A Mini Project with Seminar On

CRICKET SCORE PREDICTION

Submitted in partial fulfillment of the requirements for the award of the

Bachelor of Technologyy

in

Department of Computer Science and Engineering (Data Science)

by

Mr. AJITH VARMA	**20241A6726**
Mr. ABHINAV SAI RATAN	**20241A6701**
Mr. VALAPARLA SATHVIK	**20241A6754**
Mr. MOHAMMED SHOIAB	**20241A6736**

Under the Esteemed guidance of

Dr Sanjeev Polepaka

Professor

Department of Computer Science and Engineering (Data Science)

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Approved by AICTE, Autonomous under JNTUH, Hyderabad)

Bachupally, Kukatpally, Hyderabad-500090

GOKARAJU RANGARAJU INSTITUTE OF ENGINEERING AND TECHNOLOGY
(Autonomous)

Hyderabad-500090

CERTIFICATE

This is to certify that the mini project entitled "**CRICKET SCORE PREDICTION"** is submitted **AJITH VARMA (20241A6726), ABHINAV SAI RATAN (20241A6701), VALAPARLA SATHVIK (20241A6754), MOHAMMED SHOIAB (20241A6736)** in partial fulfillment of the award of degree in BACHELOR OF TECHNOLOGY in Computer Science and Engineering (Data Science) during the Academic year 2023-2024.

Internal Guide

Dr Sanjeev Polepaka
Professor

Head of the Department

Dr. G. Karuna
Professor

External Examiner

ACKNOWLEDGEMENT

There are many people who helped us directly and indirectly to complete our project successfully. We would like to take this opportunity to thank one and all. First, we would like to express our deep gratitude towards our internal guide **Dr Sanjeev Polepaka,** Department of Computer Science and Engineering (Data Science), for his support in the completion of our dissertation. We wish to express our sincere thanks to **Dr. G. Karuna,** Head of the Department**,** and to our principal **Dr. J. PRAVEEN,** for providing the facilities to complete the dissertation. We would like to thank all our faculty and friends for their help and constructive criticism during the project period. Finally, we are very much indebted to our parents for their moral support and encouragement to achieve goals.

Mr. AJITH VARMA **(20241A6726)**
Mr. ABHINAV SAI RATAN **(20241A6701)**
Mr. VALAPARLA SATHVIK **(20241A6754)**
Mr. MOHAMMED SHOIAB **(20241A6733)**

DECLARATION

We hereby declare that the mini project titled **"Cricket Score Prediction"** is the work done during the period from **17th January 2023 to 12th June 2023** and is submitted in the partial fulfillment of the requirements for the award of degree of Bachelor of Technology in Computer Science and Business System from Gokaraju Rangaraju Institute of Engineering and Technology (Autonomous under Jawaharlal Nehru Technology University, Hyderabad). The results embodied in this project have not been submitted to any other University or Institution for the award of any degree or diploma.

Mr. AJITH VARMA	**(20241A6726)**
Mr. ABHINAV SAI RATAN	**(20241A6701)**
Mr.VALAPARLA SATHVIK	**(20241A6754)**
Mr. MOHAMMED SHOIAB	**(20241A6736)**

ABSTRACT

Cricket is a popular sport that involves a high degree of variability in terms of game conditions and player performance. The ability to accurately predict cricket scores could provide valuable insights for coaches, analysts, and fans, as well as offer opportunities for sports betting and fantasy games. This project explores the use of machine learning techniques to predict cricket scores based on a variety of contextual and historical factors. The publicly available cricket dataset is used to build and evaluate several regression models that predict the total runs scored by a team in a limited-overs cricket match. This analysis includes feature engineering to extract and transform relevant input variables, model selection to compare and choose among different regression algorithms, and performance evaluation to assess the accuracy and robustness of the models. This project also conducts sensitivity analysis to identify the most influential predictors and explore the potential biases and limitations of the models. The results indicate that machine learning techniques can effectively predict cricket scores and provide valuable insights into the factors that contribute to team performance. The findings have implications for cricket teams, coaches, and analysts who seek to improve their game strategies and player selection, as well as for sports betting and fantasy game platforms that seek to provide more accurate and engaging experiences for users.

TABLE OF FIGURES

Figure No.	Figure Name	Page No.
1.1	Linear Regression	3
1.2	Logistic Regression	5
1.3	Lasso Regression	6
1.4	Ridge Regression	8
1.5	Decision Trees	10
1.6	Random Forests	12
1.7	Architecture Diagram 1	14
3.1	Architecture Diagram 2	21
3.2	Class Diagram	27
3.3	Sequence Diagram	28
3.4	Use Case Diagram	29
3.5	Activity Diagram	30
4.1	Experimental Output 1 of a Match	32
4.2	Experimental Output 2 of a Match	33
5.1	Result Graph 1	39
5.2	Output for Graph	40

TABLE OF CONTENTS

Chapter No.	Chapter Name	Page No.
	Certificate	ii
	Acknowledgment	iii
	Declaration	iv
	Abstract	v
	List of Figures	vi
	List of Tables	vii
1	**Introduction**	1
	1.1 Introduction to the project work	1
	Objectives of the project	1
	1.2 Methodology adopted to satisfy the objective	13
	1.3 Architecture Diagram with Brief Description	14
2	**Literature Survey**	15
	2.1 Literature Survey	15
	2.2 Drawbacks of Existing Approaches	19
3	**Proposed Method**	21
	3.1 Problem Statement and Objective of the Project	21
	3.2 Explanation of :	21
	Architecture Diagram	21
	Modules Connectivity Diagram	21

	Software and Hardware Requirements	24
	3.3 Modules and their Description	24
	3.4 Requirement Engineering	
	Functional	
	Non Functional	
	3.5 Analysis and Design through UML Diagram:	28
	Class Diagram	28
	Sequence Diagram	29
	Use case Diagram	30
	Activity Diagram	31
4	**Results and Discussions**	32
	4.1 Description of Dataset	32
	4.2 Detailed Explanation of Experimental Results	32
	4.3 Significance of the proposed methods with its Advantages	35
5	**Conclusion and Future Enhancements**	37
	Summary	37
	Objective	39
	Importance and Significance	38
	Approach Adopted	38
	Results	41
	Conclusion	43
	Future Enhancements	44
6	**Appendices**	45

6.1 Source Code 45

7 References 49

CHAPTER 1

1.1 INTRODUCTION

Cricket is a popular sport that involves a high degree of variability in terms of game conditions and player performance. The ability to accurately predict cricket scores could provide valuable insights for coaches, analysts, and fans, as well as offer opportunities for sports betting and fantasy games. This project explores the use of machine learning techniques to predict cricket scores based on a variety of contextual and historical factors. The publicly available cricket dataset is used to build and evaluate several regression models that predict the total runs scored by a team in a limited-overs cricket match. This analysis includes feature engineering to extract and transform relevant input variables, model selection to compare and choose among different regression algorithms, and performance evaluation to assess the accuracy and robustness of the models.

This project also conducts sensitivity analysis to identify the most influential predictors and explore the potential biases and limitations of the models. The results indicate that machine learning techniques can effectively predict cricket scores and provide valuable insights into the factors that contribute to team performance. The findings have implications for cricket teams, coaches, and analysts who seek to improve their game strategies and player selection, as well as for sports betting and fantasy game platforms that seek to provide more accurate and engaging experiences for users.

The game includes predetermined rules and a score structure. The match's outcome is greatly influenced by the match site and the performance of each player. It is challenging to produce an accurate prediction of the match because of how interdependent these many elements are on one another. In this project, we'll create a prediction system that analyses data from previously played matches and foretells details of upcoming matches, like the outcome of the game's final score and whether it will be a win or a defeat.

By examining previously stored match data using a variety of machine learning methods, our system will be able to anticipate match outcomes. In terms of the match venue, we plan to employ more features like pitch quality, weather condition, toss result, and individual player performance. Finally, our system displays quantitative findings using the most accurate and best-suited methodology. Moreover, showcasing how well our algorithms

estimate the number of runs scored, one of the key factors in a match's outcome. Linear regression, logistic regression, lasso regression, ridge regression, decision trees, random forests, and many other methods fall within the category of machine learning. The background and justification for each machine learning algorithm are provided below.

Linear Regression

With the purpose of establishing a linear relationship between a dependent variable and one or more independent variables, linear regression is a well-liked and frequently applied statistical modelling technique. It is a supervised learning algorithm belongs to the moreithm that inclusive regression analysis subset. After the model has been fitted, predictions can be made using fresh, unforeseen data by entering the values of the independent variables. Based on the calculated coefficients and the input values, the model determines the predicted value of the dependent variable.

Relationships between many variables can be studied using linear regression. It may deal with several independent variables, enabling the investigation of more intricate interactions and taking into account various variables that have an impact on the dependent variable.

The benefits of linear regression are numerous. It is straightforward and computationally effective, making it simple to use and comprehend. It enables hypothesis testing and variable selection by revealing the strength and direction of the correlations between variables.

The benefits of linear regression are numerous. It is straightforward and computationally effective, making it simple to use and comprehend. It enables hypothesis testing and variable selection by revealing the nature, intensity, and direction of the correlations between variables. Additionally, it serves as the basis for numerous additional sophisticated regression and machine learning techniques.

However, for linear regression to produce correct results, several conditions must be satisfied. These presumptions include homoscedasticity (the variance of errors should be constant across all levels of the independent variables), independence of errors (the errors should be uncorrelated), normality of errors (the errors should follow a normal distribution), and linearity (the relationship between variables should be linear). To create a linear relationship between a dependent variable and one or more independent variables, linear regression is a potent statistical modelling tool. It offers a simple method for predicting and comprehending how independent factors will affect the dependent variable, making it a crucial tool in data analysis, economics, social sciences, and many other disciplines.

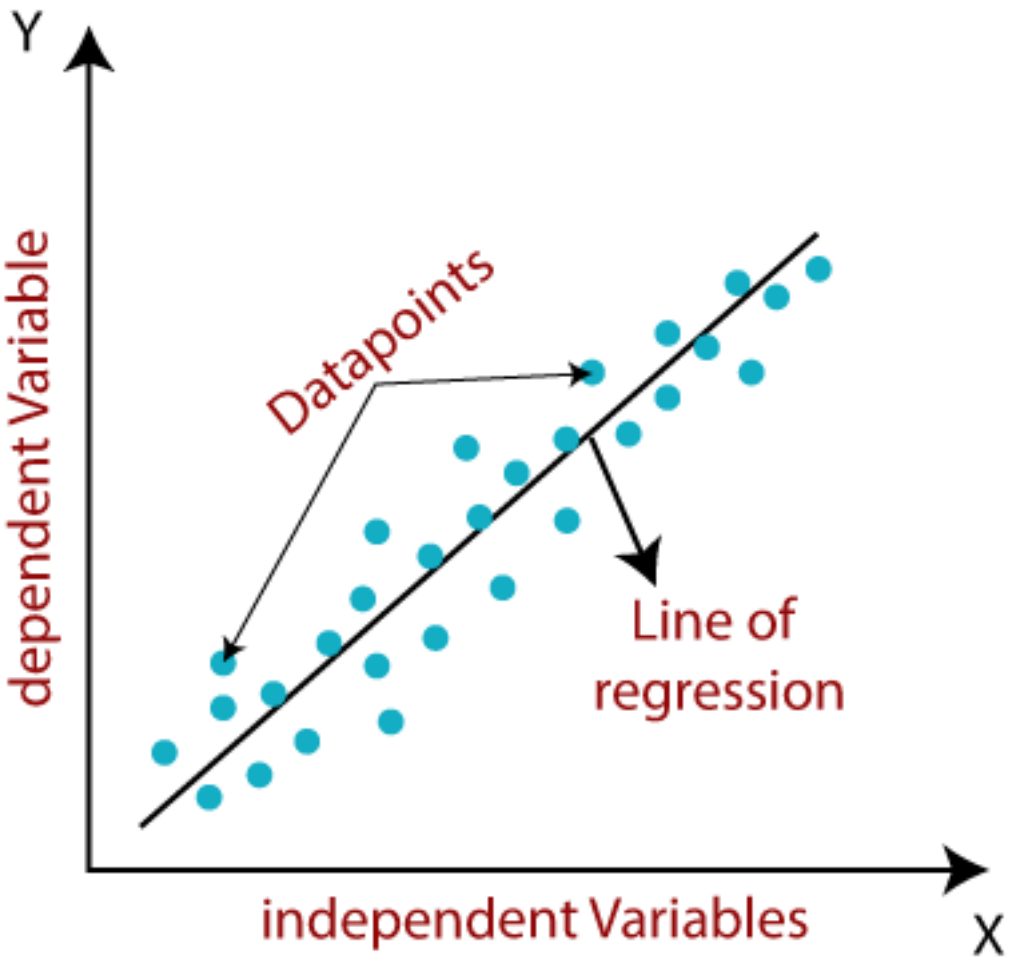


Fig 1.1 (Linear Regression)

Logistic Regression

Logistic regression is intended to forecast the likelihood that an event will occur or not, as opposed to linear regression, which tries to predict continuous numeric values. When the dependent variable is binary or dichotomous—that is, when it can only have one of two potential values—such as "yes" or "no," "success" or "failure," or 0 and 1—it is especially helpful. A common statistical modelling method for forecasting probability or binary outcomes is logistic regression. Even though its main objective is classification rather than regression, it is a supervised learning approach that is included in the regression analysis category.

The logistic regression model uses maximum likelihood estimation or other optimisation methods to estimate the coefficients. The intensity and direction of the association between the independent variables and the likelihood that the event will occur are indicated by these

coefficients. In contrast to a negative coefficient, which denotes a negative correlation, a positive coefficient indicates a positive association.

A threshold is often specified in order to divide the probabilities into the two groups before applying logistic regression to create predictions. The event is anticipated to happen if the predicted probability is greater than the threshold; else, it is predicted not to happen.

Numerous industries, including healthcare, finance, marketing, and social sciences, use logistic regression. Predicting outcomes like the existence or absence of an illness, customer turnover, credit default, and more is a typical use for it.

The independence of observations, the linearity of the connection between the independent variables and the outcome's log-odds, the lack of multicollinearity, and an acceptable sample size are only a few of the assumptions made by logistic regression.

For binary classification and probability prediction, logistic regression is a frequently used statistical modelling technique. It offers a method for modelling the connection between independent factors and the probability of an event happening. Logistic regression is a useful tool in many areas of data analysis and predictive modelling because it provides predictions and insights into binary outcomes by estimating coefficients and applying the logistic function.

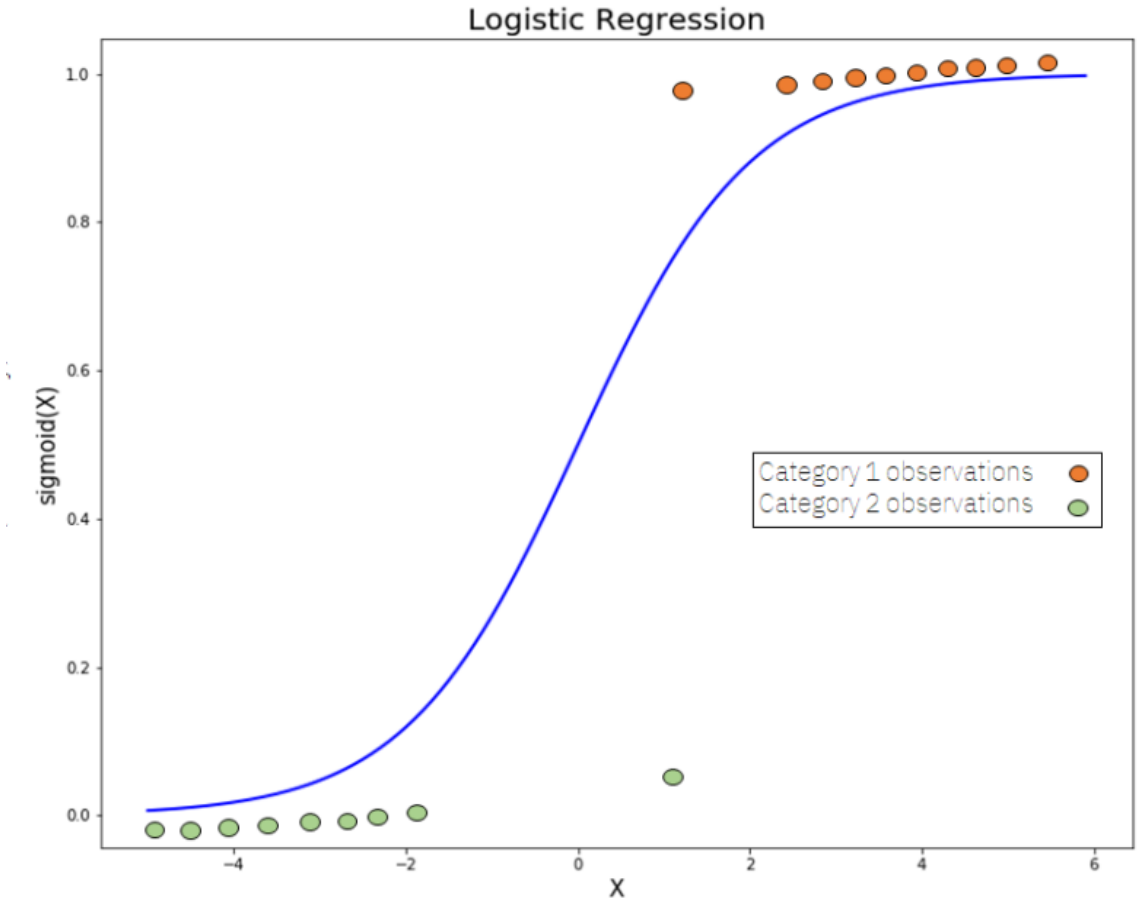


Fig 1.2 (Logistic Rgression)

Lasso Regression

The objective of conventional linear regression is to reduce the squared sum of errors between the predicted and actual values of the dependent variable. But in lasso regression, an extra penalty component is introduced to the objective function, which is the result of adding the absolute values of the regression coefficients and a regularisation parameter, lambda.

By reducing part of the regression coefficients to zero, lasso regression has the main benefit of performing automatic feature selection. By raising the value of, less significant variables suffer more penalties, leading to a sparser model that only retains the most pertinent predictors. Because of this characteristic, lasso regression can be used to handle datasets with a lot of characteristics or when there is a possibility that the predictors may be multicollinear.

Finding the set of regression coefficients that minimises the objective function is the first step in fitting a lasso regression model. Different optimisation strategies, such as coordinate descent or least angle regression (LARS), can be used to accomplish this.Lasso regression has a variety of real-world uses, including as dimensionality reduction, predictive

modelling, and feature selection. It's crucial to remember that lasso regression has comparable assumptions to those of linear regression, including linearity, observational independence, and error normality. Additionally, lasso has a tendency to arbitrarily choose one predictor from correlated predictors, which could bring bias into the model.

As a regularisation method, lasso regression extends linear regression by including a penalty term based on the total absolute value of the regression coefficients. It is useful for managing high-dimensional data and determining significant predictors since it offers a method for feature selection and model complexity control. In many areas of data analysis and predictive modelling, lasso regression offers workable solutions by finding a compromise between prediction accuracy and model simplicity.

There are many uses for lasso regression in a variety of fields, including finance, healthcare, social sciences, and more. It offers a versatile and understandable method for choosing features and making predictions, allowing for insights into the most crucial predictors and their effects on the outcome variable.

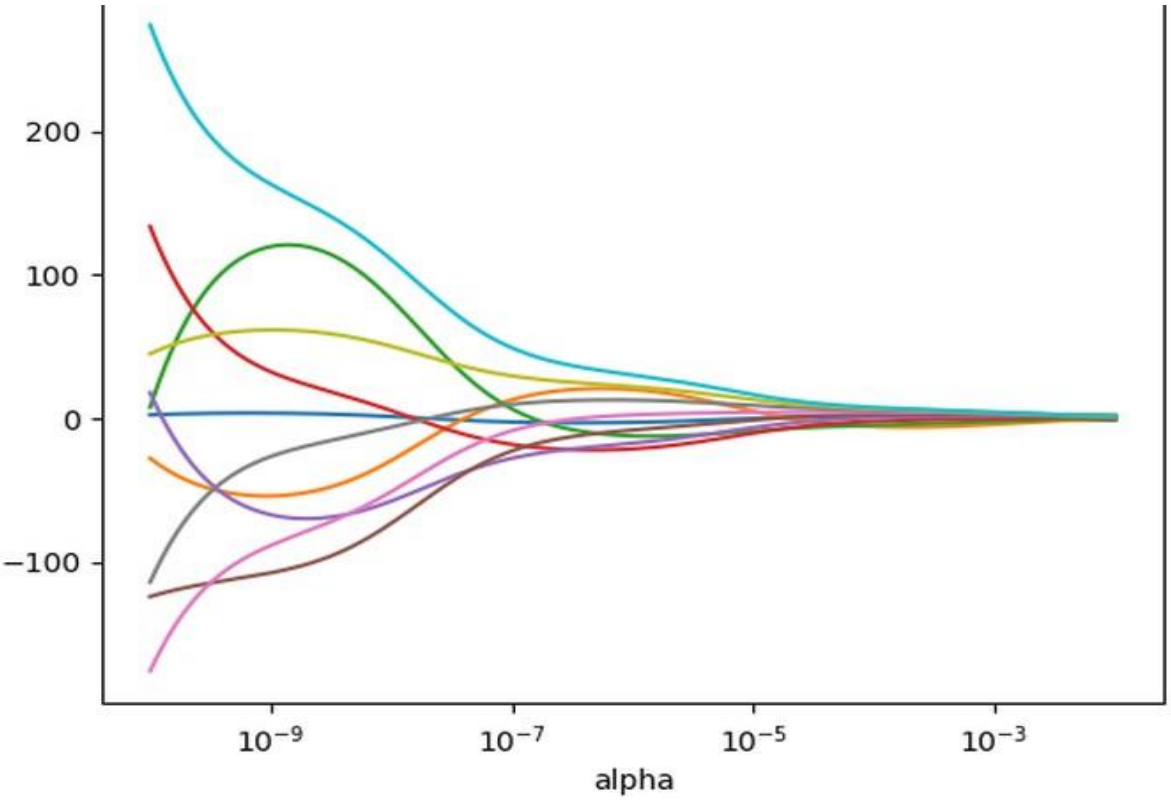


Fig 1.3 (Lasso Regression)

Ridge Regression

By adding a regularisation term to account for multicollinearity and complex control models, ridge regression is a regression approach that improves on ordinary least squares regression. It frequently applies when the dataset contains correlated predictors and works by introducing a penalty term into the regression goal function to guard against overfitting.

Minimising the sum of squared errors between the predicted values and the actual values of the dependent variable is the objective of conventional linear regression. The objective function is added a second penalty component in ridge regression, however, that is proportional to the square of the regression coefficients . In several disciplines, including finance, economics, and engineering, ridge regression is frequently employed. It aids in controlling multicollinearity, enhancing the stability of coefficient estimations, and making forecasts that are more reliable .

It is crucial to remember that ridge regression has comparable assumptions to those of linear regression, including linearity, observational independence, and error normality.

In conclusion, ridge regression is an effective method for addressing multicollinearity and managing model complexity by including a regularisation component in the objective function. It is effective for predictive modelling and lessens overfitting since it strikes a balance between model flexibility and stability. When there are associated predictors, ridge regression is important because it makes predictions and insights more precise.

Different optimisation methods, such as gradient descent or closed-form solutions, can be used to achieve ridge regression. Through methods like cross-validation, where several values of are examined to find the best balance, the regularisation parameter can be identified.

Ridge regression has a number of benefits, including

Ridge regression effectively handles multicollinearity by minimising the influence of linked predictors, resulting in more stable and trustworthy coefficient estimations.

Improved model performance: Ridge regression can result in greater generalisation and prediction performance on unobserved data by minimising the variance and overfitting.

Feature shrinkage rather than elimination: In contrast to feature selection techniques, ridge regression simply causes predictors' coefficients to approach zero, which permits the inclusion of all predictors in the model.

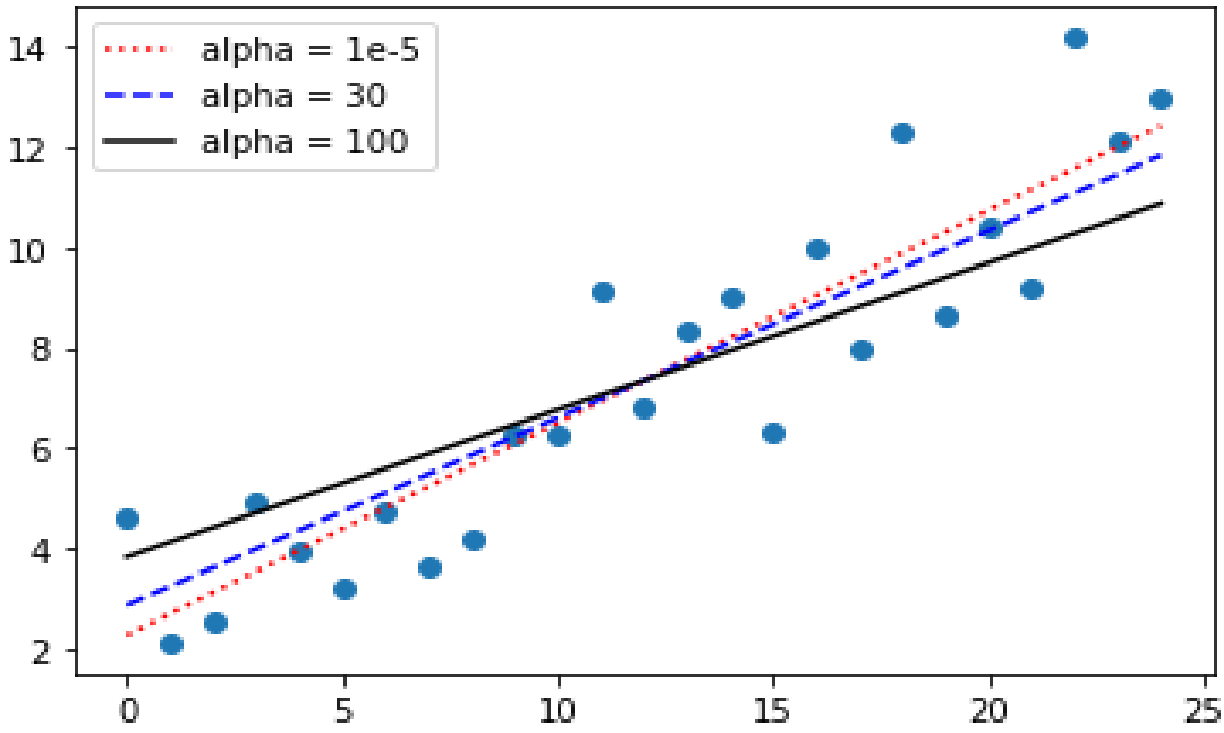


Fig 1.4 (Ridge Regression)

Decision Trees

For both classification and regression issues, a supervised machine learning strategy called a decision tree is used. Its form is like a flowchart, with each internal node representing a characteristic or attribute, each branch a set of rules, and each leaf node a conclusion or prediction.

The decision tree algorithm starts with the entire dataset and splits it into smaller groups iteratively based on the values of several attributes. The information benefit is maximised or the impurity measure (such as Gini impurity or entropy) is minimised at each stage of the partitioning procedure. A minimal number of samples are taken from each leaf if a stopping condition is met, such as when a maximum depth is reached. Applying the decision rules along the way, you start at the root node of a decision tree and work your way to the leaf node. The leaf node gives the input instance the anticipated class or value. Because the final model is clear and easy to understand, decision trees have a variety of advantages, one of which is their interpretability. It is possible to support both categorical and numerical features, and non-linear interactions between features are also taken into account. Decision trees also do implicit feature selection since important features often appear higher up the tree.

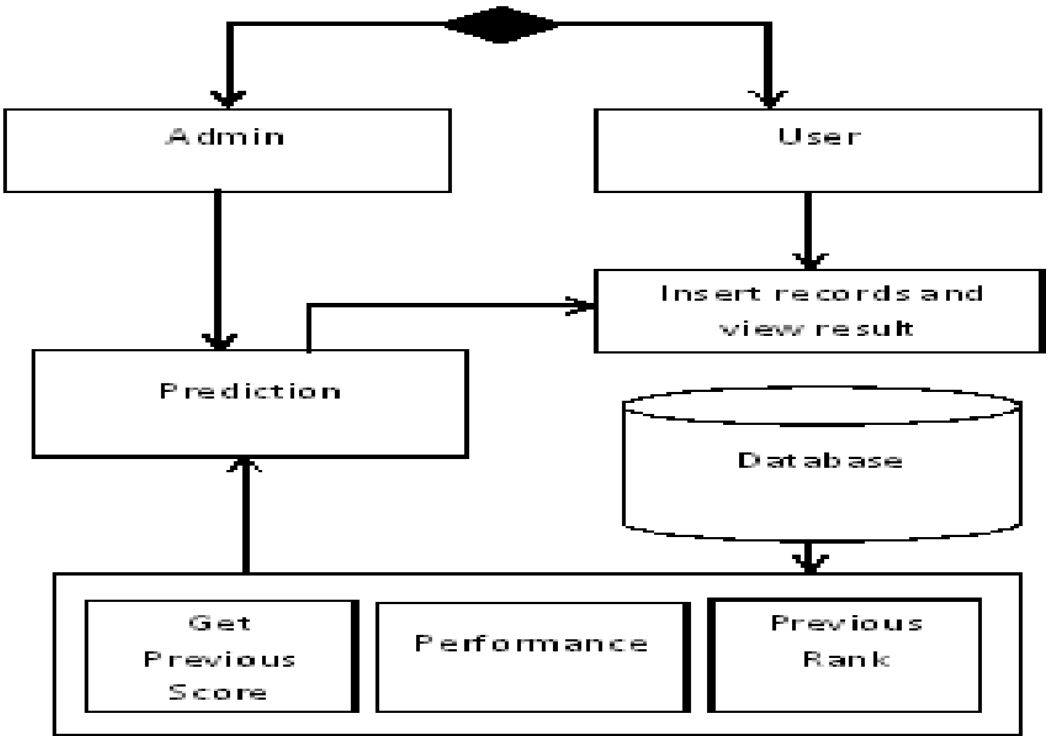
But when the trees are deep and complicated, overfitting can occur in decision trees. When a model gets overly specialised to its training set and struggles on new data, it has overfitted. Techniques like pruning, establishing a limit depth, or using ensemble methods like random forests or gradient boosting can be used to reduce overfitting. Due to its simplicity, interpretability, and efficacy across a variety of domains, decision trees are generally well-liked and frequently employed in machine learning.

Based on the characteristics of the input instance, you follow the path from the root node to a leaf node to make predictions using a decision tree. Here is a detailed procedure

1. Start at the decision tree's root node.
2. Based on the input instance, evaluate the feature or property at the current node.
3. Observe the branch that leads to the feature's value.
4. Depending on the selected branch, go to the following node (child node).
5. Up until you reach a leaf node, repeat steps 2-4.
6. The outcome or value indicated by the reached leaf node is the prediction.
7. It is significant to remember that the training data and selected splitting criteria affect how the decision tree is built. The quality and representativeness of the training data as well as the complexity of the tree all affect how accurate and reliable predictions made using decision trees are.

In the history of machine learning and data mining, decision trees have played a significant role. Decision trees have been around since the first decade of the 1960s.
Decision trees and their variations have been utilised extensively over time in a variety of industries, including banking, healthcare, marketing, and more. Their acceptance has been aided by their understandability, clarity, and efficiency in handling both categorical and numerical information. To improve decision tree algorithms' performance and handle issues like overfitting and handling big datasets, researchers are always investigating new approaches and enhancements.

Fig 1.5 (Decision Trees)



Random Forest

The strength of several decision trees is combined by the machine learning algorithm known as Random Forest to produce a reliable and precise predictive model. It fits into the ensemble learning area, where various models are merged to enhance overall performance.
The essential tenet of Random Forest is that the individual flaws of each decision tree can be reduced by mixing many trees, each of which has been trained on a distinct sample of the data. The term "Random Forest" refers to the process of randomly choosing the subsets of data and features that will be utilized to construct each decision tree.
Numerous industries, including finance, health care, and image identification, use Random Forest extensively. It is a preferred option for many machine learning applications due to its adaptability, capacity for handling big data-sets, and robustness.

To increase prediction accuracy and decrease over fitting, they described Random Forest as an ensemble method that brings together various decision trees. The study showed how well Random Forest performed classification and regression tasks.

In the machine learning field, the Random Forest method gained popularity and recognition in 2003. Researchers and practitioners were interested in it because of its capacity to manage high-dimensional data, reduce variation, and offer measurements of feature value. 2008 saw the implementation of the Random Forest algorithm in well-known machine learning packages including scikit-learn in Python and random Forest in R. It is now easier to use and more accessible to a wider audience thanks to these features.
With the publication of the study "Deep Forest- Towards an Alternative to Deep Neural Networks" by Pierre in 2012, Random Forest received more attention. For some applications, this research suggested employing Random Forest instead of deep neural networks. It examined Random Forest's potential for learning intricate representations and obtaining high accuracy.
Random Forest is still a popular and significant machine learning method today. It continues to be a popular option for many applications in academia and industry due to its versatility in handling different data formats. To solve certain issues and enhance performance, a number of Random Forest extensions and variations have been created over time. These include methods like Extremely Randomised Trees, which add more randomness to the tree-building process, and Parallel Random Forest, which efficiently handles big data-sets by parallelizing the training process.

Random Forest has evolved because of ongoing research and development, becoming a strong and popular algorithm for a variety of machine learning tasks. Making predictions with Random Forest entails leveraging the trained ensemble of decision trees to anticipate novel, unforeseen events. A potent machine learning method called Random Forest combines the predictions of various decision trees to produce predictions that are more reliable and accurate. An overview of the Random Forest prediction process is provided below

Phase One The Random Forest model must be trained on a labelled dataset before it can provide predictions. The Random Forest technique creates an ensemble of decision trees using the supplied training data during the training phase.

The Random Forest is an ensemble of decision trees, where each tree is constructed using a random subset of the training data. The input data is run through each decision tree in the Random Forest to forecast outcomes for new instances. The majority vote is used to determine the class prediction for classification problems. The class with the most votes, as determined by the votes of each decision tree in the ensemble, is chosen as the final forecast.

For regression tasks, the final prediction is calculated by averaging the projected values from each decision tree.

Random Forest offers a way to gauge the confidence or unreliability of predictions. The likelihood or percentage of trees that selected a particular class during a classification challenge can be used to calculate the confidence level.

Numerous industries, including banking, medicine, marketing, and more, have successfully used Random Forest. It is a well-known and effective technique in the field of machine learning because it can produce precise predictions by combining the results of various decision trees.

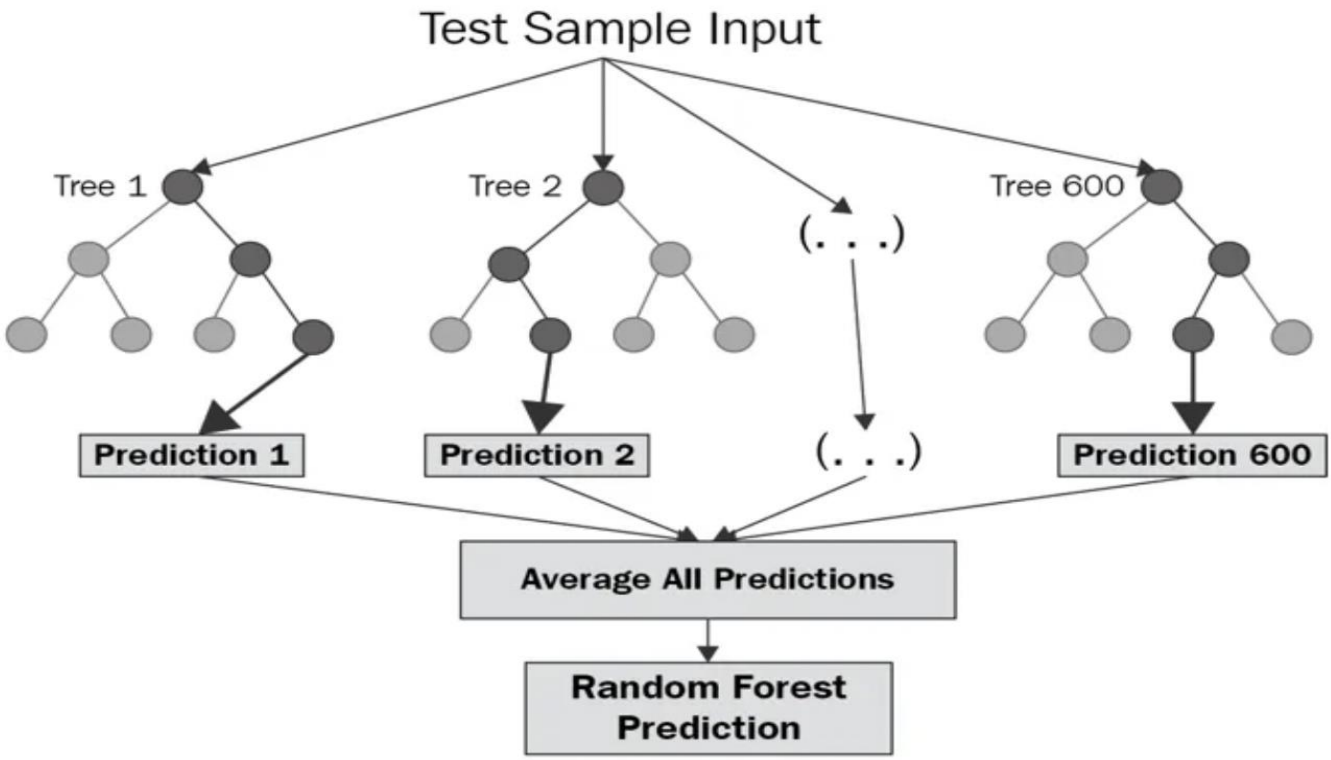


Fig 1.6 (Random Forest)

1.2 METHODOLOGY

1.2.1 System Architecture

The system will first be fed an input data set that includes information about players, scores, and venues for matches, among other things. The data will next undergo additional processing before being divided into training and testing data sets. Now, supervised and unsupervised learning are applied to the training data set. For the supervised learning data sets, some appropriate algorithms will be used, including the Lasso Regression, Naive Bayes, Logistic Regression, Support Vector Machine, and Random Forest Algorithms.

1.2.2 Input Pre-Processing

The data set will be loaded first, followed by the application of the analytical rules, as part of the input pre-processing procedure. The cleaning technique will be used to remove the outliers because the data is not pure and clean. The individual model that is used to gauge accuracy and aid in score prediction must then be trained. To make this happen, we must choose the most suitable model and then choose the model that best fits the facts.

1.2.3 Algorithm

An approach to regularization is lasso regression. For precise prediction, it is preferred to regression techniques. It makes use of shrinking. When data values shrink to the mean, this is referred to as shrinkage. The lasso method favours thin, uncomplicated, and easy models. Random Forest is a classifier that uses several decision trees on different subsets of the provided data set and averages the results to increase the data set's predicted accuracy. Using predictions from all of the decision trees rather than just one, the random forest forecasts the outcome based on most votes.

1.2.4 Logistic Regression

The classification method of logistic regression is one that machine learning has adapted from statistics. A data set with one or more independent variables that affect an outcome can be analyzed statistically using logistic regression.

1.3 ARCHITECTURE DIAGRAM

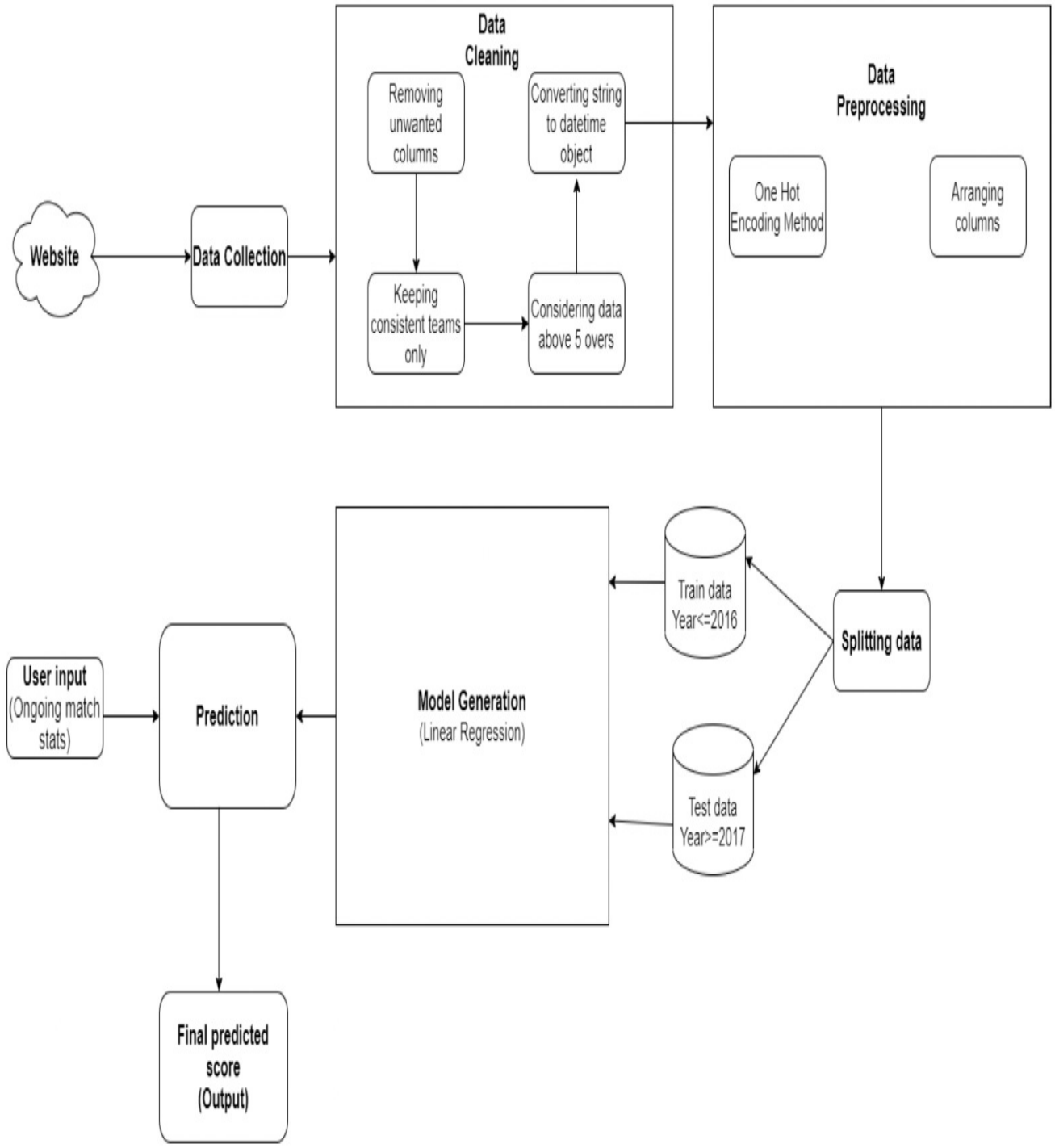


Fig 1.7 Architecture Diagram [1]

CHAPTER 2
LITERATURE SURVEY

1. Cricket Match Output Estimation Using ML

The twenty20 format of cricket is the most popular and well-liked by the public because it makes matches unpredictable until the final ball of the final over. The Indian Premier League (IPL), which began in 2008, is now the most well-known T20 league worldwide. We made the decision to create a machine learning model to forecast the results of its matches as a result. Cricket match outcomes are largely determined by several important variables, including home field advantage, previous performances on the field, records at the same location, the players' cumulative experience, records against specific opponents, and the team's and the individual player's overall present form. This paper briefly discusses the major variables that have an impact on the outcome of a cricket match, as well as the regression model that best fits the data and makes the most accurate forecasts.

2. Cricket match Score Estimation Using ML

The team's final score can currently be calculated using a technology that can determine the present run pace. The number of wickets and the venue where the game is being played are not taken into account. The existing system's shortcomings include its inability to forecast both the second team's score and the win %. This system, which is currently being developed, will have two models. The first model will forecast the score a team would receive after playing 50 overs in the given scenario. The second method uses player selection to forecast the win percentage for both teams even before the game has begun. We discovered that the regression error towards the mean classifier might be a smaller quantity in predicting match outcome than Naive mathematician, which was sixty-eight ab initio from 2-15 overs to ninety-one until the top of 42nd over.

3. Cricket Score Prediction using ML Algorithms

Cricket is a ground-based, 11-player team sport. In India, cricket is extremely popular. Due to the large number of spectators, many people attempt to anticipate the results of games using their own cricket intuition. The game includes predetermined rules and a score structure. The match's outcome is greatly influenced by the match site and the performance of each player. It is challenging to produce an accurate prediction of the match because of

how interdependent these many elements are on one another. In this project, we'll create a prediction system that analyses data from previously played matches and foretells details of upcoming matches, like the outcome of the game's final score and whether it will be a win or a defeat. Final quantitative findings are displayed by the most accurate and best-suited algorithm by our system. Additionally displaying how well our algorithms anticipate the number of runs scored, which is one of the most crucial factors in how a match will turn out.

4. Cricket Score and Winning team evaluation

Cricket is the most popular game, as we are all aware. The Indian Premier League (IPL) is one of the various cricket series that are played in our nation. 8 teams are now participating in it. The system that we suggest consists of a model with two components: the first portion predicts the score, and the second part predicts the team's likelihood of winning. While in winning prediction, the Lasso Regression method is used to predict the score. SVM, decision trees, and random forests are all employed as classifiers. The winning prediction is made by the model using supervised machine learning. In order to get the desired anticipated output, the Random Forest Classifier is employed for good accuracy and steady accuracy.

5. IPL Cricket Score and Innings Prediction Using Machine Learning

Cricket is the most popular game, after all. One of the various series contested in the nation is the Indian Premier League (IPL). 8 teams are now participating in it. The model that has two methods—the first being score prediction and the second being team victory prediction—has been put out in these articles. While winning prediction uses SVC classifier, decision tree classifier, and random forest classifier, score prediction in these uses linear regression, lasso regression, and ridge regression. The winner was predicted by the model using the supervised machine learning technique. To achieve the desired anticipated output with accuracy, Random Forest Classifier is utilized.

6. Sport analytics for cricket game outputs using machine learning

One of the more well-known cricket competitions in the world is the Indian Premier League (IPL), whose revenue rises each year as well as the number of viewers and the size of the IPL betting market. Cricket is a very dynamic game; thus, bookmakers and bettors are motivated to wager on the outcomes of matches because the game alters ball by ball. This study investigates using machine learning to solve the issue of predicting cricket match

outcomes using previous IPL match data. Filter-based techniques such as Correlation-based Feature Selection, Information Gain (IG), Relief, and Wrapper have been used to find the dataset's influential features. More importantly, machine learning techniques including Naïve Bayes, Random Forest, K-Nearest Neighbor (KNN) and Model Trees (classification via regression) have been adopted to generate predictive models from distinctive feature sets derived by the filter-based methods. A home team advantage subset and a toss decision subset were developed as two prominent subsets. On both feature sets, a predictive model was created using a few machine learning approaches. When compared to Kumesh Kapadia, Hussein Abdel-Jaber, Fadi Thabtah, and Wael Hadi, experimental testing reveals that tree-based models, in particular Random Forest, fared better in terms of accuracy, precision, and recall metrics. released in the journal Applied Computing and Informatics. Emerald Publishing Limited has published. Published under the Creative Commons Attribution (CC BY 4.0) license, this article. This material may be copied, distributed, translated, and used in other works if full credit is given to the original publication and authors and is not altered in any way.

7. The Cricket Winner Prediction with Application of Machine Learning and Data Analytics

Every organization is utilizing the most recent technology to expand their operations because of the evolution in the field of data sciences. There is rivalry in the market to provide better management, better evaluation quality, and better services. The analysis of data must be done with more accuracy and purity if all these requirements are to be met. With the use of existing data, machine learning is a new science that helps predict future outcomes so that better decisions can be made. Cricket is a well-known sport that is played and viewed in 104 different nations all over the world. Many of these cricket supporters want their team to play well and be declared the winner. Teams should focus on improving individual and collective performances if they want to win. Several variables, including team strengths, field conditions, weather, and batsman performances, affect how well one can predict the outcome of a cricket match. In this study, several features were examined to forecast the game's winner. This study focuses on picking the winner of an IPL match before it begins. By using the chosen features to train machine learning models, the IPL winner is predicted. Different machine learning techniques, including Random Forest, SVM, Naive Bayes, Logistic Regression, and Decision Tree, have been utilised for this model-building

purpose on test and training datasets of various sizes. The prediction model will be useful for cricketing boards in terms of cricket analysis and team strength evaluation. This model will be an unexpected blessing for gambling applications and match reporting media.

8. Cricket score prediction

In modern cricket, the CRR approach is used to predict the eventual score in the first innings. To calculate the final score, multiply the average number of runs scored in each over by the total number of overs. These kinds of algorithms are useless when taking into account T20 matches since in T20 cricket, the match's status can change incredibly quickly regardless of the present run rate. Within one or two overs, the game may be decided. it, to anticipate the score accurately, we should create a system that can do it more skillfully. Many people like both watching cricket and making predictions about the outcome. The goal of this study is to develop a method for accurately predicting cricket results for live IPL games while taking into account the prior dataset that is already accessible and several other key variables.

9. Cricket score Analysis and Prediction of actual Score and Winner using Machine Learning

This paper discusses a model that can predict both the final score and the winner of an IPL cricket match. The effectiveness of the model is affected by several factors, including the number of wickets taken in the previous five overs, the number of runs scored in the previous five overs, the number of overs, the overall score, and the number of wickets in the current ball. Data from IPL games played between 2008 and 2019 are included in the suggested model. This essay will outline the process for estimating the first inning's anticipated score while the game is still in progress. The score is predicted using the linear regression algorithm. This model accounts for around 75.226% of the data. The model focuses primarily on using the data from the previous five overs to forecast what might be the anticipated final score of the game, which has not been taken into account by any other models. Using this model, we can predict with reasonable accuracy how many runs the current batting team will score throughout a match.

10. Cricket score and winning estimation using data mining

A emerging area of computer science study with many challenges is data mining and machine learning in sports analytics. The aim of this project is to develop a system that can predict the outcome of a T20 cricket match, particularly an IPL match, while it is taking place. To determine the optimal output, various Machine Learning and statistical approaches were used. In order to compare the results, a widely utilised mathematical method called multiple linear regression is applied. Predictive modelling uses this model extensively. Currently, the first innings score in Twenty-Twenty (T20) cricket matches is forecasted using the current run rate, which may be computed as the number of runs scored per total number of overs bowled. It excludes things like the amount of wickets lost, the location of the game, and the toss. Additionally, there is no way to forecast the game's outcome in the second innings. In this study, a model that predicts the score in each of the innings using multiple variables linear regression, logistic regression, and finally the match winner using the Random Forest method is proposed.

1.1DRAWBACKS OF EXISTING APPROACHES

1. When the basic assumptions are starting to fail, the approaches that we've used cannot process the matches that have high scores.
2. Accuracy for prediction results is comparatively less.

1.2 SUMMARY

Ref No.	Description	Algorithm	Accuracy
[1]	Score and Winning Prediction in Cricket through Data Mining Oct 8-10, 2015	1.Linear Regression algorithm 2.Naive Bayes classifier	Accuracy of LR is 20% greater than CRR method.
[2]	Money Ball - Data Mining on Cricket Dataset 2019	1.Naive Bayes classifier 2.Support Vector Machine 3.K-Nearest Neighbor method 4.Random Forest method	Linear regression 80.76 Ridge regression 80.69 Lasso regression 81.00

[3]	Cricket Match Outcome Prediction Using Tweets and Prediction of the Man of the Match using Social Network Analysis: Case Study Using IPL Data 2018	1.Naive Bayes classifier 2.Support Vector Machine 3.K-Nearest Neighbor method 4.Random Forest method 5.Logistic Regression	1.Sentiment analysis has an accuracy of up to 85%. 2.Tweet based and Mixed model has 89% accuracy. 3.Natural attributes-based model has accuracy up to 83%.
[4]	Cricket Squad Analysis using multiple Random Forest Regression 2019	1. Linear Regression 2. SVR 3. Decision Tree 4. Random Forest	Accuracy can be increased by predicting matrices of the player against the player.
[5]	Player's Performance Prediction in ODI Cricket Using Machine Learning Algorithms 2018	1. Linear Regression 2. SVM with linear kernel 3.SVM with polynomial kernel Li near Regression	Accuracy (with 90 % training data): SVC: 43 Decision Tree: 61 Random Forest: 76

CHAPTER 3

PROPOSED METHOD

3.1 PROBLEM STATEMENT

In the profession of cricket score prediction, a variety of strategies are used to forecast the innings score of a cricket match. Numerous systems and prediction computations are used to forecast the outcomes of ODI and T20 cricket matches. When predicting the results of cricket matches, the CRR technique is frequently used. The total number of overs in an inning is multiplied by the number of runs scored in an over in the CRR technique. This method excludes all other factors and only considers runs scored in an over. By accounting for many variables, we are striving to improve the accuracy of the current systems and the predictions. Our objective is to forecast a live game's result.

3.2 ARCHITECTURE DIAGRAM

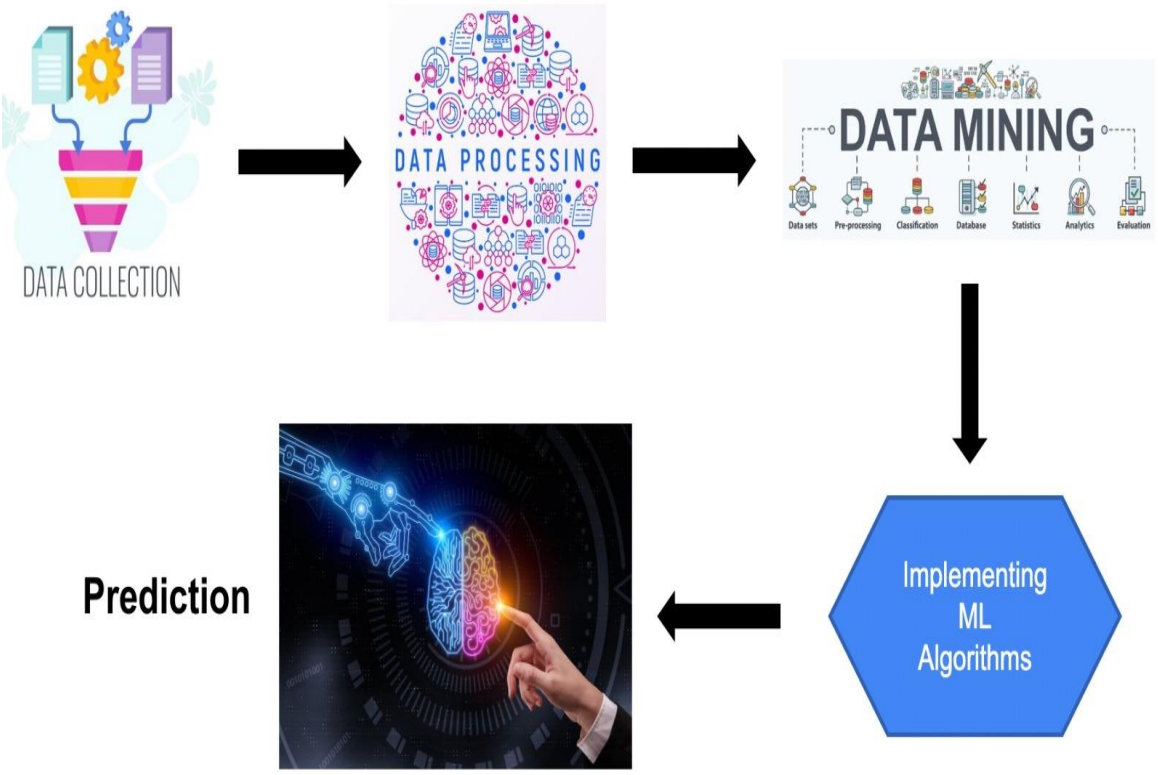


Fig 3.1 Architecture Diagram

The aim is to predict the match outcome and performance of each player based on the previous data. To achieve a reliable accuracy, we need to analyze a large amount of data. Therefore, the initial step of the implementation was to collect data for all possible matches. Data-set is collected from various websites such as ESPN, Kaggle, etc.

We collected data which provides ball by ball details for all the matches. Various analytical rules are used to filter collected data with respect to the selected feature. Features can be matching venue, playing 11, weather condition, performance of individual player. Further, cleaned data is split into training (80%) and testing data (20%). Training data is fed to all machine learning model and accuracy of each model is noted. Model having highest accuracy is selected for further prediction. This model is used to make perdition over the input data provided, input data may include – home team, opposition team, current weather condition and analysis done on historical data.

We want to create a model that is effective at predicting the score in a live IPL match. We want to create a reference that can take several factors into account when predicting scores.

(A) Data Collection - We will obtain the data-set from the Kaggle data-sets. We'll collect the data in CSV format. The following step will involve cleaning the data that was obtained from the website.

(B) Data Cleaning - As part of the data cleaning process, we wish to eliminate all extra columns, such as the match id, location, bowler and batsman names, as well as the striker and non-striker scores. These columns will be omitted as estimation won't need them. Some teams no longer engage in the league, in accordance with IPL data. The IPL does not feature the Deccan Chargers, Gujarat Lions, Pune Warriors India, or Rising Pune Supergiant. As a result, we need to exclude those teams from the data collection and only take into consideration those who are reliable teams. After five overs, we will evaluate the data. The date column, which is available in the data collection in string format, is the subject of a variety of manipulations that we would like to carry out.

(C) Data Preprocessing - After data has been cleaned, we will need to prepare our data. The process of data preparation will include the one-shot encoding. One hot encoding covers a lot of the actualization portion. We'll need to redesign the columns in our data collection during the preprocessing phase of the data. Adjusting the columns is essential since we need to appropriately arrange our columns in particular series. After gathering the data, we will separate it so that IPL matches played prior to 2016 will be utilized for the model's training and matches played after 2016 will be used as test data.

(D) Algorithms - For the forecast, we'll use the Random Forest Regression, the Linear Regression model, and the Lasso Regression model. The model with the greatest precision will be used to make the prognosis. The model that we are going to use for the prediction will be explained during action chapter.

(E) Prediction - The model will process the data before gathering user inputs. After gathering user inputs and comparing them to historical data, we will be able to estimate a score range, or from lower bound to upper bound. The model architecture of the CFP system is shown in the diagram above.

The data set was encoded in one step. Data must be organized logically for one-hot encoding to occur. We encode data into numbers because many machine learning algorithms cannot function on categorical input. We have columns for the batting and bowling teams in our data gathering. However, our model must be able to understand the user's input when we provide it. The batting and bowling columns both contain several squads. We rely on one-hot encoding because we don't want to provide the batting team and bowling team's input in string format. The model receives the encoded data frame as input. After the user submits the form, the model makes an estimation based on the data. Here we are using many algorithms for perfection like linear regression, decision tree, lasso regression, logistic regression etc.

3.2.1 SOFTWARE REQUIREMENTS

Programming Languages	- PYTHON
Modules	- Scikit Learn, Tensor Flow
Editor	- Jupyter Notebook

3.2.2 HARDWARE REQUIREMENTS

Processor	- Intel Core i5 / Ryzen 5 or better
RAM	- 8GB or more
Storage	- As per Requirements

3.3 MODULES AND DESCRIPTION

There are several modules that can be used for cricket score prediction using machine learning. Here are some popular ones

Statistical Models

Statistical models use historical data estimate the outcome of a sports event. These models can be used to analyse the output of the play, the scoreline, and the likelihood of different events occurring during the game. Some common statistical models used for sports score prediction include regression analysis, time series analysis, and machine learning algorithms. Data scientists can identify correlations between variables and produce predictions by applying statistical models to raw data to produce accessible visualisations. Common data sets for statistical analysis include census data, public health data, and data from social media.

By doing this, you will fully understand every idea from every subject. To find ideas from a set of data, we use statistical models. We can perform modelling on a relatively small sample of data to attempt and understand the core nature of the data. Each statistical model has built-in weaknesses or errors. They are used to approach reality. There are times when the model's underlying assumptions are simply excessively rigid and inaccurate. The traditional instance of using one or more variables to analyse how every explanation variable influences the independent variable is regression. The act of creating sample data and producing

predictions for the games using mathematical models and statistical hypotheses is known as statistical modelling.

Scikit-Learn

The well-known Python machine learning framework Scikit-Learn can be used to predict cricket results. It offers multiple methods, including decision trees, and random forests, for predicting cricket match results.is the most effective and trustworthy machine learning library for Python. It provides a wide range of efficient techniques for mathematical modelling and machine learning, including regression, classification, clustering, and dimensionality reduction, through a Python programming interface. This library was mainly developed in Python and is based on NumPy and Matplotlib.

Some of the most widely recognized model that Sklearn provides are as follows:

The Supervised Learning Algorithms Scikit-learn contains almost all popular supervised learning techniques, such as Linear Regression, Support Vector Machine (SVM), Decision Tree, and others. On the other hand, it also encompasses all of the widely used unsupervised learning techniques, such as unsupervised neural networks, factor analysis, PCA, and clustering.

Tensor Flow

Tensor Flow is an open-source machine learning library developed by Google. It can be used for cricket score prediction by building a deep neural network model that takes in features such as batting averages, bowling averages, and team rankings. Data sets that are established as computational nodes in a graph-like structure are used by the TensorFlow software. When the edges integrating the nodes in a structure represent multidimensional vectors or matrices, tensors are produced. TensorFlow programmed use a data flow architecture that functions with standardized intermediate outcomes of the calculations, making them especially well-suited to applications involving very large-scale parallel processing, with neural networks providing as a prevalent instance.

The framework includes sets of both high-level and low-level APIs. Google recommends employing the high-level ones as often as possible to accelerate the development of data pipelines and application programming. The company maintains that exploration and

application problems may gain benefit from knowledge of the low-level TensorFlow Core APIs.

XG Boost

It is an optimized distributed gradient boosting library that can be used for cricket score prediction. It is designed to handle large data-sets and can be used to build a model that takes in various features to guess the output of a game. Understanding XGBoost requires a thorough understanding of the mathematical foundations of decision trees, gradient boosting and supervised machine learning. In supervised machine learning, a model is trained using algorithms for identifying patterns throughout an assortment of characteristics and labels. The model is then applied for predicting the labels on the features of a new dataset.

The computational capacity for boosted tree algorithms is increased to its limit by the gradient boosting tool XGBoost. It is exceptionally precise and adaptable. Its main purpose in development was to make machine learning models more effective and efficient. For Python and R, the first XGBoost implementations were created.

As a result of its widespread usage, XGBoost now includes package implementations for languages including Scala, Julia, java, Perl, and others. When implementing XGBoost, usability, speed, and performance on big data sets were all given the greatest importance. It doesn't require parameter optimization or modifications; therefore, it may be used right away after downloading with no additional settings. The normalization tool provided by XGBoost gives you control over fitting. By employing the weighted quantitative measure sketch method, Boost can handle crowded data sets.

Simulation Models

Simulation models use computer programs to simulate the outcome of a game. These models take into account various factors such as the players' skill levels, the teams' form, and the weather conditions to predict the outcome of a game. Simulator models are frequently known. To put it another way, we are aware of how to take input data, perform a computation, and find the output. Unknown are the inputs.

We don't really know the values of the inputs because they are random variables (at least some of them). A probability distribution is created from expert estimations or it is fitted to the

input using historical data. By selecting input values at random and repeatedly calculating the output, the aim is to discover a range of results. A computer program requires a playground to experiment with concepts and learn from its failures and successes. Such a setting could exist in the real world or online. There are no restrictions on simulation models; they are practically entirely free and can be built up in a controlled manner.

3.4 ANALYSIS AND DESIGN THROUGH UML DIAGRAM

3.4.1 Class Diagram

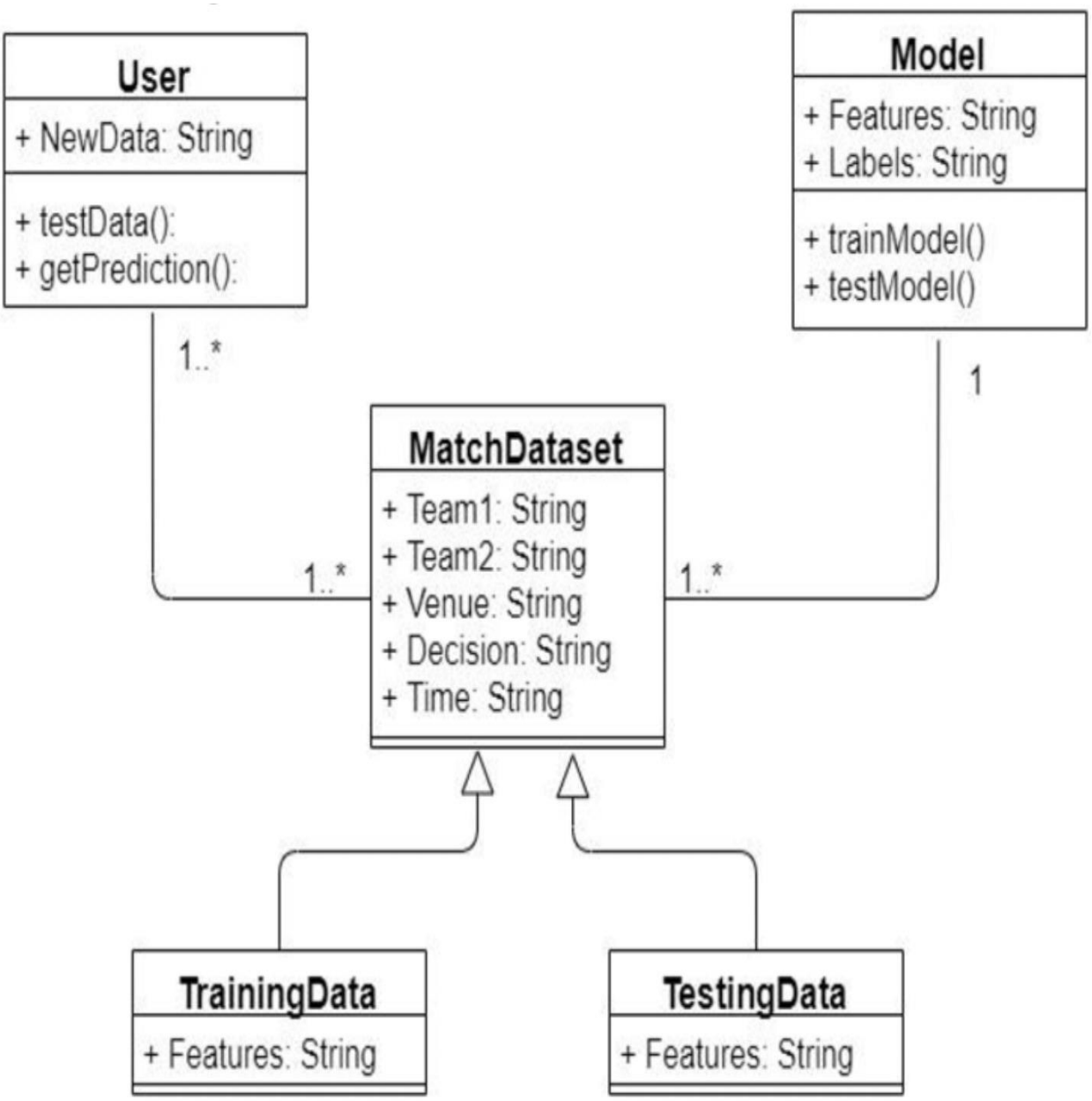


Fig 3.2 Class Diagram

3.4.2 Sequence Diagram

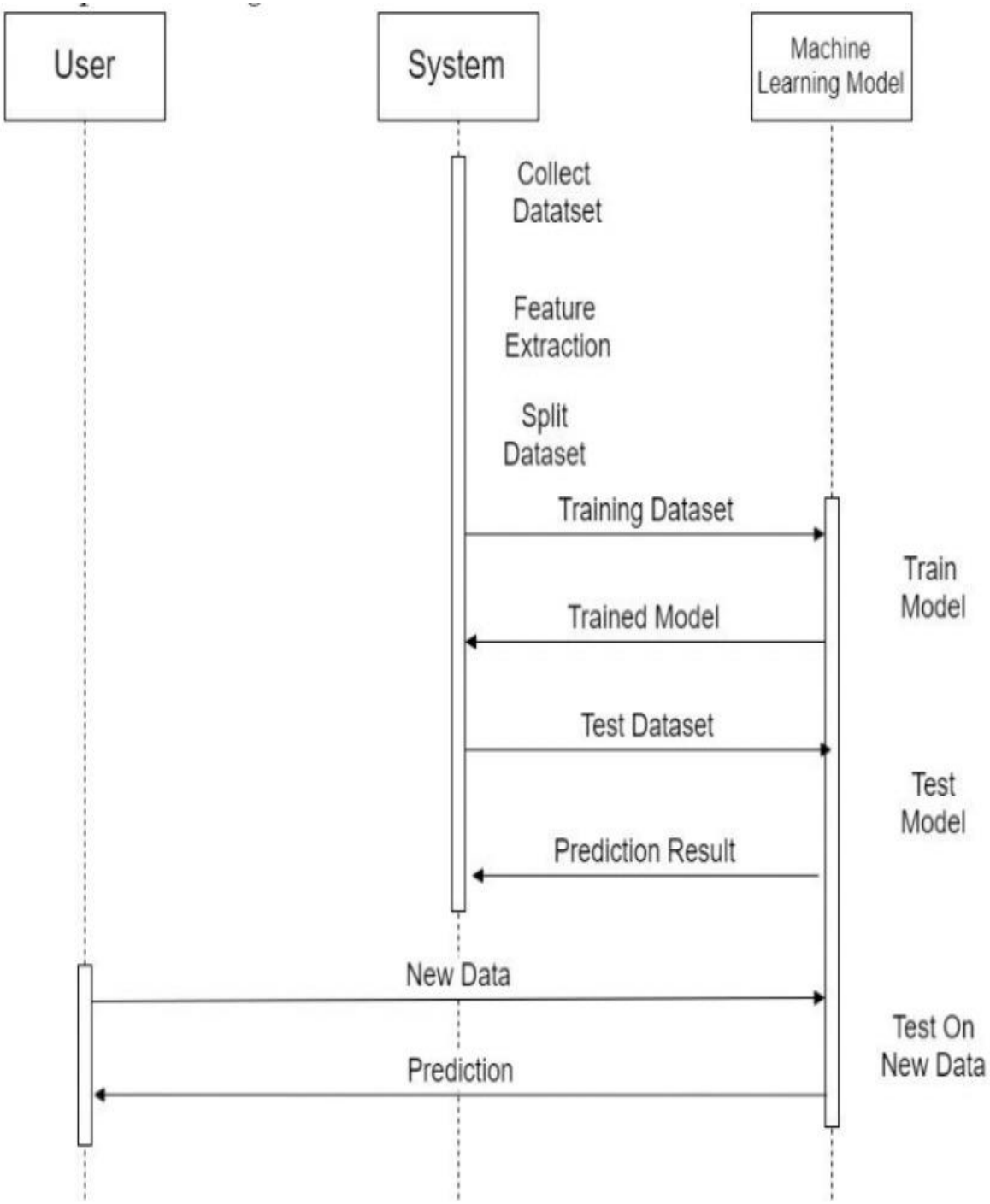
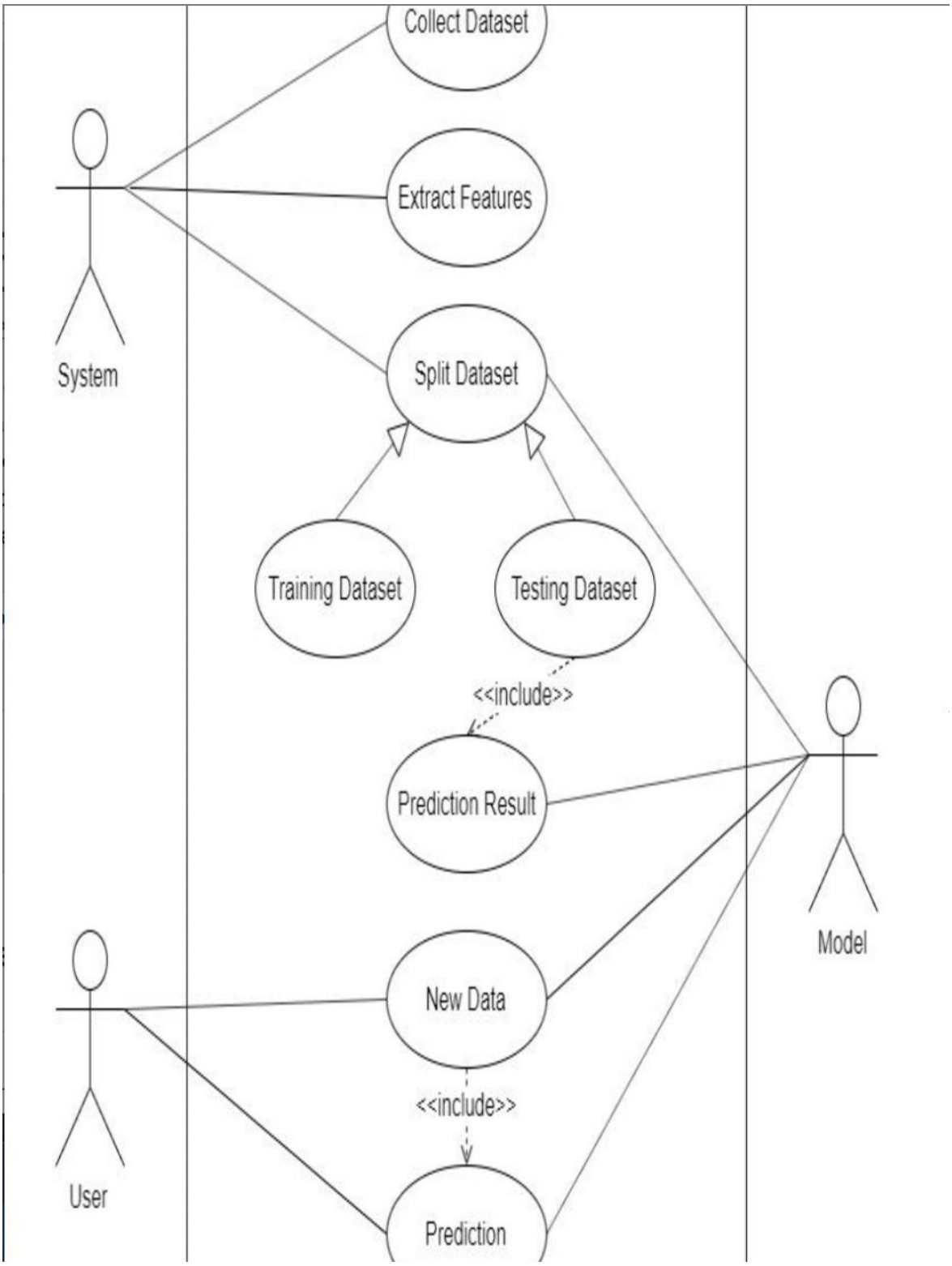


Fig 3.3 Sequence Diagram

3.4.3 Use Case Diagram

Fig 3.4 Use Case Diagram



3.4.4 Activity Diagram

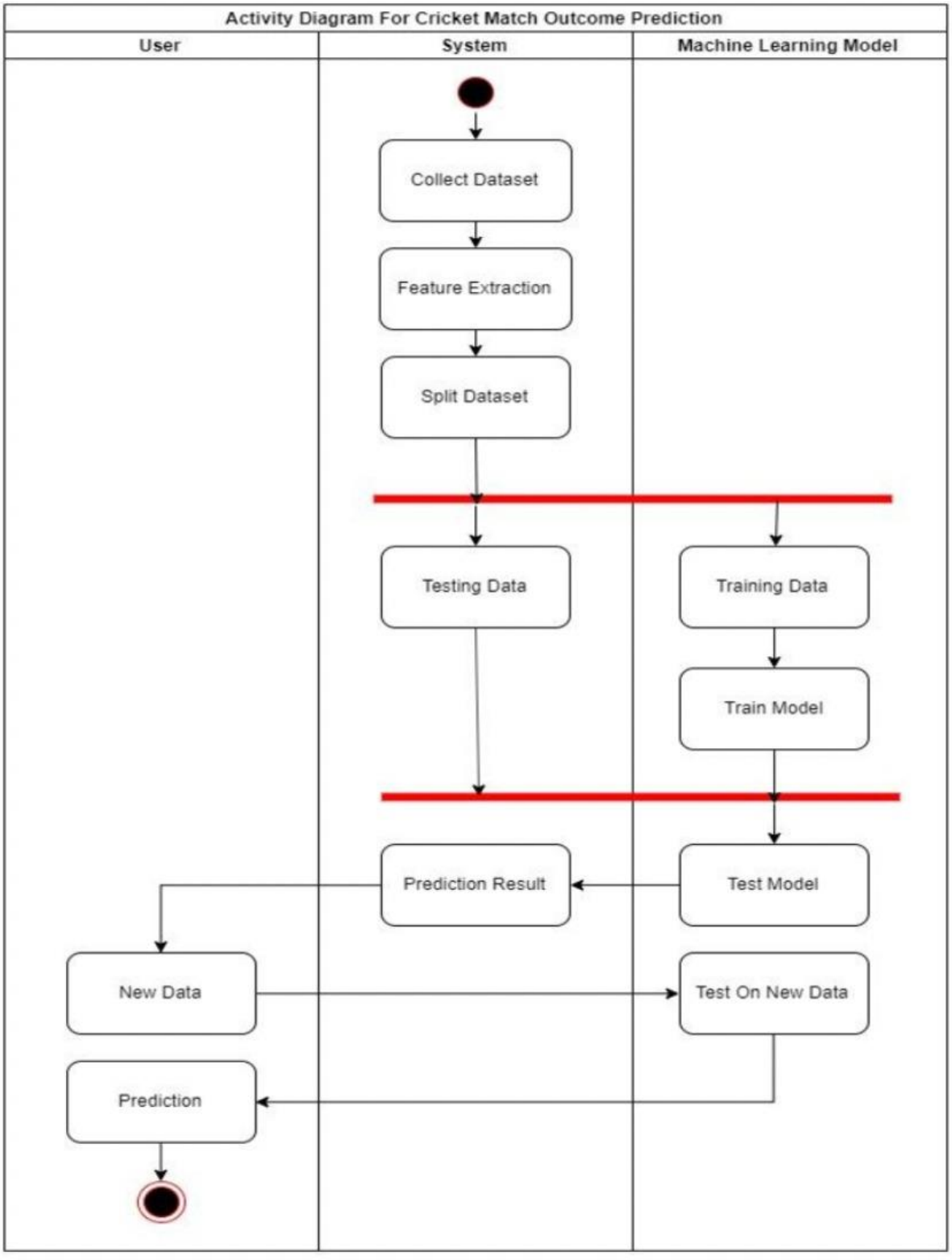


Fig 3.5 Activity Diagram

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 DESCRIPTION OF DATASETS

The datasets for our cricket score prediction is from kaggle.com. The Kaggle dataset itself contains multiple datasets that are required for the cricket score prediction using machine learning. Here are some of the datasets that perform the analysis:

1. Dataset Players
2. Dataset with Venues
3. Dataset with Player's performance
4. Dataset with Average score at venue
5. Dataset with Player's scores at individual venues

These Datasets comprise all the Cricket Matches in a particular interval of time. It consists of these files

1. Original Dataset.csv Raw Dataset File which I scraped using Pandas
2. CategorialDataset.csv Categorial Features suitable for models like MLP Classifier & DT Classifier
3. Continuous Dataset.csv Purely for Experimental Purposes
4. Labelled Dataset.csv Suitable for Support Vector Machines

4.2 EXPERIMENTAL RESULTS

1. The data set contains data from the previous five years to forecast scores. The data used for victory predictions spans a period of seventeen years.
2. We split the data into training and testing portions at a ratio of 70 to 30. 70% of the data is used for training, while 30% is tested.
3. Training - Training can be carried out using training data obtained from the data set. The system will learn from the data about the pattern and different relationships with the aid of this training.
4. Testing - Data testing is done to determine if the training phase was successful or not. The testing data is used to test the data after the training to see whether the machine learning algorithm's prediction or computation was correct or incorrect.

5. Supervised machine learning is a technique used to train an algorithm to carry out the same job on a variety of data sets to uncover patterns and relationships. To train the system, supervised learning provides data with examples and results.
6. Naive Bayes: They are very ascendable and call for a set of parameters that are linearly spaced out from the number of variables in a learning problem. Most of the time, maximum-likelihood coaching involves linearly evaluating a closed-form phrase.
7. To produce a better forecast, linear regression repeats the same task repeatedly. Future values are predicted using this model.
8. Score model: This model displays a numeric number that has been calculated and predicted using a variety of algorithms.
9. Evaluate model - This model is used to evaluate whether the prediction is right or wrong.

Out[6]:

	Unnamed: 0	match_id	venue	innings	ball	batting_team	bowling_team	striker	non_striker	bowler	runs_off_bat	extras	wicket_type	player_dism
0	0	335982	M Chinnaswamy Stadium	1	0.1	Kolkata Knight Riders	Royal Challengers Bangalore	SC Ganguly	BB McCullum	P Kumar	0.0	1.0		
1	1	335982	M Chinnaswamy Stadium	1	0.2	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	SC Ganguly	P Kumar	0.0	0.0		
2	2	335982	M Chinnaswamy Stadium	1	0.3	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	SC Ganguly	P Kumar	0.0	1.0		
3	3	335982	M Chinnaswamy Stadium	1	0.4	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	SC Ganguly	P Kumar	0.0	0.0		
4	4	335982	M Chinnaswamy Stadium	1	0.5	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	SC Ganguly	P Kumar	0.0	0.0		

Fig 4.1 Output

This is an experimental result of a match between Royal Challengers Bangalore and Kolkata Knight Riders.

Fig 4.2 Output

Out[3]:

	Team	Player	Tournament	Matches	Batting Innings	Not Out	Runds Scored	Highest Score	Batting Average	Balls Faced	...	Runs Conceded	Wickets Taken	Best Bowling Figures	Bowling Average	Bowling Economy Rate	Bowling Strike Rate
0	Delhi Daredevils	CH Morris	IPL 2016	12	7	4	195	82*	65.00	109	...	308	13	2/30	23.69	7.00	20.3
1	Delhi Daredevils	CH Morris	IPL 2017	9	9	4	154	52*	30.80	94	...	240	12	4/26	20.00	7.74	15.5
2	Delhi Daredevils	CH Morris	IPL 2018	4	4	3	46	27*	46.00	26	...	143	3	2/41	47.66	10.21	28.0
3	Delhi Daredevils	JP Duminy	IPL 2016	10	8	3	191	49*	38.20	156	...	55	2	1/4	27.50	7.85	21.0
4	Delhi Daredevils	Q de Kock	IPL 2016	13	13	1	445	108	37.08	327	...	-	-	-	-	-	-

This is the experimental result of performances of Delhi Daredevils players.

Fig 4.3

Out[16]:

	venue	innings	ball	batting_team	bowling_team	striker	non_striker	bowler	run	wickets	...	Runs Conceded	Wickets Taken	Best Bowling Figures	Bowling Average	Bowling Economy Rate	Bowlin Strik Rat
0	15	1	0.1	7	13	186	30	201	1.0	0.0	...	1	1	1	1	1	
1	15	1	0.2	7	13	30	184	201	0.0	0.0	...	0	0	0	0	0	
2	15	1	0.2	7	13	30	184	201	0.0	0.0	...	0	0	0	0	0	
3	15	1	0.2	7	13	30	184	201	0.0	0.0	...	0	0	0	0	0	
4	15	1	0.3	7	13	30	184	201	1.0	0.0	...	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
5186	31	1	6.5	14	10	127	47	220	1.0	1.0	...	0	0	0	0	0	
5187	31	1	6.5	14	10	127	47	220	1.0	1.0	...	0	0	0	0	0	
5188	31	1	6.6	14	10	48	126	220	1.0	1.0	...	0	0	0	0	0	
5189	31	1	6.6	14	10	48	126	220	1.0	1.0	...	0	0	0	0	0	
5190	31	1	6.6	14	10	48	126	220	1.0	1.0	...	0	0	0	0	0	

Fig 4.4 Output

4.3 PROPOSED METHOD AND THEIR ADVANTAGES

The proposed methods are statistical models that include regression analysis, time series analysis and machine learning algorithms. The second one is scikit learn it is one of the libraries used in python. The next one is tensor flow which is a machine learning library. The next one is XG boost which is a boosting library. And the last one is Simulation Models which use computer programs to simulate the outcomes of a game. Below are some advantages of cricket score prediction.

1. We can predict the outcomes of a match even before a match starts.
2. It helps us in changing the game plans and strategies for the team coaches and supporting staff.
3. Model that can be updated along with the instant modifications and changes.
4. Takes all important features into account the players playing in each match.
5. Good accuracy arrived through both prediction models.
6. It may prove helpful for numerous stakeholders to use machine learning to study cricket matches while taking, player performance, archaeological game data, ecological criteria, preceding game conditions, and other features into account.

CHAPTER 5
CONCLUSION AND FUTURE ENHANCEMENTS

5.1 SUMMARY

Currently, there is an algorithm that can figure out the team's final score based on the present run rate. It doesn't take into consideration the players' performance or other factors. There are two models in this system. The first model uses the present scenario to anticipate the score a team will receive after the inning. The second model makes victory percentage predictions for both sides prior to the contest even beginning.

The research is carried out using historical data in this case. Data preprocessing, visualization of data, data preparation, feature selection, and the implementation of various machine learning algorithms are just a few of the data science subfields that will converge.

1. To improve the general attraction to the Premier League.
2. To predict the cricket score.
3. Effective prediction technique.
4. Essential for making strategic decisions.

Predicting cricket match scores entails predicting the anticipated results. It helps with betting and wagering, improves fan interaction, enables performance analysis, supports broadcast and commentary, and contributes to research and statistical analysis, among other things.

Teams and players can decide on required run rates, formulate batting and bowling plans, and decide on fielding locations during games by using cricket score predictions. By enabling fans to take part in fantasy leagues, make predictions, and engage in friendly competition, accurate predictions can improve the fan experience.

Score predictions are crucial in the realm of sports betting for bookmakers and bettors. While forecasts are used by gamblers to make informed bets, bookmakers use them to set odds and betting lines. Accurate predictions support fair and transparent betting practices.

Score projections assist performance analysis by monitoring and assessing the performance of teams and individual players. By comparing actual results with predicted scores, analysts can identify trends, patterns, strengths, weaknesses, and opportunities for improvement.

Additionally, cricket score prediction aids in statistical study in the area of sports analytics. The accuracy of forecasts is examined, new algorithms and models are created, and a deeper comprehension of the variables affecting cricket scores is gained by researchers and statisticians. The choice of players, the make-up of the squad, and performance projection can all benefit from this information.

During live matches, score predictions in commentary and broadcasting offer viewpoints and analysis. Commentators frequently use projections to discuss potential outcomes, historical events, records, and pivotal moments in the game. This raises the bar for analysis and viewer participation. Cricket score prediction seeks to offer insightful information, aid in decision-making, promote fan engagement, permit ethical betting, help performance analysis, enhance broadcast and commentary, and advance statistical research in the sport of cricket.

Cricket score prediction seeks to benefit many stakeholders by offering insightful information, supporting their decision-making, and improving their overall comprehension and enjoyment of the game. It's vital to remember that cricket score prediction should be considered entertainment or an analytical tool rather than a trustworthy way to anticipate how a match will turn out. The fluid nature of the game and unforeseen occurrences that can happen during a match cannot be taken into account by predictions, even though they can offer insights and probabilities based on historical data and trends.

Because of these drawbacks and uncertainties, it is advised to proceed with caution while making cricket score predictions. Instead of relying exclusively on predictions, it is always preferable to appreciate the sport and its unpredictable nature.

5.2 OBJECTIVE

Depending on the environment and application, the goals of cricket score prediction can change. The following are some typical goals for cricket score prediction:

Cricket score prediction can aid teams and individuals in making wise strategic choices during a game. The required run rate, batting and bowling techniques, fielding positioning, and tactical modifications can all be planned out using the expected score prediction.

Increasing Fan Engagement: For cricket fans, predicting the results increases excitement and engagement. They can join in fantasy leagues, make predictions, and install a spirit of rivalry among themselves. The overall spectator experience can be improved and the interest in the game can increase with accurate score projections.

Betting and Wagering: For bookmakers and gamblers, cricket score prediction is crucial in the context of sports betting. While bettors rely on forecasts to make wise wagers, predictions also assist bookmakers in setting odds and betting lines. Predictions that are accurate can support honest and open betting procedures.

Performance analysis: The performance of teams and individual players can be evaluated using cricket score prediction. Analysts can spot trends, patterns, strengths,

weaknesses, and areas for improvement by comparing actual results with anticipated scores. It aids in comprehending how well various plans and tactics used by teams.

Score projections can be used in cricket broadcasts and commentary to offer context and analysis during live matches. When discussing potential, achievements, records, and crucial junctures in the game, commentators frequently make reference to score predictions. It gives viewers another level of insight and participation.

Cricket score prediction makes contributions to the fields of research and statistical modelling in sports analytics. Researchers and statisticians can investigate forecast accuracy, create fresh algorithms and models, and learn more about the variables affecting cricket scores. The choice of players, the make-up of the squad, and performance projection can all benefit from this information.

In general, cricket score prediction seeks to benefit a variety of stakeholders, including teams, players, spectators, broadcasters, and analysts, by offering insightful information, assisting in decision-making, and improving the overall experience and comprehension of the game.

Predicting a cricket match's outcome or final score with precision is the primary objective of cricket score prediction. Making an accurate forecast requires A looking at a variety of criteria, including team performance, player form, pitch conditions, weather conditions, and other significant information. Predicting cricket scores raises spectator interest and involvement in the sport. The fans' involvement in prediction contests, fantasy leagues, and score prediction betting increases the excitement and entertainment value of the game.

Cricket score projections can improve the viewing experience by giving spectators more information and analysis. Fans will have a better understanding of the game's dynamics and possible results thanks to the integration of predictions into live commentary and analysis. Cricket score prediction requires statistical analysis of a significant quantity of historical data, player statistics, and match circumstances. This procedure contributes to the development of sophisticated analytics tools and methods for cricket analysis by improving statistical models and algorithms.

Evaluation of team and player performance is possible thanks to cricket score prediction. Teams and players can evaluate their strengths and weaknesses, pinpoint areas for improvement, and make data-driven adjustments to their strategy by comparing actual results

with expected scores. Research and development-The goal of cricket score prediction includes sports analytics research and development.

It takes constant innovation and refinement to create reliable prediction models, which advances data analysis methods and machine learning algorithms. All the machine learning algorithms we use here for the cricket score prediction gives us an individual output but the desired prediction will be accurate enough.

Overall we can finally say objectives of cricket score prediction are

- Accuracy
- Decision-making
- Fan engagement
- Broadcast enhancement
- Statistical analysis
- Performance evaluation
- Research and development

5.3 RESULTS

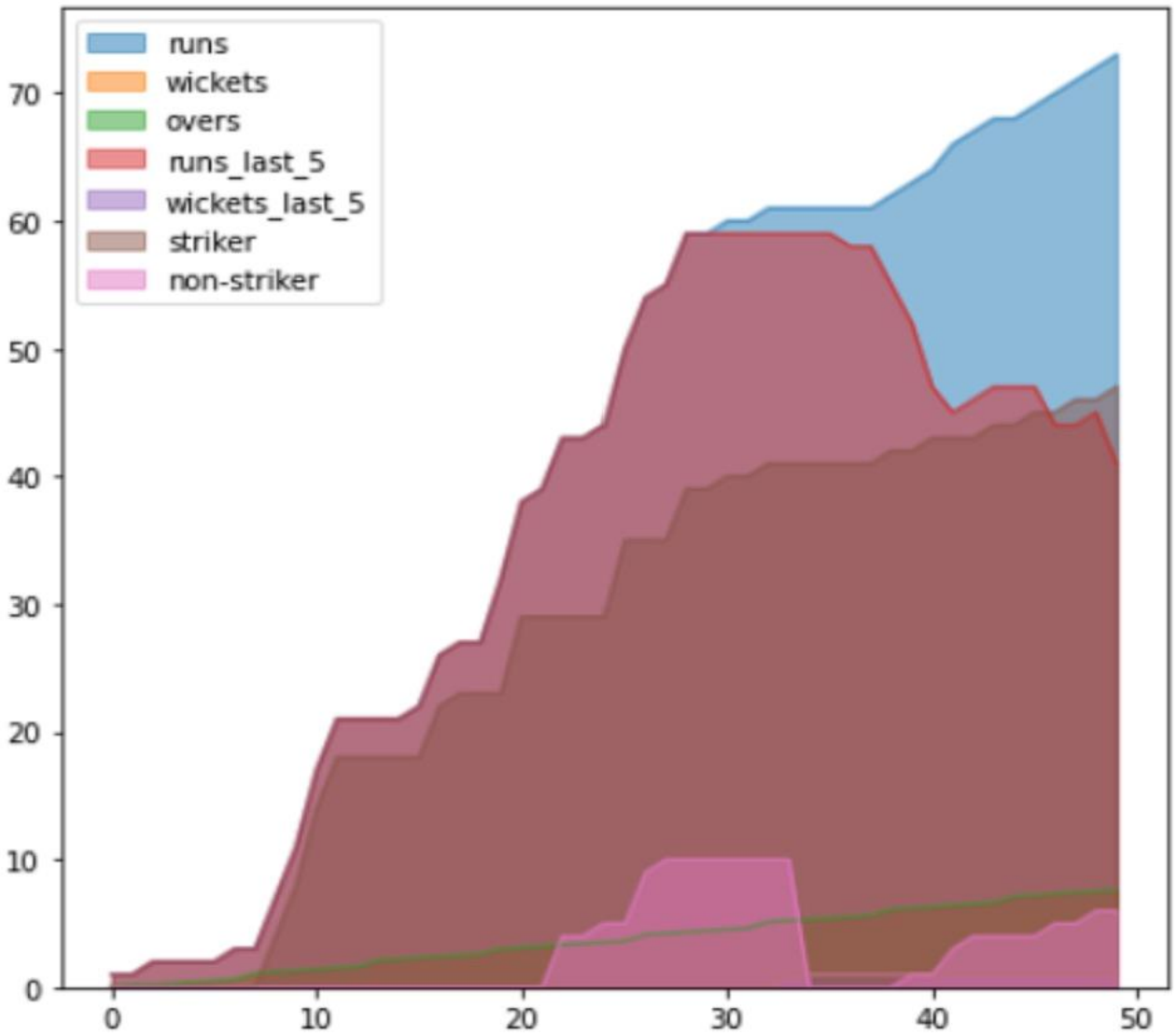


Fig5.1 Output

```
Out[38]: 9.314617116412084
```

Fig 5.2 Output

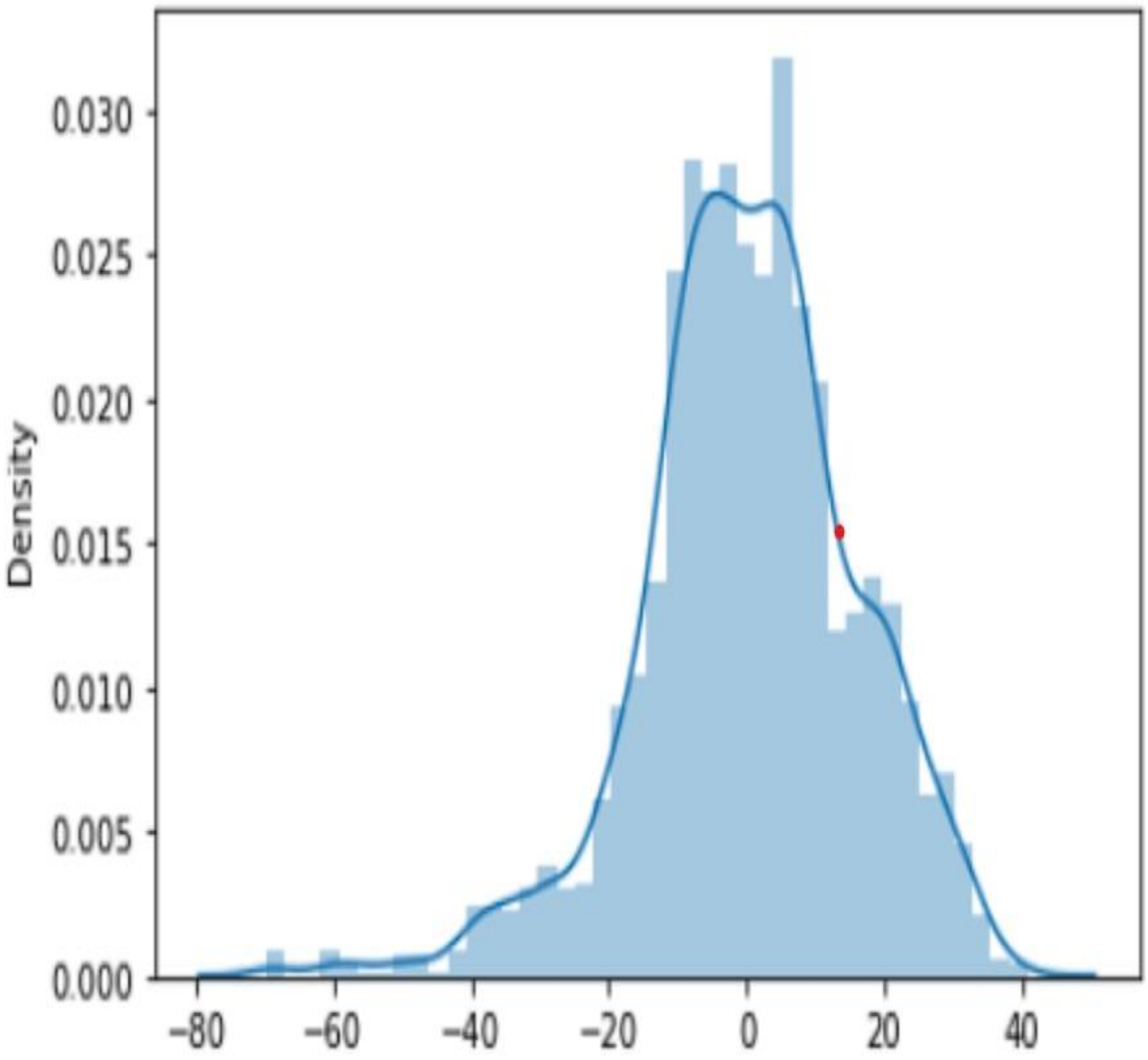


Fig 5.3 Output

Fig 5.3 Output

```
Out[39]:  11.856571876802397
```

5.4 CONCLUSION

The purpose of this study is to predict the ultimate score and match winner using historical data. To conduct the research and predict the outcome of the match, experts from several fields of data science will join. These include data pre-processing, data visualizations, data preparation, data selection, and machine learning model implementation.

To accurately forecast the number of innings and provide the desired outcome, a variety of machine learning models will be applied to the provided data. The results suggest that the Linear Regression method has the best prediction accuracy. As a result, we are employing a linear regression model to make predictions. Teams can use CFP to predict the final score before the 20 overs are bowled. The teams will be able to anticipate when to accelerate and when to play aggressively to increase the run rate while establishing an objective.

CFP can be used to assist team management in choosing a squad that can enhance performance. It can be used by fans of cricket to forecast the eventual outcome of a real-time IPL match. Strategic decision-making requires the system. It's a method that incorporates user input.

This system effectively handles a sizable data set and updates the data sets maintained with each forecast. Due to data clustering this system prioritizes player performance and processes information relatively quickly. Decision Trees, Logistic Regression, Linear Regression, Random Forests, and other techniques are employed.

5.5 FUTURE ENHANCEMENTS

1. We can use a model to predict the possibility of chasing in the future. In other words, the algorithm may be able to predict whether a team would be effective in achieving the goal.

2. The model utilized in this project can be made more accurate. The prediction can take into account variables like the venue, the playing surface, and the opposition team.

3. Additional factors such as batsmen partnerships and pitch conditions, can be introduced to further improve the model's accuracy.

CHAPTER 6

APPENDICES

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn import preprocessing

ipl = pd.read_csv('ipl.csv')
ipl.head()
```

```
data = pd.read_csv('ipl_player.csv')
data.head()

str_cols = new_ipl.columns[new_ipl.dtypes==object]
newIpl[str_cols] = newIpl[str_cols].fillna('.')

listf = []

a1 = newIpl['venue'].unique()
a2 = newIpl['batting_team'].unique()
a3 = newIpl['bowling_team'].unique()
a4 = newIpl['striker'].unique()
a5 = newIpl['bowler'].unique()

def labelEncoding(data):
    dataset = pd.DataFrame(newIpl)
    feature_dict ={}

    for temp in dataset:
        if dataset[temp].dtype==object:
            le = preprocessing.LabelEncoder()
            fs = dataset[temp].unique()
            le.fit(fs)
            dataset[temp] = le.transform(dataset[temp])
            feature_dict[temp] = le

    return dataset

labelEncoding(newIpl)
```

```
iplDataset = newIpl[['venue','innings', 'batting_team',
                     'bowling_team', 'striker', 'non_striker',
                     'bowler']]

b1 = iplDataset['venue'].unique()
b2 = iplDataset['batting_team'].unique()
b3 = iplDataset['bowling_team'].unique()
b4 = iplDataset['striker'].unique()
b5 = iplDataset['bowler'].unique()
newIpl.fillna(0,inplace=True)

temp={}

for i in range(len(a1)):
    features[a1[i]]=b1[i]
for i in range(len(a2)):
    features[a2[i]]=b2[i]
for i in range(len(a3)):
    features[a3[i]]=b3[i]
for i in range(len(a4)):
    features[a4[i]]=b4[i]
for i in range(len(a5)):
    features[a5[i]]=b5[i]
temp

X = newIpl[['venue', 'innings','batting_team',
            'bowling_team', 'striker','bowler']].values
y = newIpl['y'].values
```

```
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(
  X, y, test_size=0.33, random_state=42)

from sklearn.preprocessing import MinMaxScaler
scaler = MinMaxScaler()

X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

model = Sequential()
model_losses.plot()

predictions = model.predict(X_test)
sample = pd.DataFrame(predictions,columns=['Predict'])
sample['Actual']=y_test
sample.head(10)

from sklearn.metrics import mean_absolute_error,mean_squared_error
mean_absolute_error(y_test,predictions)

np.sqrt(mean_squared_error(y_test,predictions))
```

REFERENCES

1. Available at https://www.semanticscholar.org/paper/CRICKET-MATCH-OUTCOME-PREDICTION-USING-MACHINE-Naik-Pawar/4fc5b8dec6ee0281a63c75f93e4a040464e52c2a
2. Available at https://ijirt.org/master/publishedpaper/IJIRT157821_PAPER.pdf
3. Prasad Thorat, Vighnesh Buddhivant, Yash Sahane; Review Paper on Cricket Score Prediction; April 2021
4. Tejinder Singh, Vishal Singla, Parteek Bhatia; - Score and Winning Prediction in Cricket through Data Mining Oct 8-10,2015
5. D. Thenmozhi, P. Mirunalini, S. M. Jaisakthi, Srivatsan Vasudevan , Veeramani Kannan V, SagubarSadiq S; Moneyball - Data Mining on Cricket Dataset; 2019

6. A.N.Wickramasinghe, Roshan D.Yapa; Cricket Match Outcome Prediction Using Tweets and Prediction of the Man of the Match using Social Network Analysis: Case Study Using IPL Data; 2018
7. Nigel Rodrigues1, Nelson Sequeira2, Stephen Rodrigues3, Varsha Shrivastava4; Cricket Squad Analysis using multiple Random Forest Regression;2019
8. Animal Islam Anik, Sakif yeaser, A.G.M. Emam Hussain, Amitabha Chakraborty; Player's Performance Prediction in ODI Cricket Using Machine Learning Algorithms;2018
9. Siyamalan Manivannan, Mogan Kausik; Convolutional Neural Network and Feature Encoding for Predicting the Outcome of Cricket Matches;2019
10. Manuka Madranga Hatharasinghe, Guhanathan Poravi Data Mining and Machine Learning in Cricket Match Outcome Prediction: Missing Link;2019
11. Jalaz Kumar, Rajeev Kumar, Pushpender Kumar; Outcome Prediction of ODI Cricket Matches using Decision Trees and MLP.
12. Pallavi Tekade, Kunal Markad, Aniket Amage and Bhagwat Natekar (2020). "CRICKET MATCH OUTCOME PREDICTION USING MACHINE LEARNING" .
13. Prof. R. R. Kamble, Nidhi Koul, Kaustubh Adhav, Akshay Dixit and Rutuja Pakhare (2021). "Cricket Score Prediction Using Machine Learning" .
14. Rohit Khade, Nikhil Bankar, Prashant Khedkar and Prof. Prashant Ahire (2019). "Cricket Score Prediction using Machine Learning Algorithms" .
15. Omkar Mozar, Soham More, Shubham Nagare and Prof. Nileema Pathak (2022). "Cricket Score and Winning Prediction" .
16. Dhonge, N., Dhole, S., Wavre, N., Pardakhe, M., & Nagarale, A (2021). "IPL Cricket Score and Winning Prediction Using Machine Learning Techniques".
17. Kumash Kapadia, Hussein Abdel-Jaber, Fadi Thabtah, Wael Hadi (2019). "Sport analytics for cricket game results using machine learning: An experimental study".
18. Daniel Mago Vistro, Faizan Rasheed, Leo Gertrude David (2019). "The Cricket Winner Prediction With Application Of Machine Learning And Data Analytics".
19. Prasad Thorat, Vighnesh Buddhivant, Yash Sahane (2021). "CRICKET SCORE PREDICTION".
20. Apurva Lawate, Nomesh Katare, Salil Hoskeri, Santosh Takle, Prof. Supriya. B. Jadhav (2021).