https://doi.org/10.65091/icicset.v2i1.30

Evaluation of Machine Learning Techniques in Winner Predicting of The Hundred Games

Rudra Nepal
Department of Software Engineering
Nepal College of Information Technology
Lalitpur, Nepal
rudra.nepal@ncit.edu.np

Nishan Paudel
Department of Software Engineering
Nepal College of Information Technology
Lalitpur, Nepal
nishan.221627@ncit.edu.np

Anish Kumar Neupane
Department of Software Engineering
Nepal College of Information Technology
Lalitpur, Nepal
anish.221608@ncit.edu.np

Rishav Chapagain
Department of Software Engineering
Nepal College of Information Technology
Lalitpur, Nepal
rishav.221636@ncit.edu.np

Rushab Risal
Department of Software Engineering
Nepal College of Information Technology
Lalitpur, Nepal
rushab.221637@ncit.edu.np

***Abstract*—Cricket, being one of the most popular sports worldwide, attracted significant interest in developing accurate result prediction models. The Hundred is one of the several leagues that are contested in the world. It was important to research on accurate result prediction model in this league as the fan following and attention towards this league were increasing rapidly. The dataset was divided into training and test sets, and the models were evaluated on both datasets to measure their generalization performance. The findings demonstrated the potential of machine learning techniques in accurately forecasting Hundred match outcomes, enabling stakeholders to make informed decisions in the dynamic and unpredictable domain of cricket. Logistic Regression, Decision Tree and Random Forest models were implemented for match-winner prediction, while Gradient Boosting Regressor was used for score forecasting. This study gathered and analyzed Hundred data spanning multiple years, including player, match, team, and ball-to-ball information, to generate several conclusions that helped improve a team's performance. The study highlights the applicability of supervised learning methods for enhancing decision-making in the dynamic and unpredictable environment of The Hundred.**

***Index Terms*—Logistic regression, Gradient Boosting Regressor, Hundred Winning Prediction, Hundred Score Prediction, ball-to-ball statistics**

I. Introduction

The Hundred is a 100-ball cricket tournament involving teams in major cities across England and Wales run by the England and Wales Cricket Board (ECB) which took place for the first time in 2021. It features eight city-based teams from England and Wales, with both men's and women's competitions running concurrently. The format is designed to be fast paced and innovative, with each team batting for a maximum of 100 balls, making it shorter than traditional T20 matches. The tournament aims to attract a broader audience to cricket, combining high-energy matches with entertainment to appeal to fans of all ages.

Cricket has long been a sport driven by statistics, with player performance, team strategies, and game outcomes often influenced by a wealth of data. With eight city-based teams competing in a shorter, faster-paced format, The Hundred has garnered significant attention and fan engagement. Predicting match outcomes in this unique format presented an intriguing challenge due to its novelty, condensed structure, and dynamic gameplay. This study focused exclusively on the men's matches in The Hundred, leveraging machine learning techniques to predict results based on historical performance and other contextual factors. By analyzing the distinct features of The Hundred's format, predictive models were developed to identify patterns and trends in match outcomes. The findings provided valuable insights for analysts and stakeholders, highlighting the potential of machine learning in revolutionizing sports analytics.

A. Motivation

The game of cricket is highly unpredictable, with momentum shifting frequently between the two competing teams. Many matches are decided on the final ball, making outcome prediction a challenging yet engaging task. Given these unpredictable scenarios, there is significant interest among spectators in making predictions either at the start of the game or during the game itself. Thousands of enthusiastic fans participating in fantasy cricket leagues all over the world have the quest for gaining a competitive edge and maximizing team performance. To address this need, machine learning algorithms have gained considerable attention. Machine learning techniques, with their ability to analyze vast volumes of data and identify patterns, have the potential to revolutionize the world of cricket as well.

B. Problem Statement

Coaches and team management often face challenges in making data-driven decisions during matches in The Hundred. This includes adjusting tactics or modifying strategies to improve their chances of winning. Leveraging historical data, match statistics and other influencing factors, machine learning models can provide critical insights and predictions. Such models can also prove invaluable for fantasy cricket enthusiasts by helping them assemble competitive teams designed to outperform opponents and secure victories. However, there is currently very less study going on evaluating the efficiency of machine learning models specifically tailored to the unique context of The Hundred, leaving all related stakeholders without the effective ML tools needed to maximize their potential.

C. Objective

To evaluate machine learning models specifically tailored to predict match outcomes in the men's competition of The Hundred, incorporating the unique rules and characteristics of the tournament.

II. Literature Review

Machine learning has increasingly been employed in sports analytics to predict match outcomes, evaluate player performance, and provide actionable insights for stakeholders. Research in sports analytics has experimented with a range of machine learning algorithms. Studies typically use multiple models to compare performance. Common algorithms include Logistic Regression, Decision Trees, Gradient Boosting, Neural Networks. Cricket datasets typically combine structured data from sources such as: Ball-by-Ball Data, Match Info Data, Player Statistics and were often imbalanced and highly granular.

A. Related Works

Early studies focused primarily on descriptive statistics and visualization rather than predictive modeling. For example, the work in [3] emphasized team selection and profit strategies using IPL data but did not evaluate machine learning models for outcome prediction, limiting its applicability for real-time decision support. Several researchers have employed traditional classification algorithms such as Logistic Regression and Naïve Bayes due to their simplicity and interpretability [1], [2]. While these models provide reasonable baseline performance, they assume linear decision boundaries and independence among features. Such assumptions are often violated in cricket, where match outcomes depend on complex, non-linear interactions between runs, wickets, overs, and contextual pressure. This limitation explains the moderate accuracies reported in prior studies and motivates the use of more expressive models.

Tree-based and ensemble learning approaches have consistently demonstrated superior performance in cricket prediction tasks. Studies such as [7] and [8] report that Decision Trees, Random Forests, and Gradient Boosting outperform linear models by effectively capturing non-linear feature relationships. However, many of these studies rely on large datasets comprising thousands of IPL matches, raising concerns about scalability and generalization when applied to newer leagues with limited historical data. Additionally, extremely high reported accuracies suggest potential overfitting, especially when cross-validation or pruning strategies are insufficiently discussed. Score prediction has proven to be more challenging than match outcome classification. Gradient Boosting Regression has been applied to cricket score forecasting with mixed success [4]. While ensemble regressors reduce bias and variance, their effectiveness diminishes in highly volatile formats. The Hundred's condensed 100-ball structure encourages aggressive batting and frequent momentum shifts, resulting in score distributions that are harder to model using aggregate features alone. Prior work largely overlooks this issue by focusing on longer formats such as ODI and T20, where scoring patterns are relatively stable.

Player-centric studies [5], [6] provide valuable insights into individual and team performance but are primarily analytical rather than predictive. These approaches do not directly address match-level outcome prediction and often require granular player form data that may not be available in real time. Similarly, fantasy cricket prediction frameworks [9] demonstrate the usefulness of data-driven decision-making but prioritize player selection over match result forecasting.

A notable gap in existing literature is the lack of studies specifically tailored to The Hundred tournament. Most models are developed and validated using IPL or international cricket data, assuming transferability across formats. However, The Hundred introduces unique rules, shorter innings, and different strategic dynamics, rendering direct application of existing models ineffective. Furthermore, many studies fail to justify methodological choices such as feature selection, dataset size, and model complexity, leaving unanswered questions regarding why certain approaches succeed or fail.

In contrast, this study evaluates commonly used machine learning models within the constrained and highly dynamic context of The Hundred. By critically analyzing model performance, overfitting risks, and error behavior, this research addresses key limitations in prior work. The findings highlight that while tree-based classifiers are effective for match outcome prediction even with limited data, score prediction remains an open challenge requiring richer contextual features and temporal modeling.

III. Methodology

A. Dataset

The data was collected from the Cricsheet website, where raw, uncleaned ball-by-ball data of every cricket game is available. The website is inspired by the Retrosheet website. The data format consists of two files per match. The files for a match are named `id_info.csv` (for the match information) and `id.csv` (for the ball-by-ball data), where `id` is the Cricinfo ID for the match. The ball-by-ball file contains one row per delivery in the match, along with the list of players

participating in the match, date the match was played, and the outcome.

The info file contains metadata about the actual match, such as when and where it was played, event details, and the match type. The fields included in the info section appear as one or more rows in the data. Some of the fields are required, while others are optional. If a field has multiple values (e.g., `team`), each value appears in its own row.The first row of each ball-by-ball CSV file contains the headers, with each subsequent row providing details on a single delivery. The headers in the file are:

```
match_id, season, start_date, venue,
innings, ball, batting_team,
bowling_team, striker, non_striker, bowler,
runs_off_bat, extras, wides,
noballs, byes, legbyes, penalty, wicket_type,
player_dismissed,
other_wicket_type, other_player_dismissed,
over
```

Overall, data from 133 matches spanning from 2021 to 2024 were collected. All ball-by-ball data were combined into a single CSV file, and all match info data were combined into another single CSV file.

B. Models Used

Deep learning models were not employed due to the limited dataset size (133 matches), which is insufficient for stable training of sequence-based architectures. This study employed a set of supervised machine learning models for two prediction tasks: match-winner classification and team score regression. Each model was integrated through preprocessing pipelines that handled one-hot encoding of categorical features and direct processing of numerical variables. The models predicts win probability during live second-innings scenarios, not pre-match outcomes.

Logistic Regression: A linear classification model used as the baseline for winner prediction. It estimates the probability of a team winning using a logistic function. In this study, it was implemented within a preprocessing pipeline and evaluated using accuracy, precision, recall, and F1-score.

Decision Tree Classifier: A tree-based model that splits data into hierarchical decision rules. It was applied to capture non-linear relationships in match-outcome prediction. The model was trained on processed features and assessed using standard classification metrics and confusion matrix analysis.

Random Forest Classifier: An ensemble of multiple decision trees designed to reduce variance and improve predictive stability. In this study, it provided more robust winner predictions compared to a single tree, achieving high accuracy and strong generalization in evaluation.

Gradient Boosting Regressor: A boosting-based model used for score prediction. It builds sequential weak learners to minimize prediction error and capture complex run-scoring patterns. The model was evaluated using MAE, MSE, RMSE and R^2, and supported by diagnostic plots such as predicted vs actual etc.

C. Feature Selection

Effective feature selection is critical for building reliable prediction models in a dynamic and short-format tournament such as The Hundred. This study adopted a domain-driven feature engineering approach, ensuring that all selected features were interpretable, non-redundant, and computable in real time from ball-by-ball data.

1) Features for Match Winner Prediction: For match outcome classification, nine features were selected to capture team context and live match state during the second innings. Categorical variables included *batting_team*, *bowling_team*, and *city*, which encode team strength, historical performance, and venue-related effects. Numerical match-state features comprised *runs_left*, *balls_left*, *wickets_left*, and *total_runs_x*, representing target difficulty, time pressure, and remaining batting resources. Additionally, two derived rate-based features, *current run rate* and *required run rate*, were included to quantify batting momentum and chase pressure. Redundant or non-informative attributes such as match identifiers, current score, delivery numbers, and player-level details were excluded, as their information content was already captured through aggregate and derived features. This compact feature set effectively represents chase dynamics and achieved strong predictive performance, particularly with tree-based classifiers.

2) Features for Score Prediction: For score prediction, a separate feature set was designed to model run-scoring behavior independent of chase context. Categorical variables included *batting_team*, *bowling_team*, and *city*, while numerical features consisted of *over*, *balls_left*, *wickets_left*, and *current run rate*. These variables capture match phase, urgency, batting depth, and scoring momentum. Target-dependent features such as *runs_left* and *required run rate* were excluded to avoid information leakage, as score prediction focuses on estimating run output rather than match outcome. The selected features emphasize temporal and contextual scoring patterns characteristic of The Hundred format.

Overall, the feature selection strategy balances domain knowledge with statistical efficiency, enabling accurate winner prediction while highlighting the inherent complexity of score forecasting in a fast-paced 100-ball tournament.

D. Implementation Details

1) Importing the Relevant Libraries: After the design was built, it was time to start implementing the system. Several essential libraries were imported to facilitate data handling, preprocessing, model training, and visualization. Pandas was used for reading and manipulating the dataset, while NumPy provided numerical computing support. For data visualization and exploratory analysis, Seaborn and Matplotlib were employed.

The Scikit-learn library played a crucial role in machine learning tasks, offering tools for preprocessing, model selection, feature transformation, and both classification and regression models. Specific functions such as `OneHotEncoder` for categorical encoding, `LogisticRegression`, `Random Forest`

`Classifier and Decision Tree` for win probability prediction, and `GradientBoostingRegressor` for score prediction were utilized.

2) Reading and Loading the Dataset and Exploratory Data Analysis: The dataset was loaded using Pandas, enabling efficient handling and manipulation of the data. The shape, features, and column names were examined to understand the structure of the dataset. Exploratory Data Analysis (EDA) was conducted to gain insights into key variables, their distributions, and important relationships.

Unique values in categorical fields such as team names and match venues were identified to ensure consistency. Missing values were checked, and appropriate strategies were planned for handling them.

3) Data Preprocessing, Train–Test Split, and Model Implementation: The dataset underwent preprocessing to ensure clean and structured input for machine learning models. Grouping and merging operations were performed to streamline the data, and team names were standardized. Special handling was applied to matches affected by the Duckworth-Lewis method. Null values were addressed, and the target variable analysis was performed. The data was then split into training and testing sets to enable effective model training and evaluation.

`OneHotEncoder` was applied to categorical variables to convert them into machine-readable numerical form. For prediction tasks, Logistic Regression was used for win probability estimation, while Gradient Boosting Regression was implemented for score prediction. Random Tree Classifier and Decision Tree Classifier were tested in terms of win probability estimation. The trained models were saved as pickle files (`pipe.pkl` and `score.pkl`) and integrated into a Streamlit-based user interface to enable real-time predictions.

IV. Result and Discussion

A. Models Evaluation

TABLE I
WIN PREDICTION MODEL PERFORMANCE

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	63%	0.61	0.57	0.59
Decision Tree	100%	1.00	1.00	1.00
Random Forest	100%	1.00	1.00	1.00

TABLE II
SCORE PREDICTION MODEL PERFORMANCE

Model	MAE	MSE	RMSE	R^2
Gradient Boosting Regressor	1.22	2.62	1.62	0.05

It is important to note that the regression model was trained at the ball-by-ball (delivery) level, and therefore all reported error metrics (MAE, MSE, RMSE) represent average prediction error per delivery, not per innings or per match.

B. Confusion Matrix Analysis

The confusion matrix summarizes the classification performance using True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). Higher values along the diagonal indicate better predictive performance.

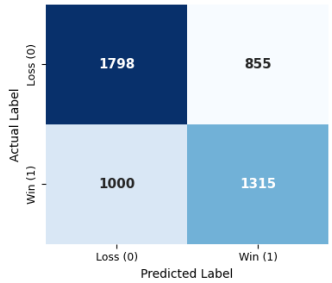

Fig. 1. Confusion Matrix for Logistic Regression

Logistic Regression shows relatively weak performance with a high number of misclassifications. The large number of false positives and false negatives indicates limited class separability, resulting in lower accuracy and F1-score compared to tree-based models.

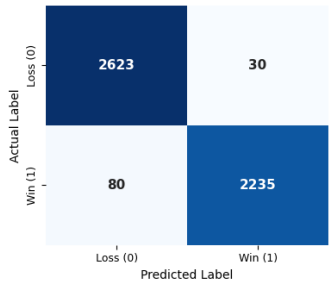

Fig. 2. Confusion Matrix for Decision Tree Classifier

Decision Tree Classifier achieves significantly improved results with very low false positives and false negatives. While the accuracy is high, such near-perfect performance may suggest potential overfitting to the training data.

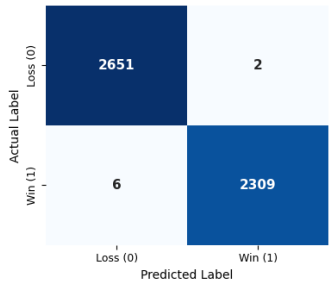

Fig. 3. Confusion Matrix for Random Forest Classifier

Random Forest Classifier demonstrates the best overall performance, with almost no misclassifications. The ensemble nature of the model improves generalization and robustness,

making it the most reliable classifier among the evaluated approaches.

C. Error Analysis of the Score Prediction Model

To further evaluate the performance of the Gradient Boosting Regressor in predicting the total runs scored in The Hundred matches, three diagnostic plots were generated: (1) Predicted vs Actual values, (2) Residual Distribution, and (3) Residuals vs Predicted values. These visualizations help assess not only the accuracy of the model but also the patterns and biases in its prediction behavior.

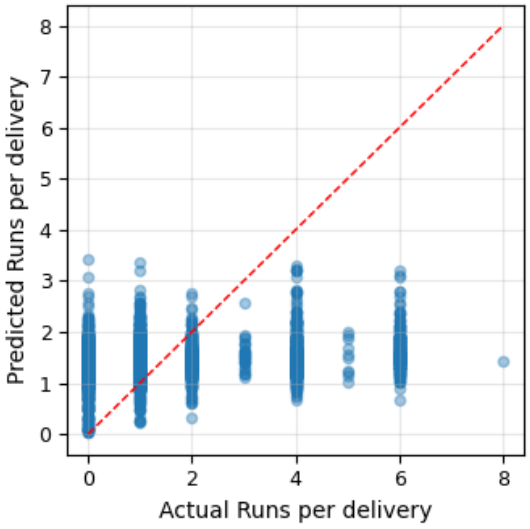

Fig. 4. Predicted vs Actual Runs per delivery for the Gradient Boosting Regressor.

1) Predicted vs Actual Analysis: Figure 4 shows the relationship between actual and predicted *runs per delivery*. The points are tightly clustered, indicating that the model predicts within a narrow range (approximately 1–3 runs per delivery), which reflects the dominance of low-run outcomes in ball-by-ball data. The diagonal reference line highlights a consistent underestimation of higher run values, suggesting that the model struggles to capture boundary events and the upper tail of the scoring distribution due to data imbalance and limited contextual features.

2) Residuals Distribution: Figure 5 illustrates a a slightly multi-peaked distribution of residuals, indicating that the model exhibits different error behaviors across subsets of the data. A substantial concentration of residuals around zero suggests that the model achieves reasonable accuracy for a large portion of observations. However, the distribution is positively skewed, implying that the model more frequently underpredicts match totals. The presence of multiple peaks further indicates that the error structure is not uniform, suggesting that a single regression function may be insufficient to capture the heterogeneous scoring patterns inherent in The Hundred's fast-paced format.

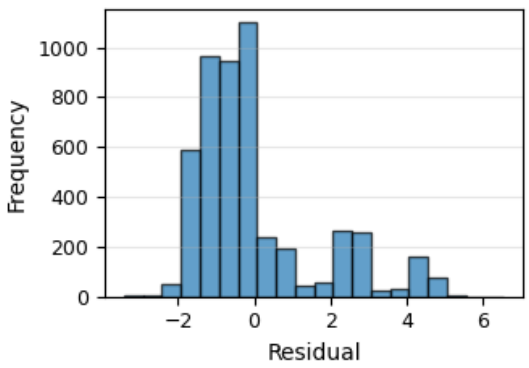

Fig. 5. Residual distribution (Actual – Predicted) showing model error behaviour.

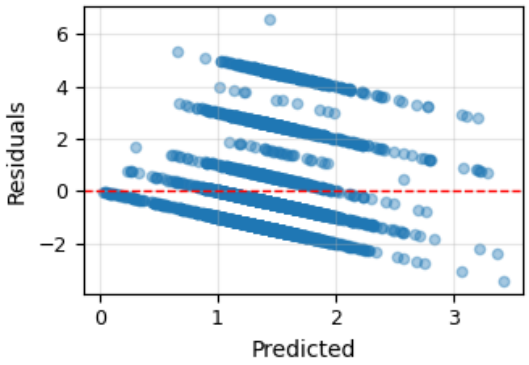

Fig. 6. Residuals vs Predicted values showing error trends across predicted run levels.

3) Residuals vs Predicted Values: Figure 6 reveals a clear downward-sloping pattern, where residuals become increasingly negative as predicted values increase. This indicates a systematic bias: the model tends to under-predict higher-scoring matches. Such a non-random residual pattern demonstrates that the model does not fully capture the complex scoring dynamics of The Hundred, and that additional features such as player form, pitch conditions, and team combinations may be required to improve predictive accuracy.

Overall, these error analyses highlight that while the Gradient Boosting Regressor performs adequately for common score ranges, its predictive efficiency decreases for higher totals and exhibits structural biases that warrant further feature engineering and model optimization.

D. Discussion on Weak Performance of Gradient Boosting Regressor

Although Gradient Boosting Regressor (GBR) is widely regarded as a powerful ensemble-based regression algorithm, its performance in predicting scoring patterns in The Hundred tournament was relatively weak, as reflected by a low R^2 value of 0.05. Since the model was trained and evaluated at the *delivery level*, this indicates limited explanatory power in capturing the variance of runs scored per ball.

One primary reason for this behavior lies in the highly volatile and context-dependent nature of scoring in The Hundred format. Unlike traditional formats such as ODI or T20, The Hundred encourages aggressive batting, frequent momentum shifts, and condensed innings, resulting in high variability in per-delivery outcomes that are difficult to model using static match-state features alone. As observed in the Predicted vs Actual analysis, the model consistently underestimates higher run outcomes, particularly boundary deliveries.

Another contributing factor is feature limitation. The regression model relied primarily on aggregated match-state variables such as overs completed, wickets fallen, and team combinations. While informative, these features do not fully capture critical contextual elements such as batter–bowler matchups, pitch behavior, boundary dimensions, or real-time acceleration in run scoring. Consequently, the model exhibits regression-to-mean behavior, producing predictions clustered within a narrow range.

Data imbalance further exacerbates this issue. Most deliveries result in low run values (0–2 runs), with boundary events occurring less frequently. Gradient Boosting, despite its strength, may therefore prioritize minimizing overall error, leading to underrepresentation of rare but high-impact scoring events. In summary, while the Gradient Boosting Regressor performs adequately for common delivery outcomes, its low R^2 highlights the intrinsic difficulty of modeling per-delivery scoring behavior in The Hundred.

E. K-Fold Cross-Validation and Overfitting Analysis

To evaluate the generalization capability of the proposed models and to address potential overfitting, 5-fold cross-validation was performed on the training dataset.

TABLE III
5-FOLD CROSS-VALIDATION PERFORMANCE OF WIN-PREDICTION MODELS

Model	Accuracy (± SD)	Precision	Recall	F1-Score
Logistic Regression	61.40% ± 0.90%	0.5967	0.5517	0.5733
Decision Tree (Pruned)	94.88% ± 1.19%	0.9364	0.9588	0.9465
Random Forest (Pruned)	99.76% ± 0.05%	0.9985	0.9965	0.9975

Cross-validation was performed exclusively on the training set to prevent data leakage. As shown in Table III, Logistic Regression demonstrates limited performance, while pruned tree-based models exhibit significantly improved generalization.

The pruned Decision Tree classifier demonstrated substantially improved performance with an average accuracy of 94.88% (± 1.19%), maintaining balanced precision and recall.

The pruned Random Forest classifier achieved the highest and most stable performance, with an average accuracy of 99.76% (± 0.05%), indicating consistent predictive behavior across validation folds. To mitigate overfitting, pruning constraints such as maximum tree depth and minimum samples per leaf were applied to all tree-based models, reducing variance and improving robustness.

Although tree-based models exhibit near-perfect accuracy, these results should be interpreted as strong pattern learning within the available dataset rather than guaranteed real-world deployability. Further validation on unseen seasons or external leagues is required to confirm true generalization.

F. Prediction Results

TABLE IV
SCORE PREDICTION TEST RESULTS

City	Batting	Bowling	Score	Overs Completed	Wickets Fallen	Predicted Score
Cardiff	B. Phoenix	M. Originals	135	10	4	100
Cardiff	M. Originals	B. Phoenix	135	10	4	100
Cardiff	M. Originals	B. Phoenix	135	14	4	200
Nottingham	O. Invincibles	B. Phoenix	170	12	0	300
Nottingham	O. Invincibles	B. Phoenix	170	12	1-3	200
Nottingham	O. Invincibles	B. Phoenix	170	12	≥4	100
London	O. Invincibles	B. Phoenix	130	18	8	200
London	B. Phoenix	O. Invincibles	130	18	8	100

TABLE V
WIN PREDICTION TEST RESULTS (LOGISTIC REGRESSION)

City	Batting	Bowling	Target	Score	Ov. Done	Wk Fallen	Probability of Winning	
							Batting	Bowling
Cardiff	Manchester Originals	Oval Invincibles	189	170	12	4	46%	54%
Cardiff	Manchester Originals	Oval Invincibles	189	149	12	4	48%	52%
Cardiff	Manchester Originals	Oval Invincibles	189	149	15	6	50%	50%
Leeds	Oval Invincibles	Southern Brave	189	149	15	8	64%	36%
Leeds	Southern Brave	Oval Invincibles	189	149	15	8	50%	50%
Nottingham	Southern Brave	Oval Invincibles	189	149	15	8	52%	48%

TABLE VI
WIN PREDICTION TEST RESULTS (DECISION TREE CLASSIFIER)

City	Batting	Bowling	Target	Score	Ov. Done	Wk Fallen	Probability of Winning	
							Batting	Bowling
Cardiff	Manchester Originals	Oval Invincibles	189	170	12	4	0%	100%
Cardiff	Manchester Originals	Oval Invincibles	189	149	12	4	0%	100%
Cardiff	Manchester Originals	Oval Invincibles	189	149	15	6	100%	0%
Leeds	Oval Invincibles	Southern Brave	189	149	15	8	100%	0%
Leeds	Southern Brave	Oval Invincibles	189	149	15	8	100%	0%
Nottingham	Southern Brave	Oval Invincibles	189	149	15	8	100%	0%

TABLE VII
WIN PREDICTION TEST RESULTS (RANDOM FOREST CLASSIFIER)

City	Batting	Bowling	Target	Score	Ov. Done	Wk Fallen	Probability of Winning	
							Batting	Bowling
Cardiff	Manchester Originals	Oval Invincibles	189	170	12	4	41%	59%
Cardiff	Manchester Originals	Oval Invincibles	189	149	12	4	58%	42%
Cardiff	Manchester Originals	Oval Invincibles	189	149	15	6	64%	34%
Leeds	Oval Invincibles	Southern Brave	189	149	15	8	75%	25%
Leeds	Southern Brave	Oval Invincibles	189	149	15	8	62%	38%
Nottingham	Southern Brave	Oval Invincibles	189	149	15	8	70%	30%

V. Findings

The experimental evaluation of machine learning models yielded several insights. For match outcome prediction, Logistic Regression achieved a modest accuracy of 63%, indicating limited effectiveness in separating winning and losing classes in a highly volatile, short-format cricket environment. In contrast, tree-based models demonstrated substantially stronger performance. Confusion matrix analysis revealed very few misclassifications, highlighting the ability of tree-based models to capture non-linear interactions between match-state variables such as runs remaining, balls left, wickets in hand, and run-rate pressure. Despite their strong predictive performance, the exceptionally high accuracy of these models suggests a risk of overfitting, particularly given the relatively small dataset and the structured nature of second-innings chase features. This concern is partially mitigated through the use of k-fold cross-validation and tree pruning techniques; however, the results should still be interpreted as evidence of strong pattern learning within the available data rather than guaranteed real-world generalization.

In contrast, score prediction using the Gradient Boosting Regressor proved significantly more challenging. Although the model achieved low absolute error values (MAE $= 1.22$, RMSE $= 1.62$), its explanatory power was extremely limited, as reflected by a low coefficient of determination ($R^2 = 0.05$). Diagnostic visualizations revealed that predictions were tightly clustered within a narrow range and consistently underestimated higher match totals. The residual distribution exhibited skewness and multi-modality, while residuals-versus-predicted plots showed a clear systematic bias, confirming that the model failed to capture the full variance of scoring behavior in *The Hundred*.

These findings indicate that while match winner prediction is highly tractable using structured match-state features and tree-based classifiers, score prediction remains an inherently complex task in the 100-ball format. The aggressive scoring patterns, rapid momentum shifts, and context-dependent decision-making characteristic of The Hundred introduce variance that cannot be adequately modeled using aggregate features alone.

VI. Conclusion

This study evaluated the effectiveness of machine learning techniques for predicting outcomes of The Hundred cricket tournament using ball-by-ball and match-level data from multiple seasons. The results demonstrate that machine learning models, particularly tree-based classifiers, are highly effective for match winner prediction when informed by structured match-state features such as remaining runs, balls, wickets, and run-rate dynamics. These models consistently outperformed linear approaches, highlighting their ability to capture non-linear relationships inherent in short-format cricket. In contrast, score prediction proved significantly more challenging. Despite employing Gradient Boosting Regression, the model exhibited limited explanatory power and systematic underestimation of higher totals. This reflects the fast-paced, volatile, and context-dependent nature of The Hundred format, where scoring behavior is influenced by factors not fully captured by aggregate features.

Overall, the study confirms the strong potential of machine learning for outcome classification in The Hundred while underscoring the need for richer contextual features and advanced temporal modeling to improve score forecasting.

VII. Future Enhancements

Although the developed system provides valuable insights and demonstrates the feasibility of predictive analytics in The Hundred, several avenues remain for improvement:

- **Incorporation of Advanced Features:** Future models should integrate context-rich variables such as pitch conditions, weather information, player fatigue, real-time momentum indicators, and matchup-specific statistics to enhance score prediction accuracy.
- **Real-Time Prediction Capability:** Enhancing the system to update predictions dynamically during live matches based on ongoing ball-by-ball events would increase its value for analysts, broadcasters, and fantasy sports users.
- **Expanding the Dataset:** Incorporating more seasons, women's matches, and similar domestic leagues would enrich the model and improve reliability.

References

[1] K. Suresh, B. Vikas, Kanishka, and K. Vikas, "Design and analysis of a chatbot with IPL first inning score prediction," in *Proc. Int. Conf. Advancements in Electrical, Electronics, Communication, Computing and Automation (ICAECA)*, Coimbatore, India, 2021, pp. 1–4, doi: 10.1109/ICAECA52838.2021.9675645.

[2] T. Bhalerao, S. Vijayalakshmi, and G. J., "A comparative analysis on machine learning algorithms for score prediction and proposal of enhanced Naïve Bayes," in *Proc. 4th Int. Conf. Advances in Computing, Communication Control and Networking (ICAC3N)*, Greater Noida, India, 2022, pp. 618–621, doi: 10.1109/ICAC3N56670.2022.10074063.

[3] S. G., A. Swaminathan, J. B. J., S. R., and L. Nelson, "IPL data analysis and visualization for team selection and profit strategy," in *Proc. 7th Int. Conf. Computing Methodologies and Communication (ICCMC)*, Erode, India, 2023, pp. 592–598, doi: 10.1109/ICCMC56507.2023.10083736.

[4] N. Rodrigues, N. Sequeira, S. Rodrigues, and V. Shrivastava, "Cricket squad analysis using multiple gradient boosting regression," in *Proc. 1st Int. Conf. Advances in Information Technology (ICAIT)*, Chikmagalur, India, 2019, pp. 104–108, doi: 10.1109/ICAIT47043.2019.8987367.

[5] A. I. Anik, S. Yeaser, A. G. M. I. Hossain, and A. Chakrabarty, "Player performance prediction in ODI cricket using machine learning algorithms," in *Proc. 4th Int. Conf. Electrical Engineering and Information & Communication Technology (iCEEiCT)*, Dhaka, Bangladesh, 2018, pp. 500–505, doi: 10.1109/CEEICT.2018.8628118.

[6] O. Sadekar, S. Chowdhary, M. S. Santhanam, and F. Battiston, "Individual and team performance in cricket," *Royal Society Open Science*, vol. 11, no. 7, 2024, doi: 10.1098/rsos.240809.

[7] S. K. C., A. Khetan, B. Kumar, D. Tolani, and H. Patel, "Prediction of IPL match outcome using machine learning techniques," arXiv:2110.01395, 2021. [Online]. Available: https://arxiv.org/abs/2110.01395

[8] K. Passi and N. Pandey, "Increased prediction accuracy in the game of cricket using machine learning," arXiv:1804.04226, 2018. [Online]. Available: https://arxiv.org/abs/1804.04226

[9] S. K. Sachin, H. V. Prithvi, and C. Nandini, "Data science approach to predict the winning fantasy cricket team," arXiv:2209.06999, 2022. [Online]. Available: https://arxiv.org/abs/2209.06999