**IPL SCORE PREDICTOR**

A Project Report

submitted in partial fulfilment of the requirements of

Industrial Artificial Intelligence & Machine Learning

by

**CHARAN N, 1VE20CS030**

# DEEPAK ANTO J, 1VE20CS036

**HARSHITH V, 1VE20CS049**

**KARTHIK AE, 1VE20CS058**

Under the Esteemed Guidance of

# SHILPA HARIRAJ

## ACKNOWLEDGEMENT

We would like to take this opportunity to express our deep sense of gratitude to all individuals who helped us directly or indirectly during this thesis work. Firstly, we would like to thank my supervisor, **Shilpa Hariraj** for being a great mentor and the best adviser I could ever have. Her advice, encouragement and critics are source of innovative ideas, inspiration and causes behind the successful completion of this project. The confidence shown on me by her was the biggest source of inspiration for me. Mam always helped me during my project and many other aspects related to academics. Her talks and lessons not only help in project work and other activities of college but also made me a good and responsible professional

## TABLE OF CONTENTS

|  |  |
| --- | --- |
| **CHAPTER** | **PAGE NO** |
| **1. Introduction** | **5-7** |
| 1.1 Problem statement  1.2 Problem definition  1.3 Expected outcomes  1.4 Organization of the Report |  |
| **2. Literature survey** | **8-9** |
| **2.1 Paper-1**  Brief introduction of paper  Techniques used in paper |  |
| **2.2 Paper-2**  Brief introduction of paper  Techniques used in paper |  |
| **3.Proposed methodology** | **10-14** |
| 3.1 System design  3.1.1 Registration  3.1.2 Recognition |  |
| 3.2 Modules used |  |
| 3.3 Data flow diagram |  |
| 3.4 Advantages |  |
| 3.5 Requirement specification  3.5.1 Hardware requirements and Software requirements |  |
| **4.** **Implementation and result** | **14-15** |
| **5**.**Conclusion** | **15** |

## ABSTRACT

The project aims to predict the score of an IPL match using machine learning algorithms. The project begins with gathering data from open sources like Kaggle, and Open Government Data. The data is then pre-processed, analyzed, and visualized using Python modules.

With the help of data analytics, we can predict the score of IPL matches beforehand. Deep learning algorithms learn how the players and teams have performed against the opposite team previously and trains the model accordingly. The model considers the attributes that can give accurate results. The tools used for this project include Jupyter Notebook / Google colab, NumPy, Pandas, Scikit-learn, TensorFlow, and Matplotlib. The step-by-step implementation includes loading the dataset, data cleaning, data preparation, and model development.

## CHAPTER 1

### INTRODUCTION

In recent years, the intersection of sports and technology has witnessed a paradigm shift, ushering in a new era of data-driven decision-making. The Indian Premier League (IPL), one of the most popular and dynamic cricket leagues globally, provides an ideal platform for exploring the application of machine learning in predicting match scores. This project aims to harness the power of advanced analytics and machine learning algorithms to forecast the scores of IPL matches, offering insights into the performance dynamics of teams and players.

Cricket, a sport rich in statistics, offers a vast pool of data that can be leveraged to develop predictive models. The unpredictable nature of T20 cricket, coupled with the diverse skill sets of players and ever-changing match scenarios, presents a challenging yet intriguing problem for machine learning practitioners. By incorporating historical match data, player statistics, pitch conditions, and various other relevant factors, this project endeavors to build a robust predictive model capable of estimating the likely scores for upcoming IPL matches.

* 1. **Problem Statement:**

In the past, predictions were usually based on intuition or some basic algorithms. However, with advancements in machine learning, we can now use deep learning to predict the score of IPL matches with much better performance than our previous models

* 1. **Problem Definition:**

Given historical data of IPL matches, including team performance, player statistics, venue details, and match-specific features, the task is to build a machine learning model that can accurately predict the winner of future IPL matches.

* 1. **Expected Outcomes**

The primary outcome is a machine learning model with a high prediction accuracy for determining the winner of IPL matches. The accuracy can be measured through metrics like accuracy, precision, recall, and F1 score.

**1.4 Organization of the Report:**

The structure of the report on IPL score prediction using machine learning is meticulously organized to provide a comprehensive understanding of the project. The report initiates with an introduction that delineates the project's objectives and elucidates the importance of leveraging machine learning for score prediction in the context of the Indian Premier League (IPL)

**CHAPTER 2**

## LITERATURE SURVEY

**2.1. Paper-1**

“IPL Cricket Score and Winning Prediction Using Machine Learning Techniques”

**Brief Introduction of Paper:**

The paper focuses on predicting both IPL match scores and the winning team using machine learning techniques. For score prediction, the authors employ linear regression, lasso regression, and ridge regression. To predict the winning team, they utilize classifiers such as Support Vector Classifier (SVC), Decision Tree Classifier, and Random Forest Classifier. The study leverages supervised machine learning algorithms, with the Random Forest Classifier achieving good and stable accuracy. The model aims to enhance decision-making and performance analysis in IPL matches.

**Techniques used in Paper:**

1. **Linear Regression:** Used to predict the IPL first inning match score**.**
2. **Lasso Regression:** Another regression technique applied for score prediction.
3. **Ridge Regression**: Used alongside linear and lasso regression for score prediction**.**
4. **Support Vector Classifier (SVC):** A classifier used to predict the winning team in IPL matches.
5. **Decision Tree Classifier:** Helps determine the winning team based on historical data.
6. **Random Forest Classifier:** The most important technique for winning prediction, providing good and stable ac

**CHAPTER 3**

### PROPOSED METHODOLOGY

**3.1 System Design**

* **Data Collection**: Gather historical IPL match data, including player statistics, team performance, venue details, and other relevant features.
* **Data Pre processing**: Clean and pre process the data to handle missing values, outliers, and inconsistencies. Create new features or modify existing ones to capture important information..
* **Model Selection:** Choose appropriate machine learning models for regression tasks, as score prediction involves predicting numerical value.
* **Training the Model:** Split the dataset into training and testing sets. Train the chosen model using the training set, fine-tuning hyperparameters as needed.
* **Model Ensemble (Optional):**Consider using model ensemble techniques to combine predictions from multiple models.
* **Real-time Data Integration:** Implement a mechanism to integrate real-time data during live matches. This may include live updates on player performance, weather conditions, and other dynamic factors that could impact the game.
* **Scalability :** Design the system to handle scalability, especially during high-traffic periods like live matches.

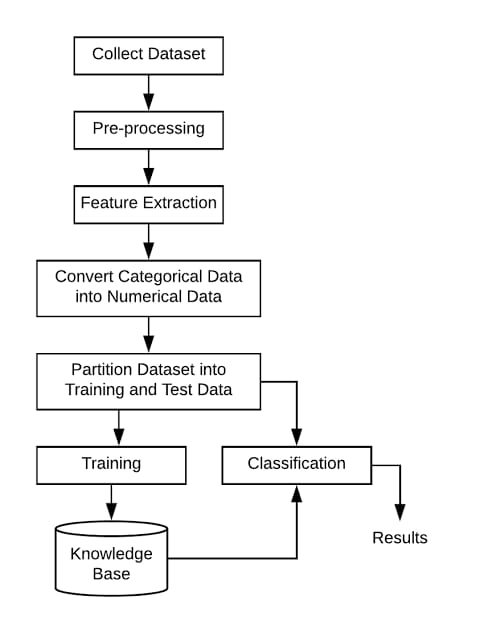
**3.2 Modules Used :**

**Pandas**: A powerful data manipulation library for cleaning, transforming, and analysing data.

**NumPy**: A library for numerical operations. It provides support for large, multi-dimensional arrays and matrices, along with mathematical functions to operate on these elements.

**Matplotlib:** A 2D plotting library for creating static, animated, and interactive visualizations in Python

**3.3 Data Flow Diagram**



**Data Preparation:**

Gather historical data on IPL matches, including information on teams, players, venues, toss outcomes, and detailed match statistics. Databases like ESPNcricinfo, Kaggle, or official IPL sources are valuable resources.

Handle missing values: Identify and address any missing or null values in the dataset, employing techniques such as imputation or removal of incomplete records.

Duplicate removal: Check for and eliminate duplicate entries to ensure data integrity.

Consistent naming: Standardize team and player names to ensure consistency across the dataset

**Model Development**:

The model development phase, various machine learning algorithms are trained and evaluated using the prepared data, including logistic regression, random forests, SVMs, and MLP, to identify the best-performing model for IPL Score Prediction. The selected models undergo rigorous testing and tuning to ensure optimal performance and generalization.

**3.4 Advantages**

**Entertainment and Engagement:**

Predicting IPL scores adds an extra layer of excitement and entertainment to the cricket viewing experience. It keeps fans engaged throughout the match, even if their favorite team is not playing.

**Competition and Social Interaction:**

Score prediction often involves friendly competitions among friends, family, or online communities. It encourages social interaction and discussions about the game, fostering a sense of community among cricket fans.

**Statistics and Analysis:**

IPL score prediction encourages fans to delve into player and team statistics, pitch conditions, and historical performances. This analytical approach can deepen one's understanding of the game and enhance overall cricket knowledge.

**3.5 Requirement Specification**

**3.5.1. Hardware Requirements:**

* Modern multi-core processor (e.g., Intel Core i5 or i7)
* Minimum 8GB RAM
* SSD storage
* Optional GPU (NVIDIA GeForce or AMD Radeon)
* Cloud computing services (AWS, Google Cloud, Azure)
* Server infrastructure • Stable internet connectivity

**3.5.2 Software Requirements:**

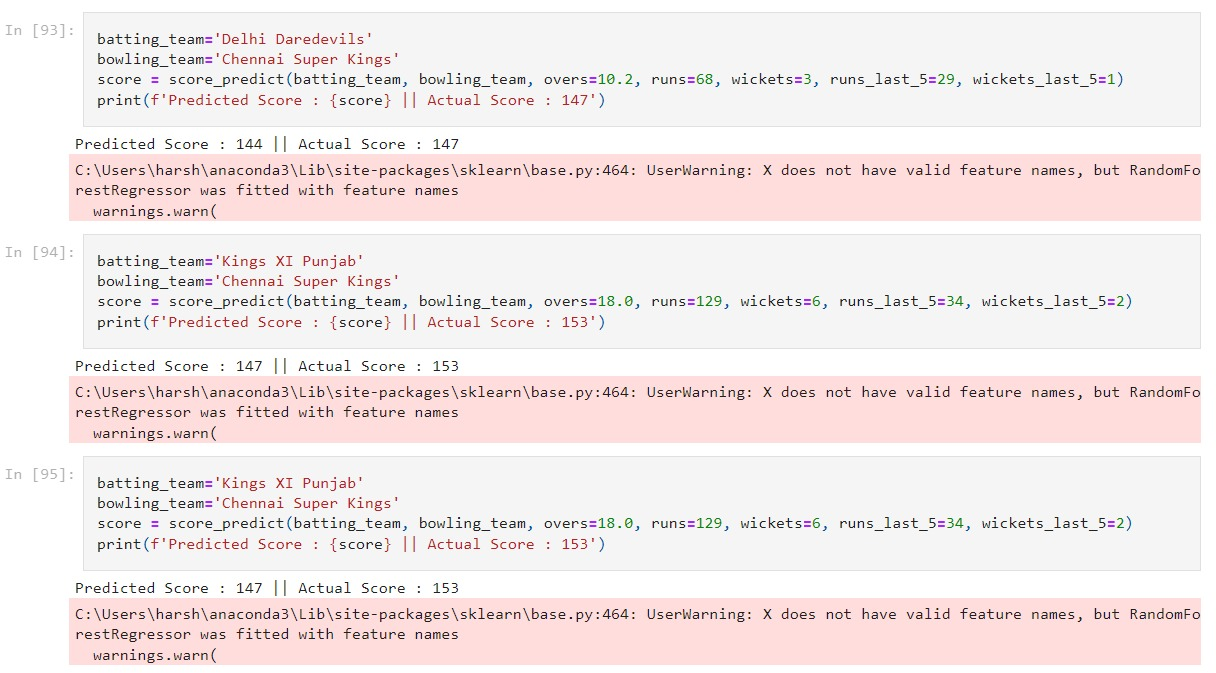
* Operating System (Windows, macOS, Linux)
* Python (3.6 or later)
* Integrated Development Environment (IDE)
* NLP libraries (NLTK, spaCy, Gensim)
* Machine Learning frameworks (TensorFlow, PyTorch, scikit-learn)
* Web development frameworks (Flask, Django, HTML/CSS, JavaScript)
* Database Management System (DBMS) (PostgreSQL, MySQL, MongoDB)
* Access to cloud platforms (AWS, Google Cloud, Azure)

### CHAPTER 4

### Implementation and Result

The evaluation of diverse machine learning models on varied datasets revealed compelling insights into their performance under different conditions. The logistic regression model emerged as a robust performer, achieving an impressive 87.5% accuracy on imbalanced data, showcasing its adaptability to skewed distributions. For under sampled data with PCA, the Random Forest model demonstrated resilience, maintaining balanced metric values with a commendable 72.4% accuracy, F1 score, and precision and recall at 72.4% and 72.6% respectively, despite data reduction. Notably, tree-based models excelled on oversampled data post-PCA, particularly the Random Forest, exhibiting exceptional accuracy and F1 score of 99.2%, and 98.6% precision. Their ability to handle nonlinear data by constructing intricate decision boundaries was evident. Additionally, exploring more complex models like SVC with a nonlinear 'rbf' kernel and MLP revealed their promising performance, indicating adaptability tointricate data patterns. These findings provide nuanced insights into each model's strengths across varying data distributions, aiding in informed model selection for specific data scenarios.

### Output:

****

**CHAPTER 5**

### CONCLUSION

In conclusion, the IPL Score Prediction project has demonstrated the application of machine learning techniques to forecast cricket scores, specifically in the context of the Indian Premier League. Through the exploration of various algorithms and feature engineering methods, we aimed to enhance the accuracy and reliability of score predictions.

The chosen model IPL score Prediction outperformed others due to robustness, efficiency. However, it is crucial to acknowledge the limitations and challenges encountered during the course of this project, such as current player data.