DATA MINING PROJECT

Analyzing Trends in IPL Over the Years

SUBMITTED BY

Ayush Yadav (21001570025)

Riyansh Sharma (21001570069)

Shrishti Rawat (21001570082)

BSc.(H) Computer Science III Year

2024

Under the guidance of

Prof. Sharanjit Kaur
Mr. Mehtab Alam

Department of Computer Science

ACHARYA NARENDRA DEV COLLEGE

ACKNOWLEDGEMENT

Apart from the efforts of the team, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project. The completion of any interdisciplinary project depends upon cooperation, coordination, and combined efforts of several sources of knowledge.

We are eternally grateful to our teachers Prof. Sharanjit Kaur and Mr. Mehtab Alam for their valuable advice and direction under which we executed this project. Their constant guidance and willingness to share their vast knowledge made us understand this project and its manifestations in great depth and helped us to complete the assigned tasks.

Ayush Yadav
(21001570025)

Riyansh Sharma
(21001570069)

Shrishti Rawat
(21001570082)

ACHARYA NARENDRA DEV COLLEGE

(University of Delhi)

CERTIFICATE

This is to certify that the project entitled "Analyzing Trends in IPL Over the Years" has been done by Ayush Yadav, Riyansh Sharma, and Shrishti Rawat of Bachelor of Computer Science (Hons.) as a part of their Semester project from Acharya Narendra Dev College, University of Delhi under the supervision of Mr. Mehtab Alam

Ayush Yadav Riyansh Sharma Shrishti Rawat

Supervisor
Mr. Mehtab Alam

Contents

Chapter 1

PROBLEM STATEMENT

The Indian Premier League (IPL) stands as a pinnacle of sports entertainment, showcasing the best of cricket talent, captivating audiences worldwide, and driving significant commercial interests. Despite its immense popularity and widespread acclaim, the IPL harbors a considerable untapped resource: the wealth of data generated throughout each season. This data encompasses player performances, match outcomes, fan engagement metrics, and various other dimensions that hold the potential to unveil critical insights for stakeholders across the IPL ecosystem.

However, the challenge lies in the underutilization of this abundant data. Despite its availability, many aspects of IPL data remain largely unexplored, limiting the league's ability to leverage data-driven decision-making and capitalize on emerging trends. This underutilization presents a significant gap, hindering teams, sponsors, broadcasters, and other stakeholders from fully maximizing the value derived from IPL participation.

Recognizing this gap, our project aims to address the challenge of IPL data underutilization by employing advanced data mining techniques to extract actionable insights from IPL data. Our primary objective is to bridge the chasm between IPL data abundance and its practical application, thereby empowering stakeholders with the information needed to make informed decisions and gain a competitive edge in the dynamic IPL landscape.

To achieve this, our project focuses on several key areas of analysis. Firstly, we seek to identify the optimal "dream team" based on player performance metrics, aiding teams in player auctions and team selection processes. Furthermore, we endeavor to quantify the influence of players based on their on-field contributions, enabling teams to assess player value more accurately. Evaluating the accuracy of player valuations and uncovering competitive trends within IPL data also form integral components of our analysis.

By comprehensively analyzing IPL data and presenting actionable insights, our project aims to empower stakeholders to make informed decisions, enhance team performance, optimize investments, and ultimately elevate the overall IPL experience for fans, players, teams, sponsors, and broadcasters alike. Through our efforts, we aspire to unlock the full potential of IPL data, transforming it from an underutilized resource into a strategic asset driving success across the IPL ecosystem.

Chapter 2

DATA MINING TECHNIQUES

Data mining [5][3][2] techniques encompass a collection of methods and algorithms devised to extract patterns, insights, and knowledge from extensive datasets. These techniques are geared towards revealing concealed relationships, trends, and valuable information that might elude human observation at first glance. By delving into large datasets, data mining endeavors to unveil nuanced connections and uncover actionable insights that can inform decision-making processes and drive innovation across various domains.

2.1 DM Techniques

2.1.1 Classification

Classification[5][3][2] is a supervised learning technique used to assign labels or categories to data instances based on their features. In a classification task, a model is trained on a labeled dataset, where each data instance is associated with a known class or category. The trained model can then predict the class label of new, unseen data instances based on their features. Common classification algorithms include decision trees, logistic regression, and neural networks. These algorithms learn decision boundaries or rules from the training data, allowing them to classify new instances into predefined categories. Classification finds applications in a wide range of tasks, including spam detection, sentiment analysis, disease diagnosis, and credit risk assessment

2.1.2 Association Rule Mining

Association rule mining [5][3][2] is a technique used to discover interesting relationships or patterns between variables in large datasets. It aims to identify associations or correlations between items based on their occurrence together. This technique is particularly popular in market basket analysis, where it helps uncover purchasing patterns and associations among products. The process of association rule mining involves two main metrics: support and confidence.By setting thresholds for these metrics, analysts can extract meaningful associations that meet certain criteria.For example, in a retail setting, association rule mining might reveal that customers who purchase bread are also likely to buy butter.

2.1.3 Clustering

Clustering [5][3][2] is a data mining technique used to group similar data points together based on their characteristics or features. The goal of clustering is to identify natural groupings or clusters within the data, where data points within the same cluster are more similar to each other compared to those in other clusters. There are various clustering algorithms, such as K-means, hierarchical clustering, and DBSCAN, each with its own approach to partitioning the data into clusters. These algorithms typically rely on distance or similarity measures to determine the proximity of data points and allocate them to clusters accordingly. Clustering finds applications in diverse fields, including customer segmentation, image segmentation, anomaly detection, and document clustering

2.2 Data mining technique used for this project

2.2.1 K-Means Clustering

For our project, we employed the K-means clustering [5][3][2] algorithm as the primary data mining technique. K-means clustering is a widely-used unsupervised learning method that partitions data into distinct clusters based on similarities in feature space. We applied this technique to our IPL dataset to identify separate clusters representing batsmen and bowlers. This approach facilitated the determination of the concept of a Dream Team, a crucial aspect of our analysis.

Initially, we selected pertinent performance metrics for both batsmen and bowlers, including batting average, strike rate, bowling average, economy rate, and wickets taken. Preceding clustering, we standardized or normalized these metrics to ensure uniformity in scale, which is pivotal for the efficacy of K-means clustering. Subsequently, we applied the K-means algorithm separately to the batsmen and bowlers data, specifying the number of clusters (K) based on domain knowledge and insights gleaned from preliminary data exploration.

Following clustering, we interpreted the resulting clusters to discern common traits or performance patterns among batsmen and bowlers within each cluster. Utilizing the clustered data, we formulated the concept of a Dream Team by selecting top-performing batsmen and bowlers from clusters representing diverse playing styles and strengths. Finally, we evaluated the effectiveness of the Dream Team formation based on predefined criteria, enabling refinement and optimization of the team lineup if necessary. Through the application of K-means clustering, we were able to extract meaningful insights from the IPL dataset, aiding in the selection and composition of an optimal Dream Team lineup that encompasses a wide range of player profiles and playing styles.

2.2.2 Regression Analysis

In our project, regression [5][3][2] played a pivotal role as a fundamental data mining technique utilized to anticipate the prices of IPL players, encompassing both batsmen and bowlers, based on pertinent attributes. Through the application of regression analysis, our objective was to unveil the intrinsic correlations between player performance metrics and their corresponding market valuations, thus facilitating the accurate estimation of player prices within the context of IPL auctions. Employing a rigorous approach encompassing meticulous data preparation, feature engineering, and model training, we successfully crafted robust regression models adept at precisely forecasting player prices. These models offer invaluable insights for stakeholders engaged in player acquisitions, auction strategies, and team composition decisions, empowering them with actionable information derived from the IPL dataset. Leveraging regression as a data mining technique has empowered stakeholders with the requisite knowledge to make informed decisions, thereby positioning them to gain a competitive advantage in navigating the dynamic landscape of IPL auctions and player valuations.

Chapter 3

DATASET DESCRIPTION

In this project, we used two different datasets namely ipl-match-data[1] and ipl-player-data[4] from Kaggle, espncricinfo and verified the data from the official IPL website.

3.1 Number of Records

In ipl-match-data dataset[4], we have 243817 records which contain ball by ball data of every match in IPL played by each player from 2008 to 2023.

In ipl-player-data dataset, we have 237 records that contain the current value of 237 players played till now in the IPL.

3.2 Number of Attributes

In ipl-match-data dataset, we have 23 attributes namely match-id, season, start-date, venue, innings, ball, batting-team, bowling-team, striker, non-striker, bowler, runs-off-bat, extras, wides, noballs, byes, legbyes, penalty, wicket-type', 'player-dismissed', 'other-wicket-type', 'other-player-dismissed', 'cricsheet-id'.

In ipl-player-data dataset, we have 3 attributes namely name, age, valueinCR.

3.3 Types of Attributes

In ipl-match-data and ipl-player-data, we have types of attributes - Numeric and Categorical.

3.3.1 Numeric Attributes

- **match-id**: Numeric or alphanumeric identifier.
- **ball**: Numeric, indicating the ball number within an innings.
- **runs-off-bat**: Numeric, indicating the runs scored by the batsman without extras.
- **extras**: Numeric, indicating extra runs scored (e.g., wides, no-balls, etc.).
- **wides**: Numeric, indicating the number of wides bowled.
- **noballs**: Numeric, indicating the number of no-balls bowled.
- **byes**: Numeric, indicating runs scored due to byes.
- **legbyes**: Numeric, indicating runs scored due to leg-byes.
- **penalty**: Numeric, indicating penalty runs awarded.
- **age**: Numeric, indicating the age of the player.
- **valueinCR**: Numeric, indicating the current value of the player in some unit (possibly in crore rupees).

3.3.2 Categorical Attributes

- **season**: Categorical (e.g., 2008, 2009, etc.).
- **venue**: Categorical, textual data.
- **innings**: Numeric or categorical (e.g., 1st innings, 2nd innings, etc.).
- **batting-team**: Categorical, textual data.
- **bowling-team**: Categorical, textual data.
- **striker**: Categorical, textual data.
- **non-striker**: Categorical, textual data.
- **bowler**: Categorical, textual data.
- **wicket-type**: Categorical, textual data indicating the type of wicket.

- **player-dismissed**: Categorical, textual data indicating the dismissed player.
- **other-wicket-type**: Categorical, textual data indicating additional types of wickets.
- **other-player-dismissed**: Categorical, textual data indicating additional dismissed players.
- **name**: The name of the player.
- **cricsheet-id**: Numeric or alphanumeric identifier.

3.4 Missing values or nulls

Some of the numerical attributes in ipl-match-data has missing values as only a few of the attributes can to a ball played in a match but we carefully handled all these in preprocessing.

3.5 Attributes Description

In ipl-match-data dataset:

- **match-id**: A unique identifier for each match.
- **season**: The season in which the match took place (e.g., 2008, 2009, etc.).
- **start-date**: The date on which the match started.
- **venue**: The location where the match was played.
- **innings**: The innings number (1st innings, 2nd innings, etc.).
- **ball**: The ball number within the innings.
- **batting-team**: The team batting during the particular innings.
- **bowling-team**: The team bowling during the particular innings.
- **striker**: The batsman facing the delivery.
- **non-striker**: The batsman at the non-striker's end.
- **bowler**: The player bowling the delivery.

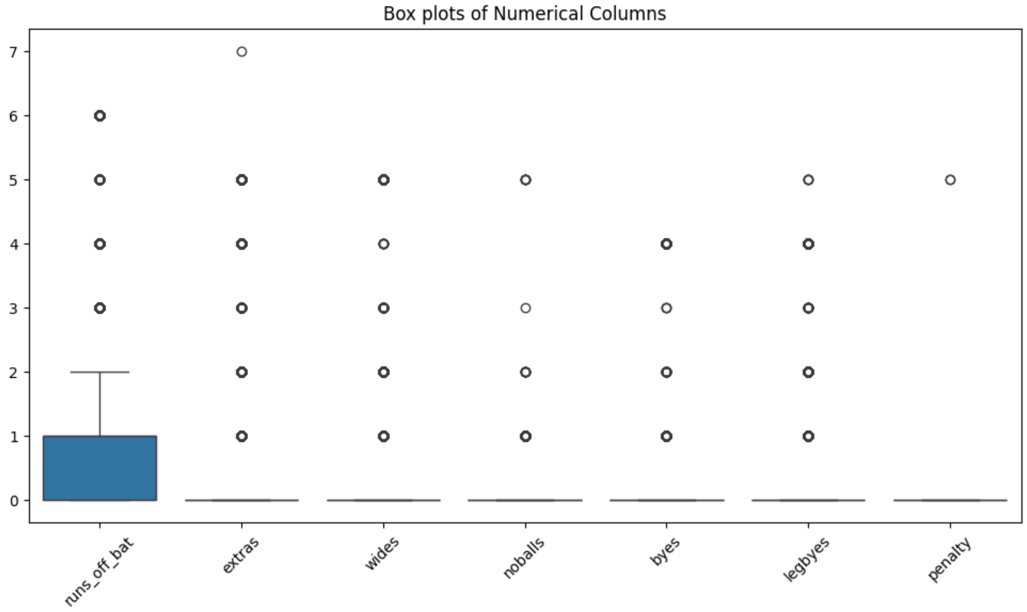
- **runs-off-bat**: The number of runs scored by the batsman without any extras (like wides, no-balls, etc.).
- **extras**: Additional runs scored due to wides, no-balls, etc.
- **wides**: Number of wides bowled in the delivery.
- **noballs**: Number of no-balls bowled in the delivery.
- **byes**: Runs scored due to the ball passing the batsman without being hit and not being intercepted by the wicketkeeper.
- **legbyes**: Runs scored after the ball hits the batsman's body or protective gear and then goes away from the fielders.
- **penalty**: Penalty runs awarded.
- **wicket-type**: Type of wicket taken (e.g., caught, bowled, lbw, etc.).
- **player-dismissed**: The batsman who got out.
- **other-wicket-type**: Additional type of wicket if it's a unique circumstance.
- **other-player-dismissed**: Additional batsman dismissed in unique circumstances.
- **cricsheet-id**: A unique identifier for each entry in the cricket match data.

In ipl-player-data dataset:

- **name**: The name of the player.
- **age**: The age of the player.
- **valueinCR**: The current value of the player, possibly in crore rupees.

3.6 Detecting Outliers

We had very few outliers in our dataset as shown below



Chapter 4

DATA PREPROCESSING

This section outlines the preprocessing steps undertaken to clean and prepare the dataset for further analysis. It describes how missing values were handled, irrelevant columns were dropped, and player statistics were calculated and consolidated. These preprocessing steps ensure the dataset is suitable for subsequent analysis and modeling tasks.

4.1 Data cleaning and Feature Engineering

4.1.1 Ipl-match-data Cleaning

1. **Columns Selection**: Initially, relevant data is selected from the ipl-match-data and added in deliveries dataframe for further analysis.

2. **Handling Missing Values**: NaN values in specified columns (`runs_off_bat`, `wides`, `noballs`, `byes`, `legbyes`, `penalty`) are replaced with integer 0.

3. **Calculating Total Runs**: The `total_runs` column is calculated by summing up the specified columns.

4. **Restructuring DataFrame**: The `ball` column is split into `over` and `ball` columns, and the DataFrame is reordered for better readability and analysis.

5. **Team Names Standardization**: Inconsistent team names (*Rising Pune Supergiant*, *Delhi Daredevils*, *Kings XI Punjab*) are standardized to match other team names in the dataset.

6. **Venue Names Cleaning**: Venue names are cleaned by removing location information after comma (if present) and renaming specific venues

(*Sardar Patel Stadium* to *Narendra Modi Stadium*, *Feroz Shah Kotla* to *Arun Jaitley Stadium*) for consistency.

4.1.2 Player Statistics Calculation

1. **Batter Statistics Calculation**: Various batting statistics such as Strike Rate (SR), Runs per Inning (RPI), Balls per Dismissal (BPD), and Balls per Boundary (BPB) are calculated for each player based on the deliveries data.

2. **Bowler Statistics Calculation**: Bowling statistics including overs, economy rate, and strike rate are computed for each bowler.

4.1.3 Consolidation of Player Data

1. **Combining Batting and Bowling Statistics**: Batting and bowling statistics are combined into a single DataFrame to create a comprehensive player profile.

2. **Data Aggregation**: Player statistics are aggregated based on batting and bowling performances for a consolidated view.

4.2 Data Sampling and Subsetting

In this section, we describe the process of sampling and subsetting the dataset for model training. The dataset contains information about players in the Indian Premier League (IPL), including various performance metrics such as runs scored, batting average, batting strike rate, wickets taken, bowling average, and economy rate.

4.2.1 Filtering Data

Initially, we filtered the dataset to include only players categorized as batsmen and all-roundersin one dataframe and bowlers in another dataframe. This step was essential as it allowed us to focus on specific player types that contribute significantly to batting or bowling performance. We achieved this filtering by selecting rows where the player type was either ”Batsman” or ”All-Rounder” or ”Bowlers”. By doing so, we ensured that the datasets were tailored to the specific characteristics of these player types.

4.2.2 Feature Selection

After filtering the data, we selected relevant features for each player type based on their performance metrics. For batsmen and all-rounders, we chose features such as `MatchPlayed`, `RunsScored`, `BattingAVG` (batting average), and `BattingS/R` (batting strike rate). These features are indicative of a player's batting performance and are commonly used to assess their contribution to the team.

Similarly, for bowlers, we selected features such as `MatchPlayed`, `Wickets`, `BowlingAVG` (bowling average), and `EconomyRate`. These features provide insights into a bowler's effectiveness in taking wickets and controlling the run rate, which are crucial aspects of bowling performance in cricket.

4.2.3 Train-Test Split

With the filtered and feature-selected dataset, we proceeded to split the data into training and testing sets. The purpose of this step was to train our machine learning models on a subset of the data and evaluate their performance on a separate subset. We used an 80-20 split ratio, where 80% of the data was allocated to the training set and 20% to the testing set.

The splitting was performed randomly to ensure that the training and testing sets were representative of the overall dataset. By splitting the data in this manner, we could train our models on a sufficient amount of data while also having a separate set for unbiased evaluation.

Chapter 5

BUILDING MODEL

In our endeavor to predict the values of IPL players for the next season, we opted to utilize the Random Forest Regressor as our primary predictive model. This decision was motivated by the versatility and robustness of the Random Forest algorithm, which is well-suited for handling complex datasets with numerous features and non-linear relationships. Additionally, Random Forests are less prone to overfitting compared to traditional regression models, making them an ideal choice for our predictive modeling task.

5.1 Model for Batsman/All-Rounders

In crafting the predictive model for batsmen and all-rounders, our strategy revolved around the meticulous selection of attributes crucial for capturing the nuanced performance dynamics inherent in IPL cricket. We carefully curated a comprehensive set of attributes, including batting average, strike rate, number of centuries, number of half-centuries, boundary percentage, and contextual variables such as recent form and match conditions. These attributes were chosen to encapsulate the multifaceted contributions of batsmen and all-rounders to team success, ensuring a holistic representation of their skill sets and impact on the game.

5.2 Model for Bowlers

For the predictive model targeting bowlers, our approach entailed the integration of key performance metrics and contextual factors essential for evaluating bowling effectiveness within the IPL framework. We selected a tailored set of attributes, comprising bowling average, economy rate, wickets taken. These attributes were carefully chosen to encompass the diverse facets of bowling prowess, including efficiency, wicket-taking ability, and adaptability across varying match conditions. By incorporating these attributes, we aimed to construct a comprehensive model capable of accurately assessing the value and impact of bowlers in the upcoming IPL season.

Chapter 6

MODEL EVALUATION AND RESULTS

In data mining, model evaluation and comparison are essential processes for assessing the performance of different models used in the analysis. These processes involve testing models against specific criteria, such as accuracy, precision, recall, and F1-score, to determine their effectiveness in solving the given problem. Techniques like cross-validation and statistical tests are employed to ensure the reliability of the evaluation results. The comparison of models helps in identifying the most suitable model for the dataset and problem statement, considering factors like scalability and interpretability. Ultimately, model evaluation and comparison play a crucial role in guiding decision-making regarding model selection and deployment in real-world scenarios.

6.1 Metrics

Data mining metrics serve as invaluable tools for assessing the quality and performance of models, algorithms, and processes employed in data mining tasks. These quantitative measures offer a systematic means of evaluating the effectiveness, accuracy, and efficiency of data mining techniques.In our analysis, we employed a Random Forest Regression model to predict the prices of IPL players for the upcoming season. To evaluate the performance of the model, we utilized several key metrics commonly employed in regression analysis. The Mean Absolute Error (MAE),the Mean Squared Error (MSE), The Root Mean Squared Error (RMSE).

6.1.1 Mean Absolute Error

The Mean Absolute Error (MAE) is a metric commonly used in regression analysis to quantify the average absolute difference between the predicted values and the actual values. It provides a straightforward measure of prediction accuracy, with lower MAE values indicating better model performance.

In our analysis, we also evaluated the Mean Absolute Error (MAE) for our models to further assess their performance in predicting player performance in the IPL. The Batsman All-Rounder Model exhibited an MAE of 1.24 for the training set and 2.15 for the testing set. Similarly, the Bowler Model showed an MAE of 1.15 for training and 1.87 for testing. These MAE values suggest that our models have a relatively low level of error in predicting player performance, indicating their effectiveness in capturing the underlying patterns in the data.

6.1.2 Mean Squared Error

The Mean Squared Error (MSE) is a widely used metric in regression analysis to measure the average squared difference between the predicted values and the actual values. It provides a measure of the average magnitude of errors made by the model, with higher MSE values indicating larger discrepancies between predicted and actual values.

For the Batsman and All-Rounder Model, the MSE values are approximately 2.84 (Train) and 10.75 (Test). These values indicate that, on average, the squared difference between predicted and actual values is relatively low, suggesting that the model is performing well in predicting player performance for batsmen and all-rounders.

Similarly, for the Bowler Model, the MSE values are approximately 2.48 (Train) and 6.23 (Test). These values also indicate that the model is performing well in predicting player performance for bowlers in the context of the IPL.

Overall, the MSE values suggest that our models are effective in capturing the underlying patterns and trends in the data, leading to accurate predictions of player performance.

6.1.3 Root Mean Squared Error

The Root Mean Squared Error (RMSE) is a widely used metric in regression analysis to measure the average magnitude of errors made by a model. It is calculated as the square root of the Mean Squared Error (MSE), providing a measure of the average deviation between the predicted values and the actual values, expressed in the same units as the target variable. RMSE is particularly useful for understanding the scale of errors relative to the target variable, with lower RMSE values indicating better model performance.

In our analysis of IPL player performance using clustering and regression techniques, we achieved notable results in terms of Root Mean Square Error (RMSE) for our models. For the Batsman All-Rounder Model, the RMSE values were 1.686 (Train) and 3.276 (Test), indicating a relatively low level of error in predicting player performance. Similarly, the Bowler Model exhibited strong predictive capabilities, with RMSE values of 1.574 (Train) and 2.495 (Test). These results suggest that our models are effective in accurately predicting player performance, particularly for batsmen, all-rounders, and bowlers in the context of the IPL.

6.2 Experimental Results and Comparison

In our project analyzing trends in the Indian Premier League (IPL) using clustering and regression techniques, we have derived several significant findings. Through clustering, we identified the best potential team from a comprehensive dataset encompassing all players who have participated in the IPL to date. Our analysis aimed to predict the performance of this theoretical team in a tournament scenario, suggesting its potential to emerge as a formidable contender for the IPL title.

Moreover, our regression analysis enabled us to forecast impact players for future IPL seasons. These predictions are crucial for team management and selection strategies, offering insights into players who are likely to significantly influence match outcomes. By identifying impact players in advance, teams can tailor their strategies to leverage the strengths of these players, potentially gaining a competitive edge in the tournament.

Furthermore, our analysis unveiled instances of both overestimated and underestimated players in the IPL. By quantitatively evaluating player performances against expectations, we were able to pinpoint players who have either consistently outperformed or underperformed relative to their perceived value. This insight is valuable for team management, providing an evidence-based approach to player evaluation and recruitment, which can lead to more informed decision-making processes.

Overall, our project's experimental results underscore the efficacy of clustering and regression techniques in analyzing IPL data. The findings not only offer strategic advantages for team management and selection but also contribute to the broader understanding of player performances and team dynamics in the context of one of the world's premier cricket tournaments.

Chapter 7

INFERENCES AND CONCLUSION

Our analysis of IPL trends, employing sophisticated clustering and regression methodologies, has provided insightful findings. Our models successfully predicted the optimal team composition from the extensive pool of IPL players, offering valuable insights into the ideal player mix to enhance team performance significantly. This finding has profound implications for team management, guiding decisions on player selection and team formation strategies to maximize competitiveness. Additionally, our regression analysis yielded a compelling prediction: the probability of the identified best team winning the tournament if played together. This prediction serves as a tangible metric for evaluating team strength and potential success in the IPL.

Furthermore, our study identified impact players—those with the most substantial influence on match outcomes. This insight can inform future team strategies, focusing efforts on recruiting or retaining such influential players to bolster overall team performance. Moreover, our analysis highlighted players who may have been either overestimated or underestimated, providing teams with crucial insights to adjust their strategies and make more informed decisions regarding player selection and team composition.

In summary, our comprehensive analysis of IPL trends, leveraging advanced clustering and regression methodologies, has revealed valuable insights into team dynamics, player efficacy, and strategic nuances. These findings have the potential to significantly benefit IPL franchises, equipping them with the strategic foresight needed to excel in future tournaments and raise the overall standard of play in the league.

Bibliography

[1] Ipl dataset from 2008-23. . Accessed: February 30, 2024.

[2] Pei J. Kamber M. Han, J. *Data mining: concepts and techniques. 3rd Edition.* Accessed : Feburary 5, 2024.

[3] Sukumaran S. Kesavaraj, G. *A study on classification techniques in data mining.* Accessed : January 29, 2024, In 2013 fourth international conference on computing, communications and networking technologies (ICCCNT) (pp. 1-7).

[4] Board of Control for Cricket in India (BCCI). Indian premier league (ipl) dataset. https://www.iplt20.com/stats/2024. Accessed: February 24, 2024.

[5] Steinbach M. Kumar V. Tan, P. N. *Introduction to Data Mining (2016).* Accessed : January 30, 2024.