IPL MATCH WIN PREDICTOR USING ML

NPTEL PROJECT

submitted to the Savitribai Phule Pune University, Pune
In the partial fulfilment of the requirements
for the award of the degree

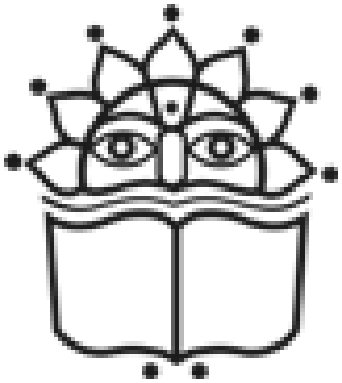
BACHELOR OF ENGINEERING
(AI&DS Engineering)

BY

YASHRAJ DEVRAT
Roll No: B190352011

Under the guidance of

Prof.Pradip Shendage
Assistant Professor



DEPARTMENT OF AI&DS ENGINEERING
Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering
and Technology,
Vidyanagari Bhigawan Road
Baramati- 413133

2022-23

CERTIFICATE

This is to certify that **Mr. Yashraj Devrat** has successfully submitted her project. report to the Department of AI&DS Engineering, VPKBIET, Baramati,on

"IPL MATCH WIN PREDICTOR USING ML"

during the academic year 2023-2024 in the partial fulfilment towards completion of Fourth Year of **Bachelor of Engineering** in **Artificial Intelligence & Data Science** .

Mr.Pradip Shendage
Assistant Professor,
Guide,
Dept of AI&DS Engineering.

Mr P.M.Paithane
Assistant Professor,
HOD,
Dept of AI&DS Engineering.

Dr. R.S.Bichkar
Principal
VPKBIET, Baramati.

Place:Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering and Technology, Baramati.
Date : _____

Abstract

The Indian Premier League (IPL) has emerged as one of the most popular and competitive cricket tournaments worldwide. With its fast-paced and unpredictable nature, accurately predicting the outcome of IPL matches has become a significant challenge. This abstract presents a data science project aimed at developing a model for IPL match win prediction, leveraging historical match data and advanced analytics techniques. The project begins by collecting a dataset consisting of various features, including team performance metrics, player statistics, venue conditions, and historical match results. These features are carefully selected based on their relevance to match outcomes and are preprocessed to ensure consistency and quality.

To evaluate the performance of the developed models, the dataset is divided into training and testing subsets using cross-validation techniques. Various evaluation metrics such as accuracy, precision, recall, and F1-score are utilized to measure the predictive performance of each model.

Keywords: Machine Learning, Classification, Support Vector Machines, Logistic Regression, Random Forest Classifier.

Contents

List of Figures

Chapter 1

Introduction

1.1 Introduction

The Indian Premier League (IPL) has revolutionized the world of cricket, captivating millions of fans with its exhilarating matches and star-studded teams. With its unpredictable nature, the IPL presents an intriguing challenge for cricket enthusiasts and analysts alike: predicting the outcome of matches. Leveraging the power of data science and advanced analytics, this project aims to develop a model for IPL match win prediction, offering valuable insights into the factors influencing match outcomes and enhancing the overall cricket viewing experience.

In recent years, data science has emerged as a powerful tool for extracting meaningful patterns and insights from vast amounts of data. By analyzing historical match data, team performance metrics, player statistics, venue conditions, and other relevant features, it becomes possible to uncover hidden relationships and trends that can aid in predicting future match results.

Ultimately, the project aims to deploy the best-performing model into a user-friendly application or web-based interface, making IPL match win predictions accessible to a wider audience. By providing real-time predictions and insights, the project seeks to enhance the excitement and engagement surrounding IPL matches.

Chapter 2

Motivation

The motivation behind this IPL match win prediction project in data science stems from the inherent excitement and unpredictability of the Indian Premier League. The IPL has captured the imagination of cricket fans worldwide, bringing together top players from different countries and creating a highly competitive and electrifying tournament.

However, the unpredictable nature of the IPL also presents a challenge. Fans, analysts, and team management are often left speculating about the outcome of matches, making it difficult to make informed decisions or engage in meaningful discussions.

The motivation behind this project is to leverage the power of data science and advanced analytics to provide a solution to this challenge. By developing a predictive model for IPL match win prediction, we aim to bring a data-driven approach to the table, enabling more accurate predictions and deeper insights into the factors influencing match outcomes.

Moreover, the project seeks to showcase the power of data science and its application in the realm of sports. By demonstrating the ability to extract meaningful patterns and insights from historical match data, it highlights the broader potential of data science in sports analytics and decision-making.

Motivation behind this IPL match win prediction project is to leverage data science and advanced analytics to enhance the cricket viewing experience, provide valuable insights into match outcomes, assist team management in decision-making, and showcase the potential of data science in the sports domain.

Chapter 3

System Architecture

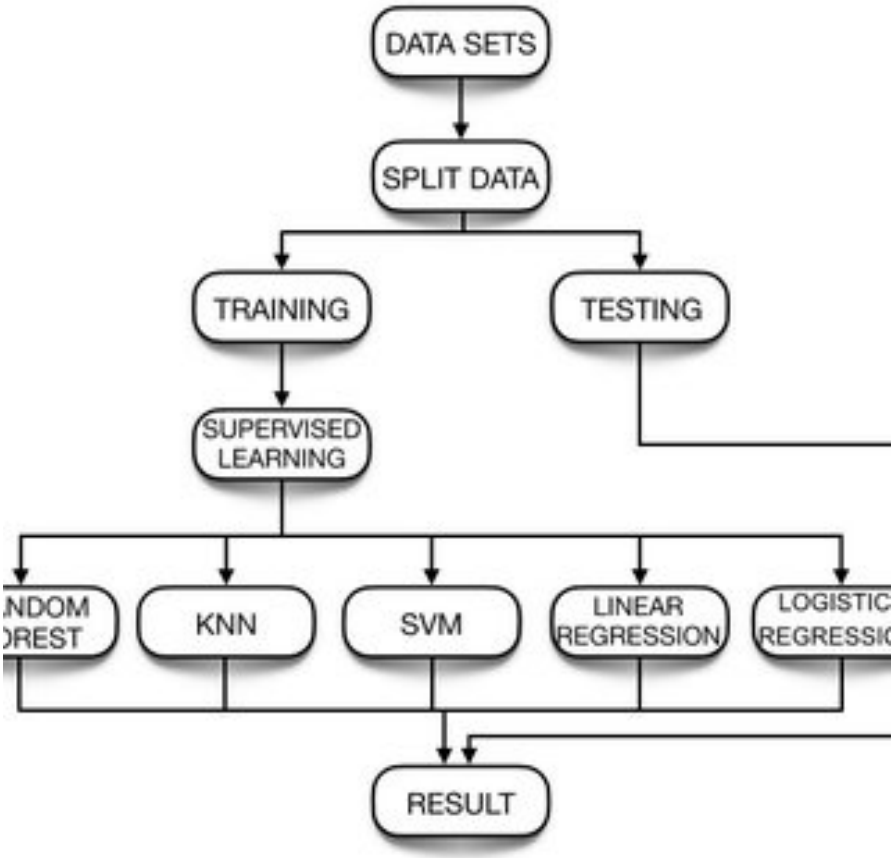


Figure 3.1: Proposed System architecture

Chapter 4

Approaches for Solving the Problem

There are several approaches that can be employed to solve the problem of IPL match win prediction using data science. Here are a few commonly used approaches:

Supervised Learning: Supervised learning involves training a model on labeled historical data, where the input features are used to predict the target variable (match outcome). Various machine learning algorithms such as decision trees, random forests, support vector machines, and gradient boosting algorithms can be applied to build the predictive model.

Time Series Analysis: Time series analysis takes into account the temporal aspect of the IPL matches. It involves analyzing the historical match data in chronological order to identify patterns, trends, and seasonality. Time series models like ARIMA (AutoRegressive Integrated Moving Average) or SARIMA (Seasonal ARIMA) can be used to forecast match outcomes based on past performance.

Ensemble Methods: Ensemble methods combine multiple models to make predictions. Techniques like bagging (e.g., Random Forests) and boosting (e.g., Gradient Boosting Machines) can be employed to build an ensemble of models that collectively provide more accurate predictions. Ensemble methods can help mitigate overfitting and improve the robustness of the predictive mode.

Deep Learning: Deep learning techniques, particularly neural networks, can be utilized for IPL match win prediction. Deep neural networks can learn complex patterns and relationships from the data, potentially capturing intricate factors that influence match outcomes. Architectures like recurrent neural networks (RNNs) and long short-term memory (LSTM) networks can be employed to model the sequential nature of match data.

Chapter 5

Comparison of the Approaches

In comparing Support Vector Machines (SVM), Logistic Regression, and Random Forest Classifier (RFC) for IPL Match Win Predictor, distinct characteristics emerge. SVM, while effective in high-dimensional space, might struggle with large datasets due to computational demands. Logistic Regression, though simple and interpretable, may lack complexity in capturing intricate relationships within the data. RFC, with its ensemble of decision trees, showcases robustness in handling complex data structures but could be prone to overfitting. SVM offers high accuracy but requires careful parameter tuning, while Logistic Regression provides simplicity but may not capture nuanced relationships. RFC stands out for its ability to handle complex data but necessitates careful management to prevent overfitting. Each approach presents trade-offs between complexity, interpretability, and performance, underscoring the importance of selecting the most suitable method based on the dataset characteristics and desired outcomes.

Sr.No	Algorithm	Accuracy
1	SVM	0.923977
2	RFC	0.982456
3	Logistic Regression	0.953216

Figure 5.1: Comparison of the three algorithms

Chapter 6

Advantages and Disadvantages

- Support Vector Machines (SVM):

  Advantages:

- Effective in high-dimensional spaces.
- Versatile due to various kernel options for decision functions.
- Works well with limited datasets, providing high accuracy.
- Robust against overfitting in high-dimensional spaces.
- Useful in cases where the number of dimensions is greater than the number of samples.

  Disadvantages:

- Computationally intensive, especially with large datasets.
- Prone to reduced performance with noisy data.
- Challenging to fine-tune hyperparameters, impacting performance.
- Inefficient for multi-class classification.
- Interpretability might be limited, especially with complex kernels.
- Logistic Regression:

  Advantages:

- Simple and easy to implement.
- Provides probabilities for outcomes.
- Efficient for binary classification tasks.
- Offers good interpretability of results.

- Less prone to overfitting, especially with a small number of features. Disadvantages:
- Limited capability to capture complex relationships in data.
- Assumes a linear relationship between features and the log-odds of the outcome.
- Sensitivity to outliers can affect performance.
- Not suitable for handling non-linear problems without feature transformations.
- May struggle with high-dimensional data or when there's multicollinearity.
- Random Forest Classifier (RFC):

  Advantages:
- Robust against overfitting due to ensemble learning from multiple decision trees.
- Handles high-dimensional data well.
- Effective for both classification and regression tasks.
- Provides feature importance scores.
- Less sensitive to outliers and noise in the data.

  Disadvantages:
- Complexity in interpreting individual trees within the forest.
- Requires careful tuning of hyperparameters to prevent overfitting.
- Slower in making predictions compared to some other algorithms.
- Not suitable for tasks where interpretability of the model is critical.
- Can be resource-intensive due to multiple decision trees within the ensemble.

Chapter 7

Data set

Out[3]:

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	35	0
1	2	IPL-2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Rising Pune Supergiant	0	7
2	3	IPL-2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	Kolkata Knight Riders	0	10
3	4	IPL-2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	0	6
4	5	IPL-2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	15	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
751	11347	IPL-2019	Mumbai	05-05-2019	Kolkata Knight Riders	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0	9
752	11412	IPL-2019	Chennai	07-05-2019	Chennai Super Kings	Mumbai Indians	Chennai Super Kings	bat	normal	0	Mumbai Indians	0	6
753	11413	IPL-2019	Visakhapatnam	08-05-2019	Sunrisers Hyderabad	Delhi Capitals	Delhi Capitals	field	normal	0	Delhi Capitals	0	2
754	11414	IPL-2019	Visakhapatnam	10-05-2019	Delhi Capitals	Chennai Super Kings	Chennai Super Kings	field	normal	0	Chennai Super Kings	0	6

Figure 7.1: Confusion Matrix and Performance Metrics

Chapter 8

Results

8.1 Performance Metrics

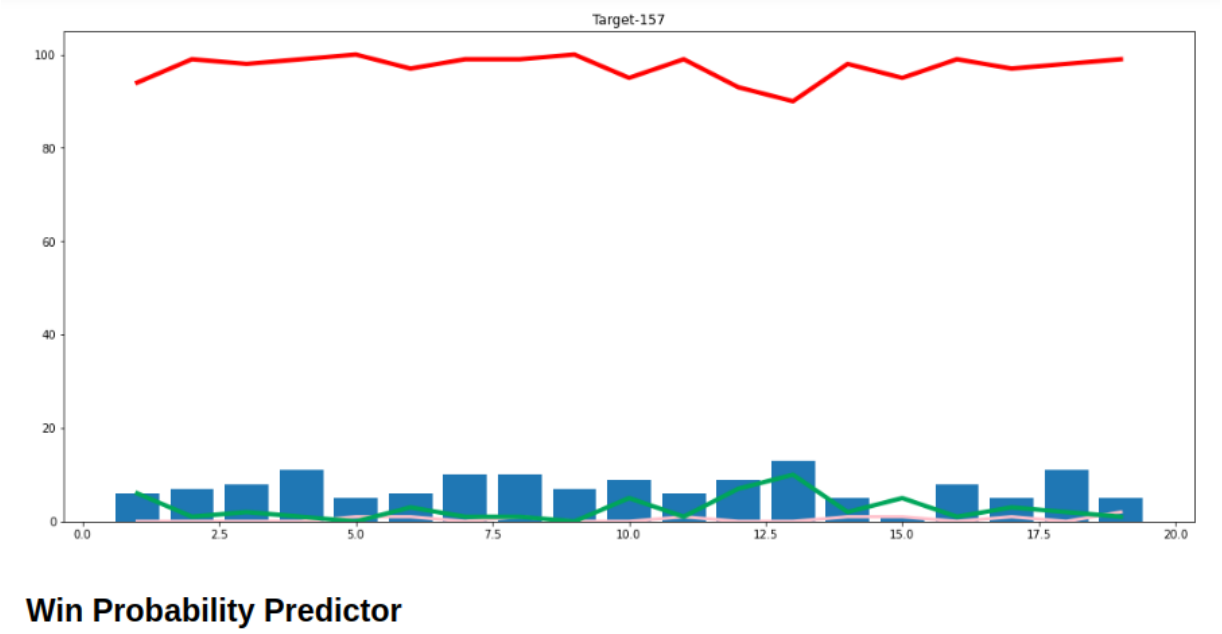


Figure 8.1: Match Score Plotting

8.2 Random Forest Classifier

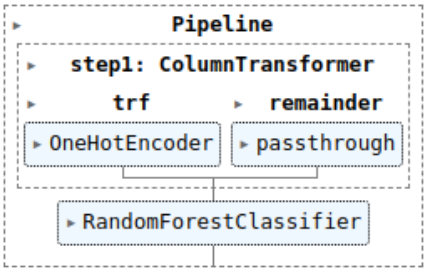
Accuracy for Random Forest Classifier

In [62]:
```
pipe = Pipeline(steps=[
    ('step1',trf),
    ('step2',RandomForestClassifier())
])
```

In [63]:
```
pipe.fit(X_train,y_train)
```

```
/home/yashraj/.local/lib/python3.10/site-packages/sklearn/preprocessing/_encoders.py:868:
as renamed to `sparse_output` in version 1.2 and will be removed in 1.4. `sparse_output`
e `sparse` to its default value.
  warnings.warn(
```

Out[63]:



In [64]:
```
y_pred = pipe.predict(X_test)
```

In [65]:
```
# RandomForest accuracy
from sklearn.metrics import accuracy_score
accuracy_score(y_test,y_pred)* 100
```

Out[65]:
```
99.92793973141173
```

Figure 8.2: Random Forest Classifier

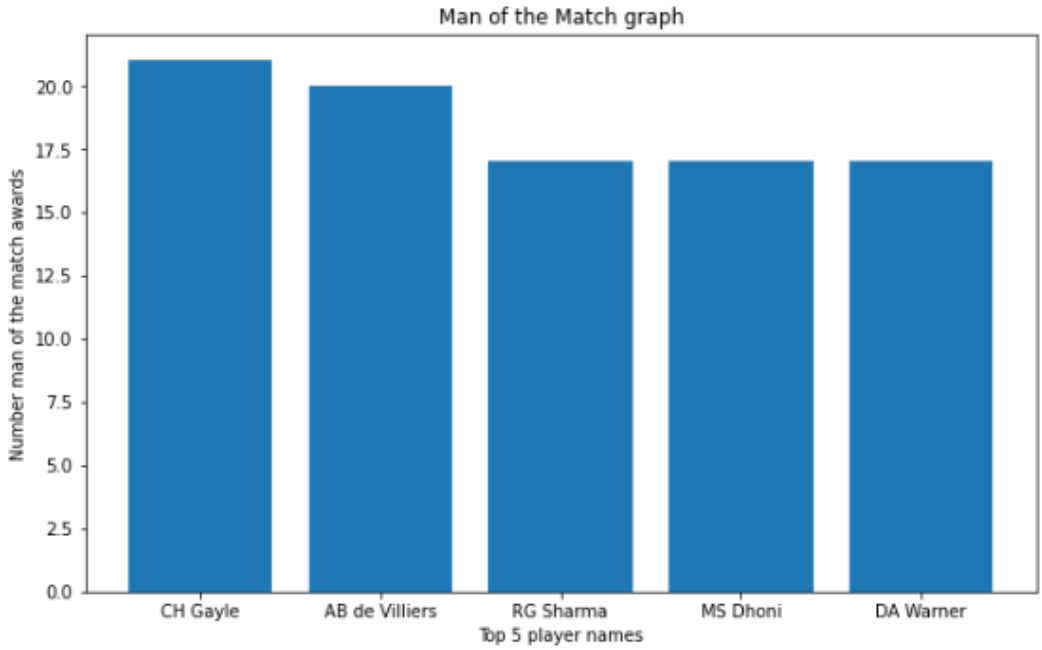
Chapter 9

Results

9.1 Data visualisation I

Barplot of top 5 player of the match

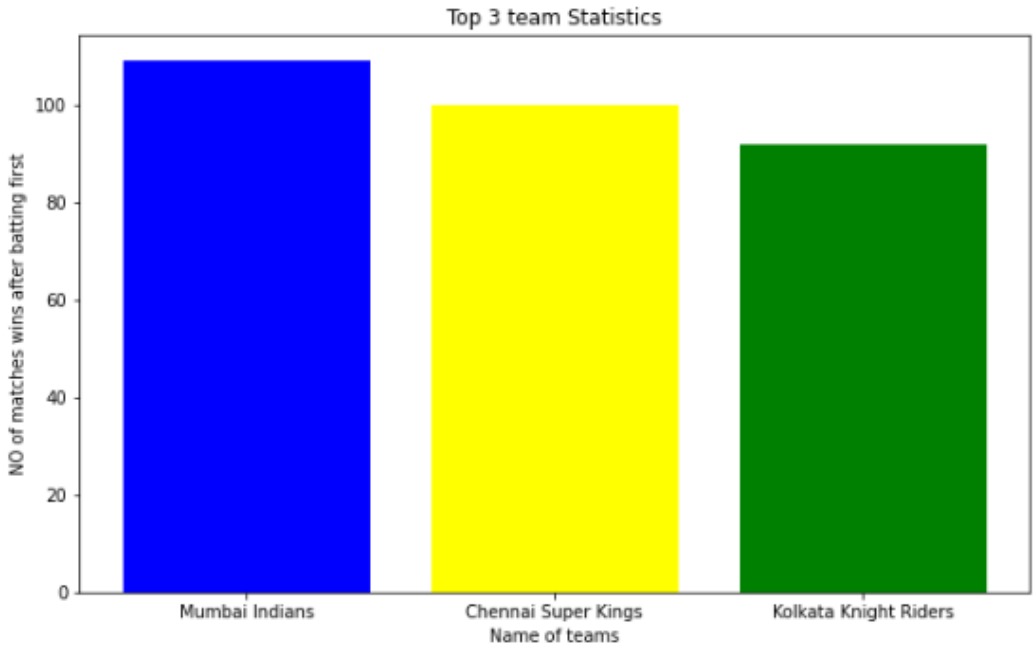
In [7]:
```
plt.figure(figsize=(10,6))
plt.bar(x,y)
plt.xlabel("Top 5 player names")
plt.ylabel("Number man of the match awards")
plt.title("Man of the Match graph")
plt.show()
```



Creating barplot for Top 3 Teamsn Wins After Batting

In [13]:

```
plt.figure(figsize=(10,6))
c = ['blue','yellow','green']
plt.bar(b,a,color=c)
plt.xlabel("Name of teams")
plt.ylabel("NO of matches wins after batting first")
plt.title("Top 3 team Statistics")
plt.show()
```



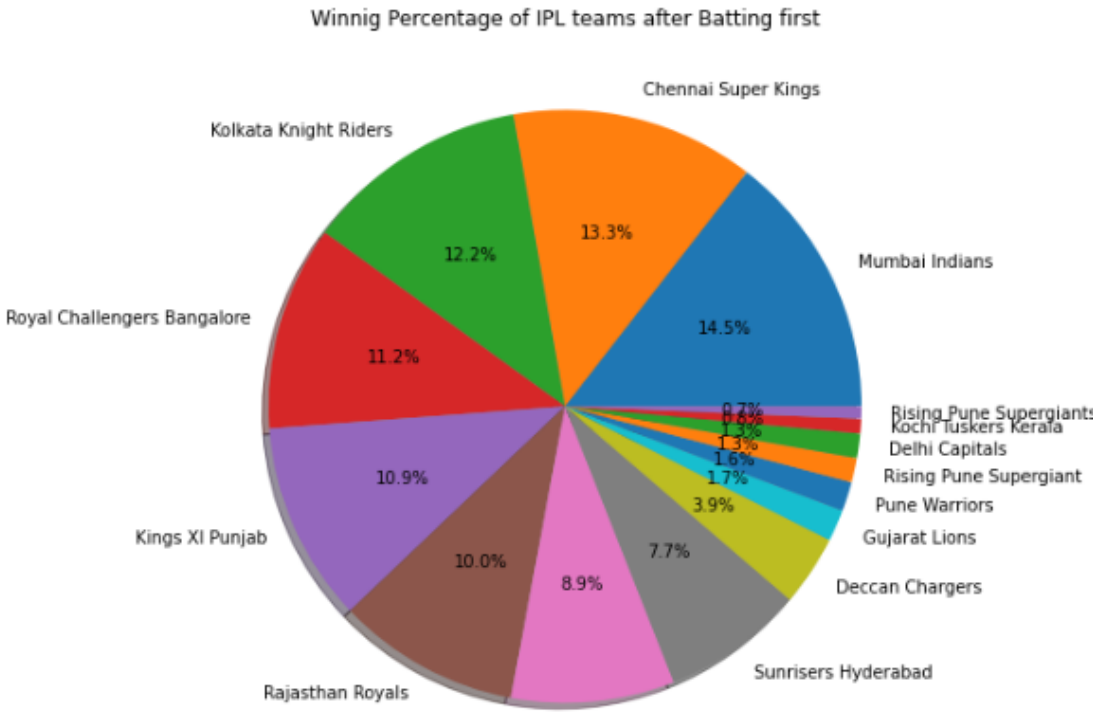
9.2 Data visualisation II

9.3 Data visualisation III

Pie Chart

In [16]:

```
plt.figure(figsize=(8,8))
plt.pie(data ,labels =Label,autopct ='%1.1f%%',shadow = True)
plt.title('Winnig Percentage of IPL teams after Batting first')
plt.show()
```



Chapter 10

Future Scope

The IPL match win prediction project in data science has several future scope opportunities that can be explored to enhance its capabilities and impact. Here are some potential areas of future development:

Incorporating Real-time Data: Currently, the project focuses on historical match data for prediction. However, integrating real-time data during matches, such as live player performance statistics, pitch conditions, and weather information, can improve the accuracy and timeliness of predictions. This would require a robust data pipeline and real-time data processing techniques.

Sentiment Analysis: Considering the impact of public sentiment and fan opinions on match outcomes can add an interesting dimension to the prediction model. Sentiment analysis of social media data, fan forums, and news articles can provide insights into the collective perception and expectations surrounding teams and players. Integrating sentiment analysis into the predictive model can help capture these intangible factors.

Player Injury Prediction: Extending the prediction capabilities beyond match outcomes, the project can explore predicting player injuries. By analyzing player workload, injury history, and other relevant factors, the model can forecast the likelihood of a player getting injured in future matches. This information can be valuable for team management in making informed decisions regarding player selection and workload management.

Advanced Feature Engineering: Continual improvement in feature engineering techniques can enhance the predictive power of the model. Exploring advanced feature selection methods, feature interaction modeling, and incorporating domain-specific knowledge can lead to the discovery of more informative features that have a direct impact on match outcomes.

Cross-Domain Prediction: Applying the developed model to other cricket leagues or even different sports tournaments can be explored. By adapting the model to different contexts, it can be used to predict match outcomes in other cricket leagues or even extend to sports like football, basketball, or baseball, provided the relevant data is available.

Chapter 11

Summary

In summary, the IPL match win prediction project in data science aims to leverage data-driven approaches to predict the outcomes of Indian Premier League matches. The project's motivation is to enhance the cricket viewing experience, provide valuable insights into match outcomes, assist team management in decision-making, and showcase the potential of data science in the sports domain.

The system architecture involves data collection, preprocessing, feature engineering, model development, model evaluation, model deployment, and continuous improvement. Various approaches can be employed to solve the problem, including supervised learning, time series analysis, ensemble methods, deep learning, and feature engineering. Experimentation, evaluation, and selection of the most effective approach are important for accurate predictions.

The future scope of the project includes incorporating real-time data, such as live player performance statistics, sentiment analysis of fan opinions, predicting player injuries, advanced feature engineering techniques, integration of match strategies, improving interpretability and explainability, exploring cross-domain prediction, and encouraging user interaction and feedback.

By exploring these future opportunities, the IPL match win prediction project can continue to evolve and provide more accurate predictions, valuable insights, and an enhanced user experience for cricket enthusiasts, team management, and individuals interested in betting.

Bibliography

[1] Rabindra Lamsal and Ayesha Choudhary, "Predicting Outcome of Indian Premier League (IPL) Matches Using Machine Learning", arXiv:1809.09813

[2] Abhishek Naik, Shivanee Pawar, Minakshee Naik, SahilMulani, "Winning Prediction Analysis in OneDay-International (ODI) Cricket Using Machine Learning Techniques", International Journal of Emerging Technology and Computer Science, Volume: 3 Issue: 2 (April 2018)

[3] Arjun Singhvi, Ashish Shenoy, Shruthi Racha, and Srinivas Tunuguntla. "Prediction of the outcome of a Twenty-20 Cricket Match." (2015).