## Artificial Intelligence

## and Machine Learning

Project Report

Semester-IV (Batch-2022)

IPL SCORE PREDICTOR

A red and white sign

Description automatically generated with low confidence

**Supervised By: Submitted By:**

**Ms. Shagun Sharma** Harshdeep Singh 2210990383

Harsimrat Singh 2210990395

Himanshu Sharma 2210990407

Himanshu Singh 2210990409

Group 1 precisionNpredictions

**Department of Computer Science and Engineering**

## Chitkara University Institute of Engineering & Technology,

## Chitkara University, Punjab

**INTRODUCTION:**

The Indian Premier League (IPL) has revolutionized the landscape of cricket, captivating millions of fans worldwide with its electrifying matches and nail-biting finishes. Amidst the frenzy of the IPL, the quest to anticipate match scores has become a tantalizing challenge for data scientists and enthusiasts alike. In this report, we embark on a thrilling journey into the realm of Artificial Intelligence and Machine Learning (AIML) to predict IPL match scores with precision and insight.

Cricket, with its multifaceted dynamics influenced by factors ranging from pitch conditions to player form, presents a formidable puzzle for predictive modelling. Yet, it is precisely this complexity that fuels our curiosity and drives the exploration of innovative AIML techniques. By leveraging historical match data, player statistics, weather conditions, and other pertinent variables, AIML endeavours to unravel the intricacies of cricket scoring and offer glimpses into the future of IPL matches.

This report serves as a comprehensive guide to the methodologies, techniques, and algorithms employed in the pursuit of accurate IPL score prediction. From data preprocessing and feature engineering to model selection and evaluation, we navigate through the intricate steps of AIML modelling, striving for both accuracy and interpretability.

Furthermore, we delve into the nuances of cricket scoring, analysing the impact of various factors such as batting order, pitch type, ground dimensions, and match context on final scores. Through meticulous analysis and experimentation, we aim to uncover hidden patterns and insights that elucidate the underlying dynamics of IPL matches.

Our endeavor extends beyond mere prediction; we seek to empower stakeholders in the cricket ecosystem, from fans and analysts to team management, with actionable insights derived from AIML models. By understanding the determinants of match scores, we aspire to enhance strategic decision-making, optimize team performance, and enrich the viewing experience for cricket enthusiasts worldwide.

As we embark on this odyssey into the future of cricket scoring, we acknowledge the inherent uncertainties of the game and the evolving nature of AIML methodologies. Through collaboration, experimentation, and continuous refinement, we endeavor to push the boundaries of predictive modelling and pave the way for a new era of data-driven insights in cricket.

Join us on this exhilarating expedition as we unravel the mysteries of cricket scoring and harness the power of AIML to illuminate the path forward in the captivating world of the Indian Premier League.

**Objective**:

The primary objective of this AIML report for IPL score prediction is to develop robust predictive models that accurately forecast the scores of IPL matches. By leveraging historical data and advanced machine learning techniques, we aim to provide stakeholders with valuable insights into the potential outcomes of matches, thereby enhancing strategic decision-making, optimizing team performance, and enriching the overall cricketing experience for fans and analysts alike.

**Significance:**

The significance of this endeavor lies in its potential to revolutionize the way cricket matches are perceived, analysed, and strategized. By employing AIML methodologies to predict IPL scores, we aim to:

1. Enhance Fan Engagement: Accurate score predictions enable fans to anticipate match outcomes and immerse themselves more deeply in the excitement of IPL matches.
2. Inform Strategic Decisions: Teams can utilize predictive insights to tailor their strategies, optimize player selection, and adapt their gameplay based on anticipated match scores and scenarios.
3. Facilitate Betting and Fantasy Cricket: Predictive models provide valuable information for betting enthusiasts and fantasy cricket players, enabling informed decision-making and enhancing the overall gaming experience.
4. Drive Data-Driven Analysis: By uncovering underlying patterns and insights in cricket scoring, this report contributes to the growing field of sports analytics, fostering a deeper understanding of the game among analysts and researchers.
5. Showcase AIML Capabilities: This report serves as a showcase for the potential of AIML in sports analytics, demonstrating the efficacy of advanced algorithms in predicting complex and dynamic phenomena such as cricket match scores.

Overall, the significance of this AIML report extends beyond the realm of cricket, highlighting the transformative power of data-driven insights in sports and underscoring the potential of AIML to reshape decision-making processes across diverse domains.

**Problem Statement:**

The problem addressed in this AIML report revolves around predicting IPL match scores accurately. Given the dynamic nature of cricket matches and the multitude of factors influencing scoring patterns, the challenge lies in developing robust predictive models that can effectively capture and leverage these factors to forecast match scores with high precision. The primary goal is to build models that not only provide accurate predictions but also offer valuable insights into the underlying dynamics of cricket scoring, thereby empowering stakeholders to make informed decisions and enhance their overall cricketing experience.

**Software Requirements:**

1. Programming Language: Python (with libraries such as NumPy, pandas, scikit-learn, TensorFlow/PyTorch for machine learning, and matplotlib/seaborn for visualization).

2. Integrated Development Environment (IDE): Jupyter Notebook for Python development.

3. Data Analysis and Visualization Tools: Data Frame, Pandas, Matplotlib, Seaborn for data exploration and visualization.

4. Machine Learning Libraries: Installation of relevant machine learning libraries for model development i.e. scikit-learn.

**Hardware Requirements:**

1. Processor: Multi-core processor (e.g., Intel Core i5 or higher) for efficient data processing and model training.

2. RAM: Minimum 8 GB RAM to handle large datasets and computational tasks involved in machine learning.

3. Storage: Sufficient disk space to store datasets, software, and models (at least 100 GB recommended).

4. Graphics Processing Unit (GPU) (Optional): For faster model training and inference, a GPU with CUDA support can be beneficial, especially for deep learning models.

**Data Sets:**

1. IPL Match Data: Historical data of IPL matches including match statistics, player performances, match venue, and other relevant details.

2. Player Statistics: Individual performance statistics of players participating in IPL matches, including batting averages, bowling figures, strike rates, etc.

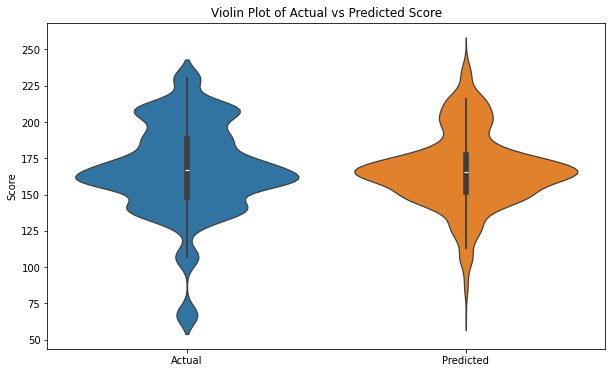
3. Pitch Conditions: Data on pitch characteristics, such as pitch type (e.g., spin-friendly, batting-friendly), historical scores on the pitch, etc.

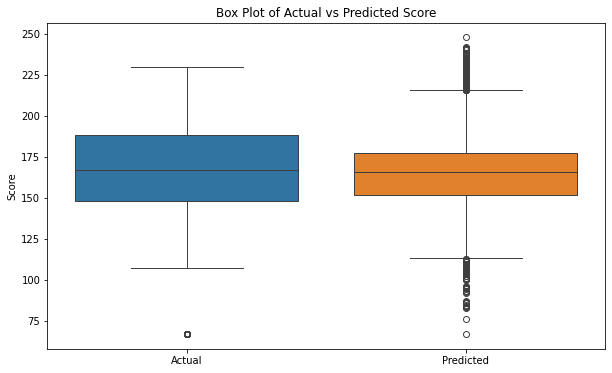
4. Weather Data: Information on weather conditions during matches, including temperature, humidity, wind speed, etc.

These datasets serve as the foundation for training and evaluating predictive models for IPL score prediction. The availability of comprehensive and accurately labelled data is crucial for building effective machine learning models in this domain.

**Schematic Diagram:**

**EDA:**

****

****

**File Structure:**

The file structure of the project can be organized to maintain modularity, clarity, and ease of navigation. It may include the following directories and files:

1. **Data/**: Contains datasets used for training and testing the models.
   * *ipl\_match\_data.csv*: Historical IPL match data.
   * *player\_statistics.csv*: Player performance statistics.
   * *pitch\_conditions.csv*: Data on pitch characteristics.
   * *weather\_data.csv*: Information on weather conditions.
2. **Notebooks:** Contains Jupyter notebooks for data preprocessing, model development, and evaluation.
   * *01\_Data\_Preprocessing.ipynb*: Data cleaning, transformation, and feature engineering.
   * *02\_Model\_Training.ipynb*: Model training using machine learning algorithms.
   * *03\_Model\_Evaluation.ipynb*: Evaluation of model performance and validation.
3. **Models/**: Stores trained machine learning models for IPL score prediction.
   * *random\_forest\_model.pkl*: Trained Random Forest model.
   * *neural\_network\_model.h5*: Trained Neural Network model.
4. **Scripts/**: Contains Python scripts for reusable functions and utilities.
   * *data\_preprocessing.py*: Functions for data preprocessing tasks.
   * *model\_evaluation.py*: Functions for model evaluation and performance metrics.
5. **Reports/**: Stores generated reports, visualizations, and documentation.
   * *AIML\_Report\_IPL\_Score\_Prediction.pdf*: Final report documenting methodologies, results, and insights.
6. **README.md**: Provides a brief overview of the project, installation instructions, and guidelines for running the code.

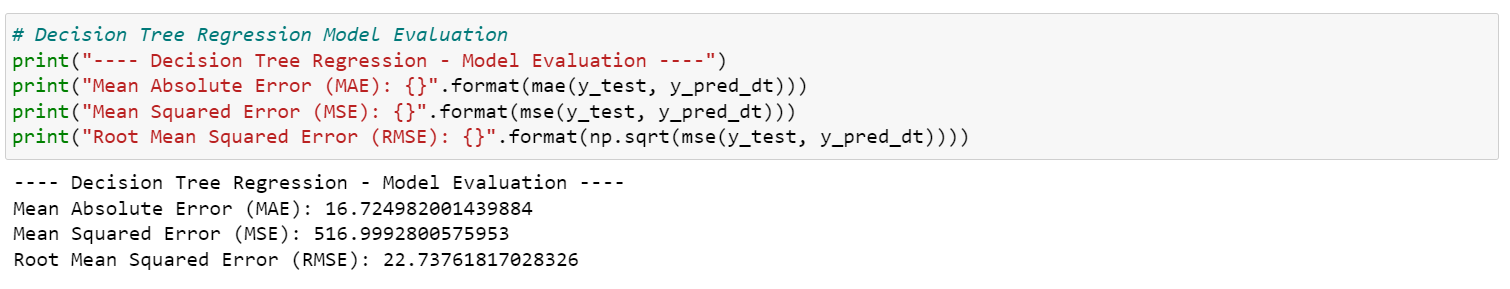
**Algorithms Used**:

The project may utilize various machine learning algorithms for IPL score prediction, including but not limited to:

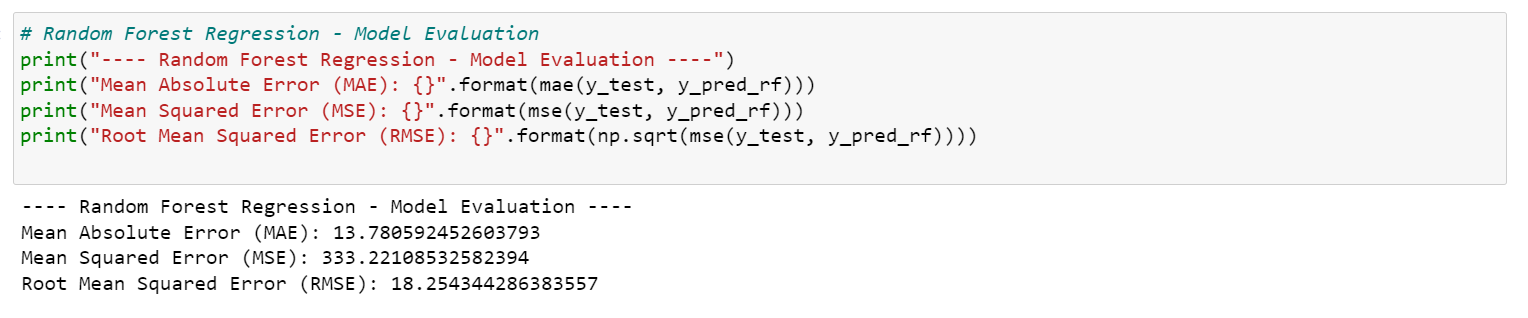
1. Random Forest: A versatile ensemble learning method capable of handling complex datasets and capturing non-linear relationships between features.
2. Gradient Boosting: Techniques such as XGBoost can be employed to build boosted tree models that excel in predictive accuracy.
3. Neural Networks: Deep learning architectures, such as feedforward neural networks or recurrent neural networks (RNNs), can capture intricate patterns in the data and offer high predictive performance.
4. Support Vector Machines (SVM): SVMs can be utilized for regression tasks to predict continuous values, making them suitable for score prediction.
5. Ensemble Methods: Combining predictions from multiple models using techniques like model averaging or stacking can often yield improved performance compared to individual models.
6. These algorithms are applied and evaluated within the project to determine their effectiveness in predicting IPL match scores accurately.

**Metrics:**

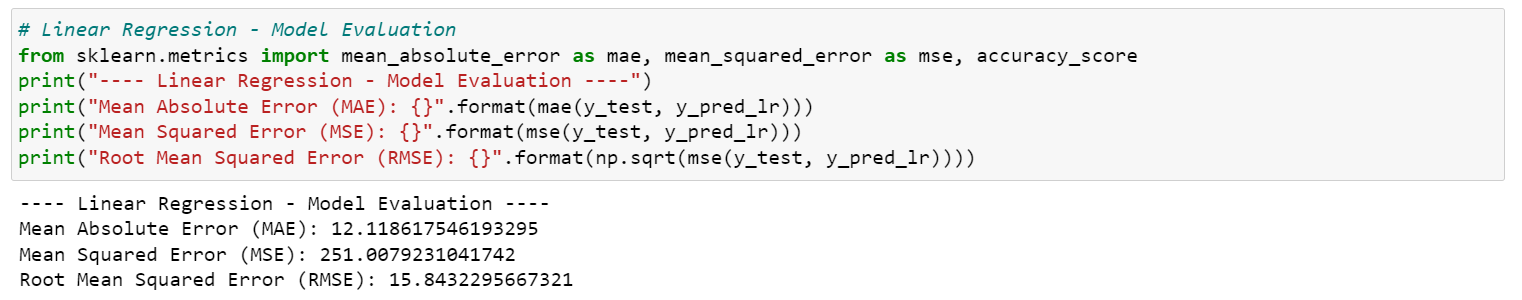
**Decision Tree:**

****

**Random Forest:**

****

Linear Regression:



**Results:**

**Predicted vs Actual Graph:**

