricket Outcome Predictor (COP) tool, for One Day International (ODI) based on game characteristics inclusive of toss and order of innings. The analyses both concluded that Random Forest and SVM models consistently predicted a match's outcome[2]. Although, both studies noted that a rigorous data pre-processing approach was required to ensure the highest level of consistency in prediction models.

Shared patterns within the literature summarize the utility of advanced preprocessing, feature engineering, and hyperparameter tuning to confront issues such as data imbalance. For instance, a study, exploring bowler performance, utilized hyperparameter tuning (with GridSearchCV) to enhance model generalizability and accuracy.In conclusion, machine learning methods, especially ensemble methods like Random Forest and XGBoost, demonstrate considerable potential for use in cricket analytics to improve predictions related to player performance, team selection, and match results. Despite data limitations (e.g., limited metrics for specific cricket roles), the continued expansion of the data itself, coupled with feature engineering, will help to enhance predictive accuracy further, across all formats of cricket[3].

3. METHODOLOGY

3.1 Data Collection:

The dataset is sourced from Kaggle[8]. The dataset is divided in 3 parts, batting, bowling, and match. Batting data contains 35 records and 28 features whereas bowling data includes 15 records and 30 features. The data related to the match has 6120 records and 15 features.

3.2 Data Preprocessing:

To make sure that the dataset was clean, reliable, and apt for training machine learning models, a number of preprocessing steps were taken. The steps assist in improving data quality, eliminating noise, and making the data ready in a well-structured form that algorithms can learn from in an efficient manner. The preprocessing step was very important to help make accurate and meaningful predictions.

The tasks covered and carried out under this phase are as follows:

- **Handling Missing Values**:
  All records containing incomplete or missing information were removed to maintain the integrity of the dataset and ensure accurate results during analysis.
- **Removing Duplicate Records**:
  Repetitive entries were identified and eliminated to prevent redundancy and avoid skewed performance metrics.
- **Correcting Inconsistencies** :
  Although not implemented in the current phase, further refinement may include standardizing text formats (such as names, dates, and categories) to ensure uniformity across the dataset and improve the reliability of data aggregation and analysis.

3.3 Feature Engineering:

Relevant features were selected, and new performance metrics were created, including a **batting performance** score based on runs, boundaries, and strike rate, and a **bowling performance** score based on wickets, runs conceded, and overs bowled. These metrics were then merged to provide an overall performance score for each player.

3.4 Data Splitting:

- The dataset is often divided, allocating approximately 70% of the data for training the model and the remaining 30% for evaluating its performance.
- A 70:30 data split aims to provide a substantial training set for model learning while reserving a sufficient test set to accurately assess the model's generalization to unseen data.

www.ijprems.com
editor@ijprems.com
INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2765-2770
e-ISSN : 2583-1062
Impact Factor : 7.001

3.5 Machine Learning Algorithms:

The machine learning algorithms below were used and evaluated.

3.5.1. Support Vector Machine (SVM):

SVM is a supervised machine learning algorithm widely used for classification and regression tasks. It works by mapping data into a multidimensional space and constructing a hyperplane that best separates the classes. The optimal hyperplane is the one that maximizes the margin between the two classes. Data points closest to the hyperplane are called support vectors. The objective is to find the hyperplane that maximizes the margin and minimizes the classification error. The hinge loss function is used in SVM to penalize misclassified points.

c(x,y,f(x)) = { 0 if y·f(x)≥1

1 - y . f(x) otherwise

3.5.2. Naive Bayes:

Naive Bayes is a probabilistic classifier based on Bayes' Theorem, which predicts the class of a given data point using the likelihood of features under each class. It assumes that all features are conditionally independent given the class label, which simplifies computation. The algorithm computes the posterior probability of each class and selects the class with the highest probability.

Bayes' Theorem is represented as:

$$P(y \mid X) = \frac{P(y) \cdot P(X \mid y)}{P(X)}$$

Where y is the class variable, X={x_1,x_2,...,x_n} are the set of features.

3.5.3. Random Forest:

Random Forest is an ensemble learning method that combines multiple decision trees to improve classification accuracy and reduce overfitting. It creates each tree using a random subset of features and data (bootstrap sampling), ensuring model diversity. The final prediction is made through majority voting in classification or averaging in regression. By averaging multiple models, Random Forest reduces variance and increases robustness. While no specific equation defines the whole forest, the underlying base model remains a decision tree built on information gain or Gini index. The prediction for classification can be expressed as:

$$\hat{y} = mode\{T_1(x), T_2(x), \dots, T_k(x)\}$$

Where $T_i(x)$ is the prediction from the i^{th} decision tree.

3.5.4. Gradient Boosting:

Gradient Boosting is an ensemble technique that builds models sequentially, where each new model aims to reduce the errors made by the previous one. It uses a gradient descent approach to minimize a specified loss function. At each stage, a new model is trained to predict the negative gradient (residuals) of the loss function with respect to the current model's output. These weak learners, typically decision trees, are added together to form a strong model.

Let Fm(x)F_m(x)Fm(x) be the model at stage mmm, then:

$$F_m(x) = F_{m-1}(x) + \alpha h_m(x)$$

Where $h_m(x)$ is the new base learner and α is the learning rate. The model minimizes the loss:

$$\underset{F}{argmin} \sum_{i=1}^{n} \quad L\big(y_i, F(x_i)\big)$$

3.5.5. Multilayer Perceptron (MLP):

MLP is a type of feedforward artificial neural network that contains one or more hidden layers between input and output layers. Each neuron in a layer is connected to every neuron in the next layer. The neurons apply a non-linear activation function to learn complex patterns. MLP is trained using backpropagation, where the error is propagated backwards to update the weights. The network attempts to minimize the difference between the actual and predicted output using a loss function. A commonly used activation function is the sigmoid function:

$$h_\theta(x) = \frac{1}{1 + e^{-\theta^T x}}$$

This function is used to map any real-valued number into the (0,1) interval, making it suitable for binary classification tasks.

www.ijprems.com
editor@ijprems.com

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2765-2770

e-ISSN : 2583-1062
Impact Factor : 7.001

3.5.6. K-Nearest Neighbors (KNN):

KNN is a non-parametric, instance-based learning algorithm. It assigns a class to a new data point based on the majority class among its K nearest neighbors in the feature space. The similarity between points is measured using a distance metric, most commonly the Euclidean distance. The algorithm does not require a training phase, making it simple yet computationally intensive for large datasets.

The Euclidean distance between two points x and x' is given by:

$$d(x, x') = \sqrt{(x_1 - x_1')^2 + (x_2 - x_2')^2 + \cdots + (x_n - x_n')^2}$$

The predicted probability of class j is:

$$P(X = x) = \frac{1}{K} \sum_{i \in A} I\left(y^{(i)} = j\right)$$

Where A is the set of K nearest neighbors and I is the indicator function.

3.6 Model Evaluation:

Model performance was evaluated using:

Accuracy: Accuracy is the percentage of instances classified correctly to total predictions. It gives an overall sense of how "correct" the model is.

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN}$$

Precision: Precision measures how many of the positively predicted instances were indeed correct. This metric is useful when the false positive rate should be minimized.

$$Precision = \frac{TP}{TP + FP}$$

Recall: Recall is the number of the actual positive instances that were predicted correctly. It helps to evaluate how well the model was able to recall all relevant cases.

$$Recall = \frac{TP}{TP + FN}$$

F1-Score: The F1-Score is the harmonic mean of precision and recall. It takes into account both metrics and is advantageous where data are found to be imbalanced.

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

3.7 Software and Tools:

- Python 3.12
- Jupyter Notebook
- Pandas, NumPy, Scikit-learn, Matplotlib

4. RESULTS & DISCUSSION

We assessed the accuracy of several models to measure their performance. Random Forest, Naïve Bayes, and Gradient Boosting models were the top performers with an accuracy of 0.92, demonstrating strong predictive capabilities and generalizability. These models showed consistent and reliable results across the dataset. SVM and KNN produced very similar results (0.85), indicating moderate performance with potential for improvement through tuning or feature selection. MLP performed the worst with an accuracy of 0.77 and can likely be eliminated from future modeling, as it probably does not generalize well for this dataset.

Algorithm	Precision	Recall	F1- Score	Accuracy
Random Forest	0.92	1.00	0.96	0.92
Naive Bayes	0.92	1.00	0.96	0.92
SVM	0.85	1.00	0.92	0.85

INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)
(Int Peer Reviewed Journal)
Vol. 05, Issue 04, April 2025, pp : 2765-2770
www.ijprems.com
editor@ijprems.com
e-ISSN : 2583-1062
Impact Factor : 7.001

KNN	0.85	1.00	0.96	0.92
Gradient Boosting	0.92	1.00	0.96	0.92
MLP	0.77	1.00	0.87	0.77

More analysis and testing needs to be conducted to ensure the reliability and robustness of the models used. It is ideal that future work implements rigorous cross-validation methods, such as k-fold cross-validation, to better assess each model's true generalization performance. Additionally, exploring regularization techniques and fine-tuning hyperparameters could further enhance model stability and reduce the risk of overfitting.

5. CONCLUSION

This research has provided evidence that machine learning is an effective means to measure cricketer performance. Both Random Forest and Gradient Boosting provided high levels of accuracy. Future research should expand on data gathering and trial other algorithms for enhanced predictive performance.

6. FUTURE SCOPE & LIMITATIONS

6.1 Limitations:

The most significant limitation of this research is the potential for overfitting, and this is more likely when you use models which are very flexible or if you work with small, simple datasets. While I tried to identify and self-correct models I may have been overfitting during experimentation at this stage, there are things you could do to remedy these issues in future work. Going forward, more robust approaches like k-fold cross-validation could give you a more solid measure of model performance. Regularization, such as L1 and L2 regularization also could be used to reduce overfitting, although I used these regularization techniques while picking models I could have done more to reduce overfitting in task. Effective feature selection and reducing dimensionality may assist in resolving overfitting, and/or reduce the complexity of the models. In addition, training the models on more complex and larger datasets will provide more generalizability. Testing on more unseen data is important to really make sure the models are robust. The use of the Naïve Bayes algorithm may also present a limitation as the algorithm assumes independence between features of the data which may not be true in terms of the data in cricket because having one attribute/feature affects another and therefore the features are not independent.

In addition, other constraints may lie in the lack of inclusion of substantial vectors in the player performance modelling, including particular vectors such as weather conditions and player fitness which could impair the predictive capacities of the modelling. While the rationale for these models is accurate, they still require more rigorous assessment, and especially their adaptability in the different formats associated with cricket must be considered: Test matches, ODIs, and T20s.

6.2 Future Scope:

Future developments of this research could include additional contextual information like weather conditions, pitch descriptions, and player fitness levels, all of which have been shown to significantly alter player performance. Incorporating these variables would not only increase the accuracy of the model, but be more realistic. Additionally, exploring other machine learning methods such as more sophisticated deep learning models could be utilized in an effort to identify more complex patterns in the data and increase predictability. A more ambitious direction would be to create real-time prediction intelligence, which would provide insight in the course of live game play, and eventually take the model to an international platform, which would give the model more dynamism and larger reach.

7. REFERENCES

[1] Kapadiya, C., Shah, A., Adhvaryu, K., & Barot, P. (2020). Intelligent cricket team selection by predicting individual players' performance using efficient machine learning technique. Int. J. Eng. Adv. Technol, 9(3), 3406-3409.

[2] Passi, K., & Pandey, N. (2018). Increased prediction accuracy in the game of cricket using machine learning. arXiv preprint arXiv:1804.04226.

[3] Vistro, D. M., Rasheed, F., & David, L. G. (2019). The cricket winner prediction with application of machine learning and data analytics. International journal of scientific & technology Research, 8(09), 21-22.

[4] Biswas, M., Niamat Ullah Akhund, T. M., Mahbub, M. K., Saiful Islam, S. M., Sorna, S., & Shamim Kaiser, M. (2022). A survey on predicting player's performance and team recommendation in game of cricket using

machine learning. In Information and Communication Technology for Competitive Strategies (ICTCS 2020) ICT: Applications and Social Interfaces (pp. 223-230). Springer Singapore.

[5] Lokhande, R., Awale, R. N., & Ingle, R. R. (2024, December). Evaluation of Various Machine Learning Models in Forecasting Cricket Player Performance. In 2024 4th International Conference on Robotics, Automation and Artificial Intelligence (RAAI) (pp. 324-329). IEEE.

[6] Neeraj Pathak, Hardik Wadhwa,Applications of Modern Classification Techniques to Predict the Outcome of ODI Cricket,Procedia Computer Science(2016).

[7] Anik, A. I., Yeaser, S., Hossain, A. I., & Chakrabarty, A. (2018, September). Player's performance prediction in ODI cricket using machine learning algorithms. In 2018 4th international conference on electrical engineering and information & communication technology (iCEEiCT) (pp. 500-505). IEEE

[8] Dataset: https://www.kaggle.com/datasets/akarshsinghh/cricket-player-performance-prediction