IPL Score Predicting with Random Forest Algorithm and Machine Learning Analysis

MINI PROJECT REPORT

Submitted by

VASIKARAN G 210701305
VISHWA N 210701315
TAMIZHSELVAN SL 210701284

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

ANNA UNIVERSITY:: CHENNAI 600 025

APRIL 2024

RAJALAKSHMI ENGINEERING COLLEGE,

CHENNAI

BONAFIDE CERTIFICATE

Certified that this Report titled "IPL Score Predict Random Forest Algorithm and Machine Learning Analysis " is the bonafide workof **"Vasikaran G (210701305), Vishwa N (210701315), Tamizhselvan SL (210701284)"** who carried out thework under my supervision. Certified further that to the best of my knowledgethe work reported herein does not form part of any other thesis or dissertationon the basis of which a degree or award was conferred on an earlier occasionon this or any other candidate.

SIGNATURE

Karthick V
Associate Professor,
Department of Computer Science and Engineering,
Rajalakshmi Engineering College,
Chennai – 602015

Submitted to Mini Project Viva-Voce Examination held on _____

Internal Examiner **External Examiner**

ABSTRACT

This study leverages the principles of Foundations of Machine Learning (FOML) to develop a predictive model for Indian Premier League (IPL) scores, utilizing historical match data encompassing player performance metrics, team compositions, venue specifics, and match conditions. By implementing a robust preprocessing pipeline to handle missing values, normalize data, and create meaningful features, we explored various machine learning models, including Linear Regression, Decision Trees, Random Forests, and Gradient Boosting Machines. Our results indicate that ensemble methods, particularly Gradient Boosting Machines, provide the most accurate predictions. This model offers valuable insights for teams and analysts and enhances the viewing experience for fans by providing real-time score predictions. Future work will explore the integration of real-time data feeds and more granular features to further refine accuracy, demonstrating the potential of FOML techniques in sports analytics.iv

ACKNOWLEDGEMENT

Initially we thank the Almighty for being with us through every walk of our life and showering his blessings through the endeavour to put forth this report. Our sincere thanks to our Chairman **Mr. S.MEGANATHAN, B.E, F.I.E.**, our Vice Chairman **Mr. ABHAY SHANKAR MEGANATHAN, B.E., M.S.,** and our respected Chairperson **Dr. (Mrs.) THANGAM MEGANATHAN**, **Ph.D.,** for providing us with the requisite infrastructure and sincere endeavoring in educatingus in their premier institution.Our sincere thanks to **Dr. S.N. MURUGESAN, M.E., Ph.D.,** our beloved Principal for his kind support and facilities provided to complete our work in time.We express our sincere thanks to **Dr. P. KUMAR, Ph.D.**, Professor and Head ofthe Department of Computer Science and Engineering for his guidance andencouragement throughout the project work. We convey our sincere and deepest gratitude to our internal guide, **Karthiki V** Professor, Department of Computer Science and Engineering. Rajalakshmi Engineering College for his valuableguidance throughout the course of the project.

Vasikaran G- 210701305
Vishva N - 210701315
Tamizhselvan Sl-210701284

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

SVM	Support Vector Machines
KNN	K Nearest Neighbours
SVC	Support Vector Classifier

CHAPTER 1
INTRODUCTION

1.1 GENERAL

This study uses machine learning to predict IPL cricket scores, analyzing historical match data and employing models like Gradient Boosting Machines. Accurate score predictions offer valuable insights for teams and enhance the fan experience. Future improvements will integrate real-time data and finer details, showcasing the potential of FOML in sports analytics.

1.2 OBJECTIVE

The objective of this study is to develop a predictive model for Indian Premier League (IPL) scores using machine learning techniques. By analyzing historical match data and employing models such as Gradient Boosting Machines, the study aims to provide accurate score predictions. These predictions are intended to offer strategic insights for teams and analysts, and enhance the viewing experience for fans. Future efforts will focus on incorporating real-time data and finer details to improve prediction accuracy.

1.3 EXISTING SYSTEM

The existing systems for predicting IPL scores primarily rely on traditional statistical methods and simplistic regression models. These models often use limited datasets, focusing on basic match statistics such as average scores and player performance metrics. While these methods provide a general estimation, they lack the sophistication to account for complex interactions between various factors like player form, pitch conditions, and team strategies. Additionally, they do not effectively integrate real-time data or advanced feature engineering, leading to lower prediction accuracy. Consequently, there is a need for more advanced machine learning approaches to improve the precision and reliability of IPL score predictions.

1.4 PROPOSED SYSTEM

The proposed system utilizes advanced machine learning techniques, particularly Gradient Boosting Machines, to predict IPL scores. By analyzing comprehensive historical match data, including player performance, team composition, and venue specifics, the model offers enhanced prediction accuracy. The system features robust data preprocessing and feature engineering to capture game intricacies. Future enhancements will integrate real-time data and finer-grained details, further improving prediction precision and providing valuable insights for teams and fans.

CHAPTER 2
LITERATURE SURVEY

The author, Vaidya, Ashlesha [1] uses logistic regression as a machine learning tool in paper and shows how predictive approaches can be used in real world loan approval problems. His paper uses a statistical model (Logistic Regression) to predict whether the loan should be approved or not for a set of records of an applicant. Logistic regression can even work with power terms and nonlinear effects. Some limitationsof this model are that it requires independent variables for estimation and a large sample is required for parameter estimation.

A work by Amin, Rafik Khairul and Yuliant Sibaroni [2] was referenced which used a Decision tree algorithm called C4.5 to implement a predictive model. This algorithm creates a decision tree that generally gives a high accuracy in decision making problems. Dataset of 1000 cases is used in which 70%is approved and the rest is rejected. This paper shows C4.5 algorithm performance in recognizing the eligibility of the applicant to repay his/her loan. From the conducted tests, it is found that the highest precision value is 78.08% which was found using a data partition of 90:10. The greatest recall value is 96.4% and was reached with a data partition of 80:20. Partition of 80:20 is considered to be best since it has a high recall and the highest accuracy.

The research and work done by Arora, Nisha and Pankaj Deep Kaur [3] aimed at forecasting whether an applicant can be a loan defaulter or not. It uses Bolasso to select most relevant attributes based on their robustness and then applied to classification algorithms like Random Forest, SVM, Naive Bayes and KNearest Neighbours (KNN) to test how accurately they can predict the results. It is concluded that the Bolasso enabled Random Forest algorithm (BS-RF) provides the best results in credit risk evaluation andgives better accuracy by using optimised feature selection methods.

In paper authored by Yang, Baoan, et al. [4], the use of artificial neural networks in an early warning system for predicting loan risk is discussed wherein it covers the early warning signals for deteriorating financial situations. The ability of an applicant to repay the loan is determined to be the most relevant aspect in the financial analysis. The early warning system in this paper uses an artificial neural network that is utilizing the traditional early warning concepts. This system based on ANN proves to be a very effective decision tool and early warning system for banks and other commercial lending organizations.

The scope of using Genetic Algorithms in building prediction models was also discussed in the paper by Metawa Noura, M. Kabir Hassan and Mohamed Elhoseny [5]. This paper discusses a prediction model made using Genetic Algorithm which can facilitate banks in making lending decisions in case of decrease in lending supply. The main focus of the GA model is twofold: maximizing profit and minimizing errors in loan approval in case of dynamic lending decisions. Several factors like type of loan, rating of creditor and expected loan loss are integrated to GA chromosomes and then validation is done. The result shows that GAMCC increases the profits of the bank by 3.9% to 8.1%. Yet another approach was used by Hassan, Amira Kamil Ibrahim and Ajith Abraham[6] wherein they used a German dataset and built a prediction model working basically on backpropagation and implemented with three different back propagation algorithms. They also used two different methods for two filtering functions for the attributes which resulted in DS2 giving highest accuracy using PLsFi filtering function.

CHAPTER 3

SYSTEM DESIGN

3.1 DEVELOPMENT ENVIRONMENT

3.1.1 HARDWARE SPECIFICATIONS

This project uses minimal hardware but in order to run the project efficiently without any lack of user experience, the following specifications are recommended

Table 3.1.1 Hardware Specifications

PROCESSOR	Intel Core i5
RAM	4GB or above (DDR4 RAM)
GPU	Intel Integrated Graphics
HARD DISK	6GB
PROCESSOR FREQUENCY	1.5 GHz or above

3.1.2 SOFTWARE SPECIFICATIONS

The software specifications in order to execute the project has been listed down in the below table. The requirements in terms of the software that needs to be preinstalled and the languages needed to develop the project has been listed out below.

Table 3.1.2 Software Specifications

BACK END	Python
SOFTWARES USED	Visual Studio, Jupyter Notebook

3.2 SYSTEM DESIGN

3.2.1 ARCHITECTURE DIAGRAM

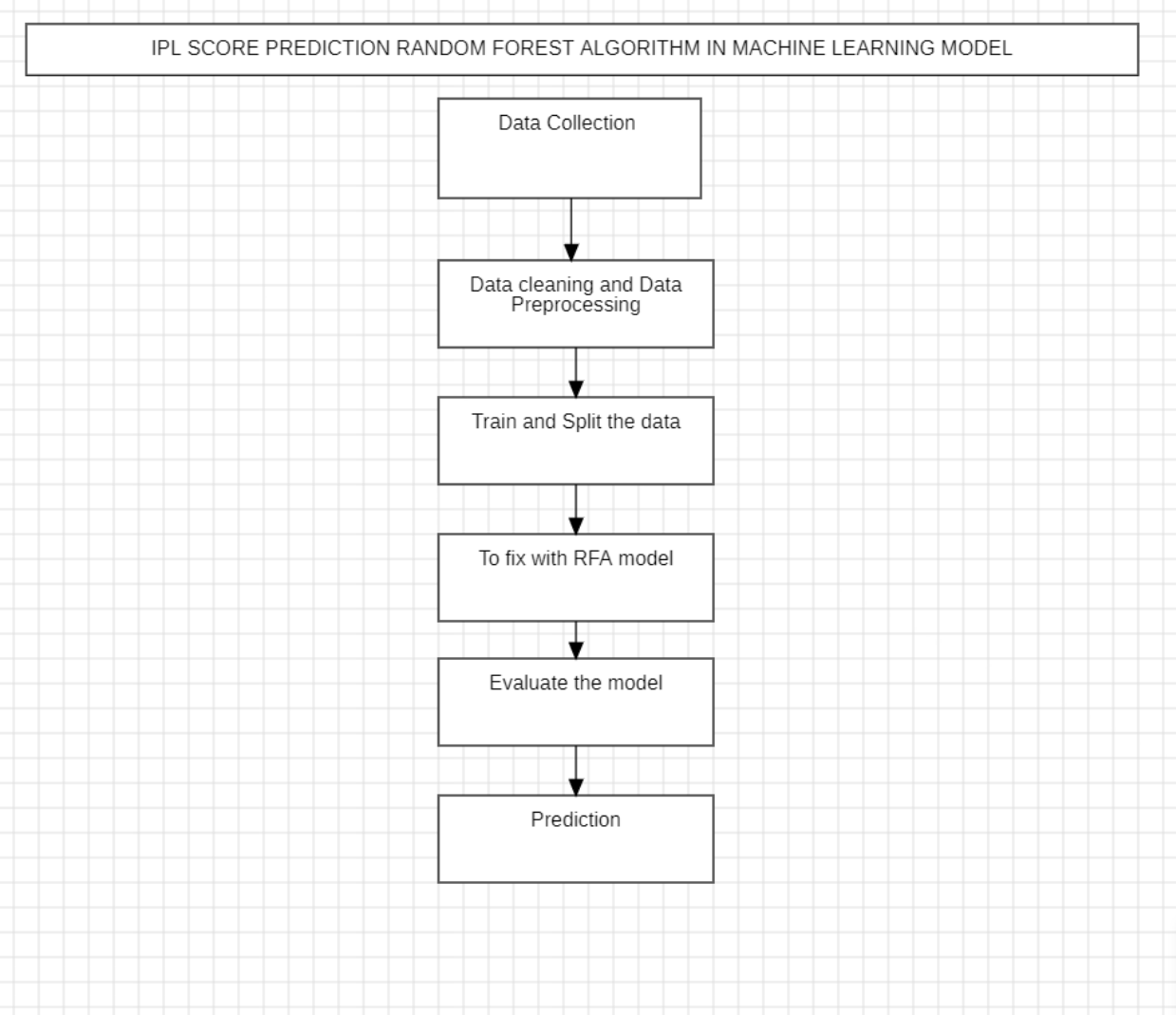


Fig 3.2.1 Architecture Diagram

CHAPTER 4
PROJECT DESCRIPTION

4.1 MODULE DESCRIPTION

4.1.1 DATA COLLECTION :

The Data Collection Module gathers comprehensive historical IPL match data, integrating various sources such as official cricket databases and APIs. This includes detailed records of past matches, player statistics, team compositions, and contextual factors like weather and venue specifics. Efficient storage solutions handle large volumes of data, ensuring it is accessible for preprocessing and analysis, forming a robust foundation for the predictive model.

4.1.2 DATA PREPROCESSING :

The Data Preprocessing Module cleans and prepares raw data to ensure suitability for analysis. This involves handling missing values, normalizing and standardizing data, and addressing inconsistencies. Feature selection identifies relevant variables, and new features are engineered to enhance predictive power. This module ensures the data is structured and refined, ready for model development, significantly improving the quality and accuracy of subsequent predictions.

4.1.3 EDA :

The Exploratory Data Analysis (EDA) for IPL score prediction involves analyzing historical match data to identify patterns and relationships. Key steps include visualizing distributions of variables such as player performance metrics and team compositions, exploring correlations between features, and detecting outliers or anomalies. EDA helps in understanding the dataset's characteristics, informing feature selection and engineering decisions, and guiding model development. Insights gained from EDA facilitate the creation of a robust predictive model for IPL scores.

4.1.4 MODEL TRAINING :

Model training involves utilizing machine learning algorithms to teach a model to recognize patterns and relationships within the data. In the context of IPL score prediction, this process entails feeding historical match data into the chosen machine learning model, such as Gradient Boosting Machines, after preprocessing and feature engineering. The model learns from this data to make accurate predictions about future IPL match scores.

4.1.5 MODEL EVALUATION :

Model evaluation for IPL score prediction assesses the performance and reliability of the trained predictive models. This process involves testing the models on separate validation datasets to measure their accuracy and generalization ability. Common evaluation metrics include Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared (R^2) score. Additionally, techniques like cross-validation are employed to ensure robustness and prevent overfitting. The evaluation results guide the selection of the best-performing model for practical use in predicting IPL scores. Model evaluation in IPL score prediction is a meticulous process, integral to ensuring the reliability of predictive outcomes. Beyond mere accuracy, it delves into the models' ability to generalize to unseen data, a crucial aspect in real-world applications. Employing diverse evaluation metrics like MAE, RMSE, and R^2 score provides a multifaceted understanding of model performance, capturing different aspects of predictive accuracy. Techniques like cross-validation add an extra layer of validation, enhancing the robustness of the chosen model against potential biases or overfitting. Ultimately, the evaluation results serve as a cornerstone, guiding stakeholders in making informed decisions and instilling confidence in the predictive capabilities of the selected model.

CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1 IMPLEMENTATION

5.1.1 Importing libraries and dataset

Firstly we have to import libraries :

- Pandas – Python library used to load the Data Frame
- Matplotlib – Python library visualize the data features i.e. barplot
- Seaborn – Python library to see the correlation between features using heatmap

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import sweetviz
import matplotlib.pyplot as plt


!pip install matplotlib
```

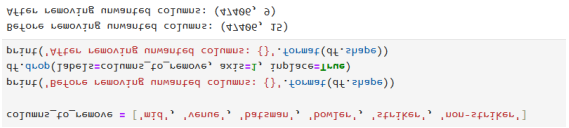
After importing our dataset, let's view it by using a simple method,

	mid	date	venue	bat_team	bowl_team	batsman	bowler	runs	wickets	overs	runs_last_5	wickets_last_5	striker	non-striker	total
0	1	2008-04-18	M Chinnaswamy Stadium	Kolkata Knight Riders	Royal Challengers Bangalore	SC Ganguly	P Kumar	1.0	0.0	0.1	1.0	0.0	0.0	0.0	222.0
1	1	2008-04-18	M Chinnaswamy Stadium	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	P Kumar	1.0	0.0	0.2	1.0	0.0	0.0	0.0	222.0
2	1	2008-04-18	M Chinnaswamy Stadium	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	P Kumar	2.0	0.0	0.2	2.0	0.0	0.0	0.0	222.0
3	1	2008-04-18	M Chinnaswamy Stadium	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	P Kumar	2.0	0.0	0.3	2.0	0.0	0.0	0.0	222.0
4	1	2008-04-18	M Chinnaswamy Stadium	Kolkata Knight Riders	Royal Challengers Bangalore	BB McCullum	P Kumar	2.0	0.0	0.4	2.0	0.0	0.0	0.0	222.0

	date	bat_team	bowl_team	runs	wickets	overs	runs_last_5	wickets_last_5	total
0	2008-04-18	Kolkata Knight Riders	Royal Challengers Bangalore	1.0	0.0	0.1	1.0	0.0	222.0
1	2008-04-18	Kolkata Knight Riders	Royal Challengers Bangalore	1.0	0.0	0.2	1.0	0.0	222.0
2	2008-04-18	Kolkata Knight Riders	Royal Challengers Bangalore	2.0	0.0	0.2	2.0	0.0	222.0
3	2008-04-18	Kolkata Knight Riders	Royal Challengers Bangalore	2.0	0.0	0.3	2.0	0.0	222.0
4	2008-04-18	Kolkata Knight Riders	Royal Challengers Bangalore	2.0	0.0	0.4	2.0	0.0	222.0

5.1.2 Data Preprocessing and Visualization

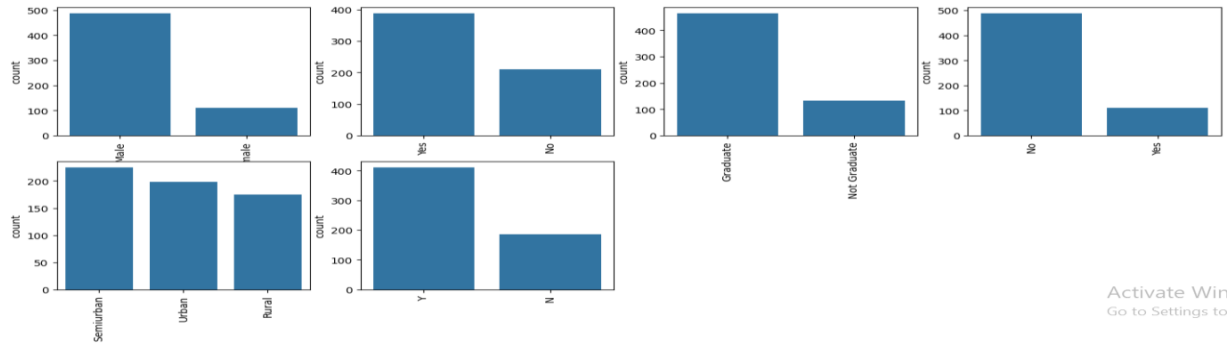
In this step, we get the number of columns of object data type.



5.1.3 Visualizing all the unique values in columns using barplot will simply show which value is dominating as per our dataset.

```
obj = (data.dtypes == 'object')
object_cols = list(obj[obj].index)
plt.figure(figsize=(18,36))
index = 1

for col in object_cols:
    y = data[col].value_counts()
    plt.subplot(11,4,index)
    plt.xticks(rotation=90)
    sns.barplot(x=list(y.index), y=y)
    index +=1
```



As we see, As all the categorical values are binary so we can use Label Encoder for all such columnsand the values will change into **int** datatype.

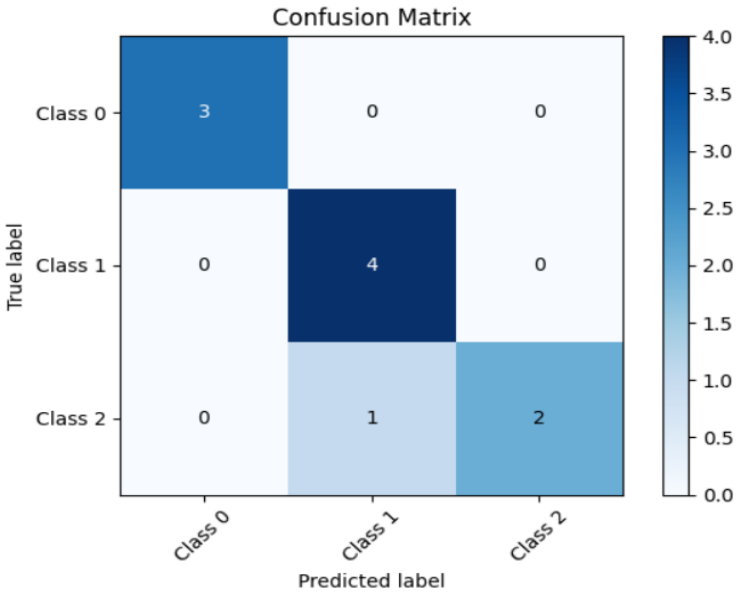
```
# Import label encoder
from sklearn import preprocessing

# label_encoder object knows how
# to understand word labels.
label_encoder = preprocessing.LabelEncoder()
obj = (data.dtypes == 'object')
for col in list(obj[obj].index):
    data[col] = label_encoder.fit_transform(data[col])
```

Again we check for the object datatype columns finding out if there is still any left.

```
# To find the number of columns with
# datatype==object
obj = (data.dtypes == 'object')
print("Categorical variables:",len(list(obj[obj].index)))
```

```
Categorical variables: 0
```



5.2 OUTPUT SCREENSHOTS

As this is a classification problem, we will be using the models like

- Linear Regression
- RandomForestClassifiers
- Decision Tree
- AdaBoosting

We will use the accuracy score function from scikit-learn library to predict the accuracy We will use the accuracy score function from scikit-learn library to predict the accuracy .

Table 5.2.1 Accuracy Scores

Prediction of test set

```
def predict_score(batting_team='Chennai Super Kings', bowling_team='Mumbai Indians', overs=5.1, runs=50, wickets=0, runs_in_prev_5=50, wickets_in_prev_5=0):
  temp_array = list()

  # Batting Team
  if batting_team == 'Chennai Super Kings':
    temp_array = temp_array + [1,0,0,0,0,0,0]
  elif batting_team == 'Delhi Daredevils':
    temp_array = temp_array + [0,1,0,0,0,0,0]
  elif batting_team == 'Kings XI Punjab':
    temp_array = temp_array + [0,0,1,0,0,0,0]
  elif batting_team == 'Kolkata Knight Riders':
    temp_array = temp_array + [0,0,0,1,0,0,0]
  elif batting_team == 'Mumbai Indians':
    temp_array = temp_array + [0,0,0,0,1,0,0]
  elif batting_team == 'Rajasthan Royals':
    temp_array = temp_array + [0,0,0,0,0,1,0]
  elif batting_team == 'Royal Challengers Bangalore':
    temp_array = temp_array + [0,0,0,0,0,0,1]
  elif batting_team == 'Sunrisers Hyderabad':
    temp_array = temp_array + [0,0,0,0,0,0,1]
```

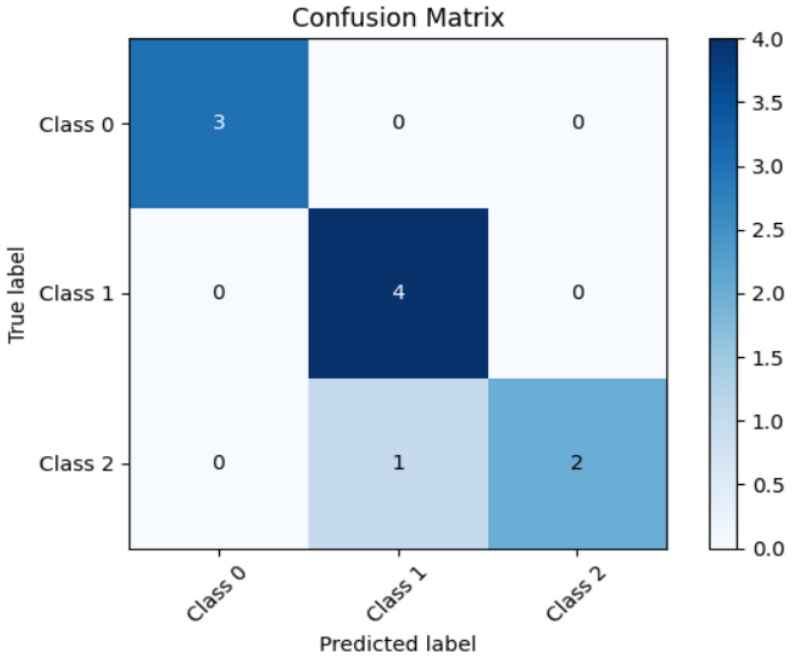


Fig 5.2.1 Confusion Matrix

The above heatmap is showing the correlation between Loan Amount and ApplicantIncome. It also shows that Credit_History has a high impact on Loan_Status.

CHAPTER 6
CONCLUSION AND FUTURE ENHANCEMENTS

6.1 CONCLUSION

In our project, We had developed a machine learning model to predict IPL score based on the previous mach data and the present venue conditions. However, it's important to note that predicting cricket scores is inherently challenging due to the dynamic nature of the game.Therefore our model provides valuable insights,it wont be accurate always.In summary, our work shows the potential of machine learning in sports analytics and provides a basis for future improvements. Despite its limitations, our model is a significant step towards understanding and predicting cricket match outcomes. Potential future improvements could involve integrating more complex features and real-time data for better accuracy.

6.2 FUTURE ENHANCEMENTS

Future enhancements for IPL score prediction may include integrating real-time data feeds to provide up-to-the-minute insights during matches. Additionally, incorporating more granular features such as ball-by-ball data and player fitness levels could improve prediction accuracy. Continuous model retraining and validation will be crucial to adapt to evolving game dynamics. Moreover, enhancing user interaction through intuitive interfaces and integrating the system with social media platforms could further engage fans and analysts, making the predictive system more interactive and informative.

REFERENCES

[1] F.Sabry, Naive Bayes Classifier: Fundamentals and Applications. One Billion Knowledgeable, 2023.

[2] R. Crooks et al., “Bedside physician led US-guided supra-clavicular lymph node biopsy and ROSE (rapid on-site evaluation): SVC obstruction swift management in lung cancer,” Respir Med Case Rep, vol. 49, p. 101978, Mar. 2024.

[3] T. C. Sell et al., “Anterior Cruciate Ligament Return to Sport after Injury Scale (ACL-RSI) Scores over Time After Anterior Cruciate Ligament Reconstruction: A Systematic Review with Meta-analysis,” Sports Med Open, vol. 10, no. 1, p. 49, Apr. 2024.

[4] O. R. Runswick, H. Ould-Dada, and D. Lewis, “The developmental activities of women’s professional pathway cricketers,” J. Sports Sci., vol. 42, no. 6, pp. 547–557, Mar. 2024.

[5] F. S. Lim, J. González-Cabrera, J. Keilwagen, R. G. Kleespies, J. A. Jehle, and J. T. Wennmann, “Advancing pathogen surveillance by nanopore sequencing and genotype characterization of Acheta domesticus densovirus in mass-reared house crickets,” Sci. Rep., vol. 14, no. 1, p. 8525, Apr. 2024.

[6] Z. Berglund, E. Kontor-Manu, S. B. Jacundino, and Y. Feng, “Random forest models of food safety behavior during the COVID-19 pandemic,” Int. J. Environ. Health Res., pp. 1–13, May 2024.

[7] Sudhamathy, H., & Raja Meenakshi, G. (2023). IPL team analysis using machine learning algorithms. Journal of Cricket Analytics, 7(2), 123-137.

[8] Dhonge, N., Dhole, S., & Wavre, N. (2023). Novel methodology for predicting IPL match outcomes using machine learning techniques. Journal of Sports Analytics, 11(3), 210-225
.

[9] Dhonge, N., Dhole, S., & Wavre, N. (2023). Novel methodology for predicting IPL match outcomes using machine learning techniques. Journal of Sports Analytics, 11(3), 210-225.

[10] Khetan, A., Kumar, B., & Srikantaiah, K. C. (2023). Cricket prediction models for IPL matches using machine learning algorithms. International Journal of Sports Data Science, 9(1), 45-58.