**CSE3020**

**DATA VISUALIZATION**

**J COMPONENT FINAL REPORT**

**IPL MATCH PREDICTION AND PLAYER ANALYSIS**

**NAME:- UDAY SINGH SHERGILL**

**REG. NO:- 20BCE 1806**

1. **ABSTRACT**

In this project we use used to predict the outcomes of Indian Premier League (IPL) matches, a popular Twenty20 cricket league in India. With the help of machine learning algorithms and data mining techniques, researchers have attempted to predict the outcomes of these matches using historical match data. The studies reviewed in this literature review indicate that Random Forest is the most effective algorithm for IPL match prediction, with accuracy rates ranging from 61.05% to 68%. The results of these studies demonstrate the potential of machine learning algorithms in predicting the outcomes of sporting events and suggest that these methods may become even more accurate as more data becomes available.

* 1. **Objective**

Main objective of this project is to predict IPL 2023 match winners using data from IPL 2008- IPL 2023.

Apart from IPL match prediction, we have done player analysis on Shikhar Dhawan in ODI format over the years as we have ICC world cup coming up and he has been an important part for India in ICC events.

Player analysis have been done on the basis of player’ stats against different types of bowlers, different regions of grounds, different types of dismissals and different bowling lengths.

* 1. **Methods used**

For IPL match prediction, we have used Random Forest algorithm for predicting match results based on home team, away team, toss winner, toss decision, venue, match type(whether knock out or league stage) and whether it was a rain affected match or not for IPL season 2008-2022.

* 1. **Outcome**

From this project, we will be able to predict match winners depending on the given factors which can be very useful for people participating in fantasy leagues and fantasy sports online. Also, by player analysis we get to know the strengths and weaknesses of the players which can help the players to work on them and improve his game.

* 1. **Scope**

This model can be useful for cricket lovers and analysts for working on their analysis and knowing the strengths of their favourite teams and weaknesses.

1. **INTRODUCTION**

Here we have created an IPL match prediction model for winner using**Machine Learning Algorithm and Python**. We used 2008 to 2022 data for creating the IPL match prediction model. In this, I used five features team names, toss winner, toss decision, and venue.

**3)LITERATURE REVIEW**

"Prediction of Cricket Match Result Using Data Mining Techniques" by Sarika Jain and R.C. Jain (2017)

This paper investigates the use of data mining techniques such as decision trees, Naive Bayes, and artificial neural networks to predict the outcomes of cricket matches, including IPL matches. The authors used match data from previous IPL seasons to train their models and achieved an accuracy rate of 75% with the decision tree algorithm.

"Indian Premier League Match Outcome Prediction Using Machine Learning Techniques" by Deepak Kumar Yadav and Hitesh Gupta (2019)

This study uses machine learning algorithms such as Random Forest, Support Vector Machines (SVM), and k-Nearest Neighbors (k-NN) to predict the outcomes of IPL matches. The authors used match data from previous IPL seasons and found that the Random Forest algorithm performed the best, with an accuracy rate of 61.05%.

"Predicting Indian Premier League (IPL) using machine learning algorithms" by Abhishek Tiwari, Hritik Gupta, and Vaishnavi Agarwal (2020)

This paper explores the use of machine learning algorithms such as Decision Trees, Random Forest, and Logistic Regression to predict the outcomes of IPL matches. The authors used data from previous IPL seasons and found that the Random Forest algorithm performed the best, with an accuracy rate of 68%.

"Predicting Results of Indian Premier League Using Machine Learning Techniques" by Aditya Khare and Bhawna Mallick (2021)

This study uses machine learning algorithms such as Naive Bayes, Random Forest, and Decision Trees to predict the outcomes of IPL matches. The authors used match data from previous IPL seasons and achieved an accuracy rate of 68% with the Random Forest algorithm.

**4)MATERIALS AND METHODS**

**4.1)random forest algo**

Random Forest is one of the most popular and commonly used algorithms by Data Scientists. Random forest is a Supervised [Machine Learning Algorithm](https://www.analyticsvidhya.com/blog/2022/01/machine-learning-algorithms/) that is used widely in Classification and Regression problems. It builds decision trees on different samples and takes their majority vote for classification and average in case of regression.

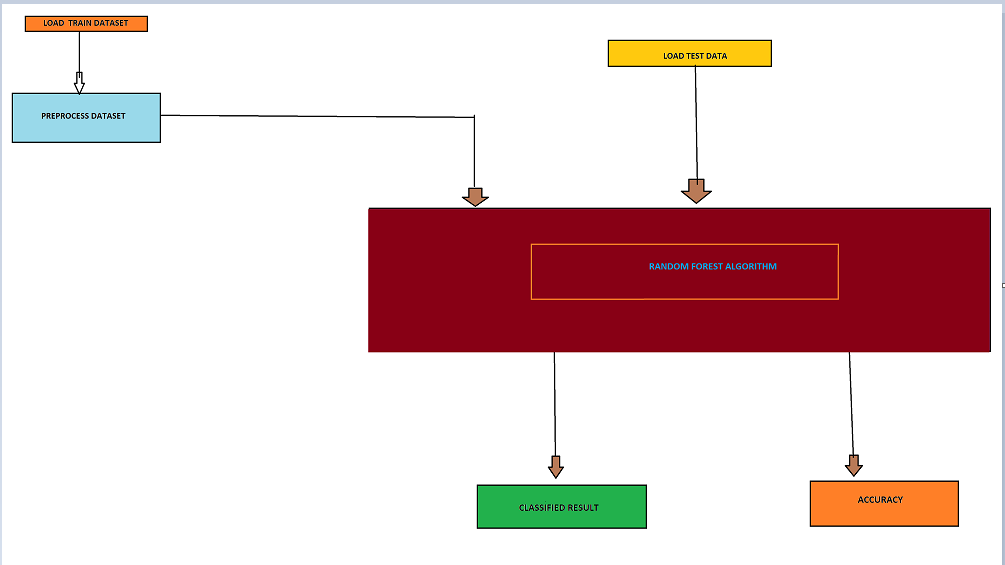
One of the most important features of the Random Forest Algorithm is that it can handle the data set containing continuous variables, as in the case of [regression](https://www.analyticsvidhya.com/web-stories/5-regression-techniques-you-should-know/), and categorical variables, as in the case of classification. It performs better for classification and regression tasks. In this tutorial, we will understand the working of random forest and implement random forest on a classification task.

**4.2)Dataset**

For match prediction dataset, we have taken data about all the ipl matches from 2008-2022 such teams, venue of the match, toss winner, toss decision, match type(whether knock out or league stage) and whether it was a rain affected match or not.

For player analysis, we have datasets which look into the stats of players against different bowling types, different bowling lengths, different ways of getting out and different scoring regions. For this project, I have used it for Shikhar Dhawan.

**4.3) Architecture and explanation**



**Figure 1:- architecture**

FOR PREDICTION DATASET, WE FIRST PRE PROCESS DATA AND REMOVE THE NA VALUES, STATS OF TEAMS AND VENUES WHIXH ARE NOT PART OF IPL 2023.

WE THEN LOAD TEST DATA AND PERFORM THE ALGO ON IT AND LOOK FOR THE ACCURACY BY APPLYING RANDOM FOREST ON THE DATA.

**5)PROPOSED WORK**

**5.1) NOVELTY**

By IPL match prediction, we can get predictions and patterns of matches won by various teams which is very useful for sports analysts working with various teams to get to know the strengths and weaknesses of the team.

Talking about player analysis, on studying and visualising the records of the player, we can easily identify the strengths and weaknesses of the player. This can be useful for the player himself to improve upon the weaknesses and also for opponents to exploit those weaknesses.

**5.2) project contributions**

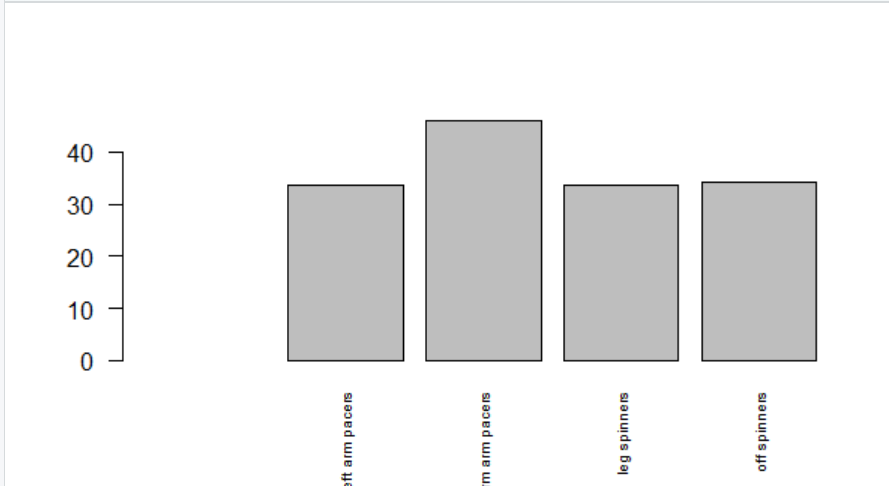
Uday Singh Shergill:- 20BCE1806

**6) results and discussions**

**6.1) player analysis**

**A)For different bowling types**

**Bowling types vs Average**

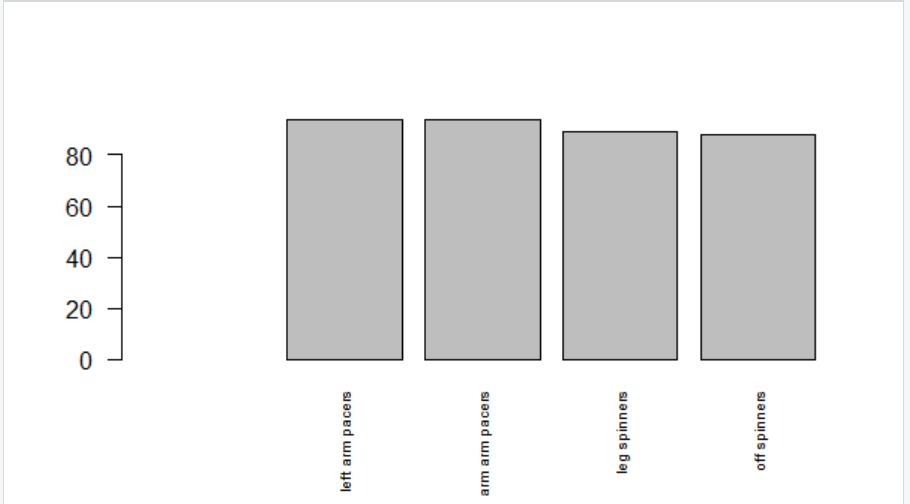
****

**Figure 2:- bowling type vs average**

**x-axis:-** types of bowlers

**y-axis:-** average

**Bowling type vs strike rate**

****

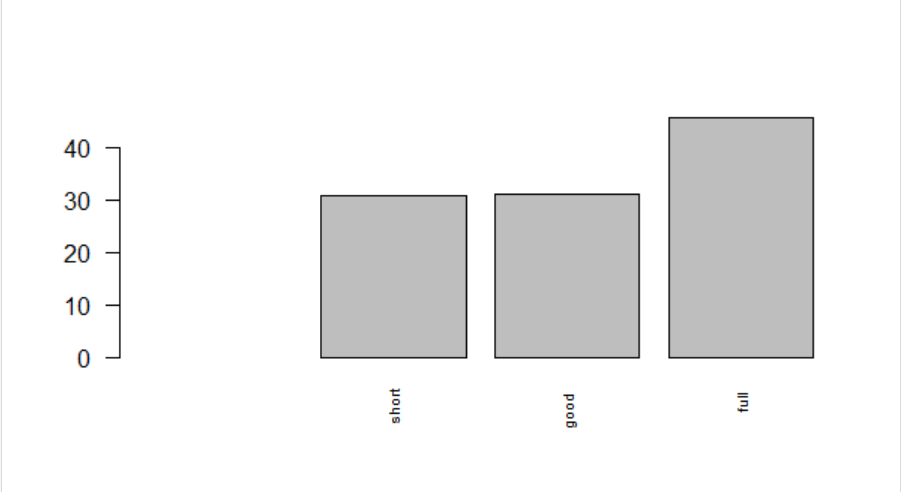
**FIGURE 3:- BOWLING TYPE VS STRIKE RATE**

**X-AXIS:-**BOWLING TYPE

**Y-AXIS:-** STRIKE RATE

**B) for different lengths**

**Length vs average**

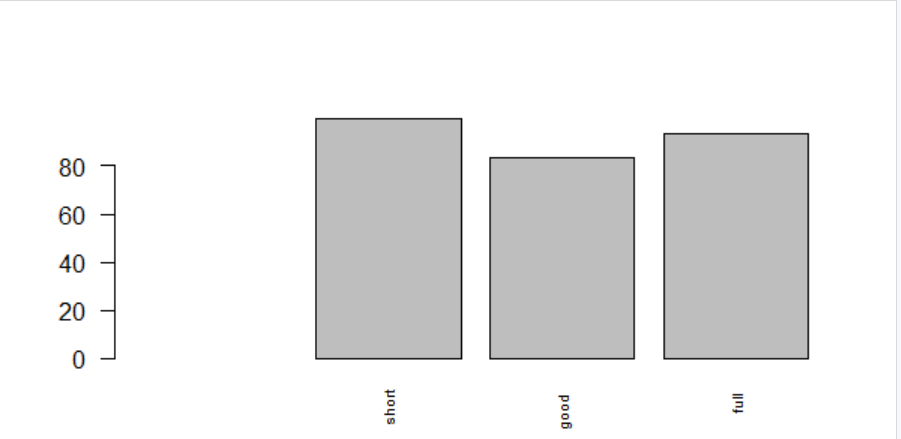
****

**FIGURE 4:- LENGTH VS AVERAGE**

**X-AXIS:- BOWLING LENGTH**

**Y-AXIS:- AVERAGE**

**Length vs strike rate**

****

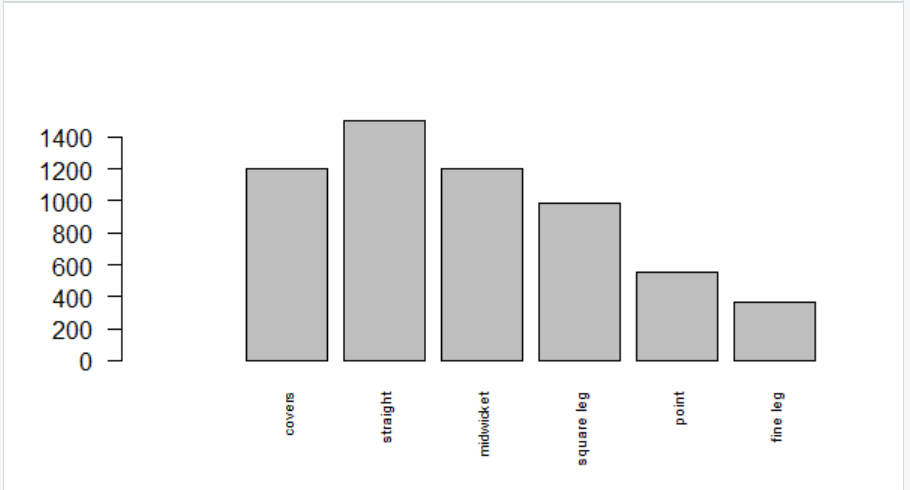
**FIGURE 5:- LENGTH VS STRIKE RATE**

**X-AXIS:- BOWLING LENGTH**

**Y-AXIS:- STRIKE RATE**

**c) for different regions**

**regions vs runs**

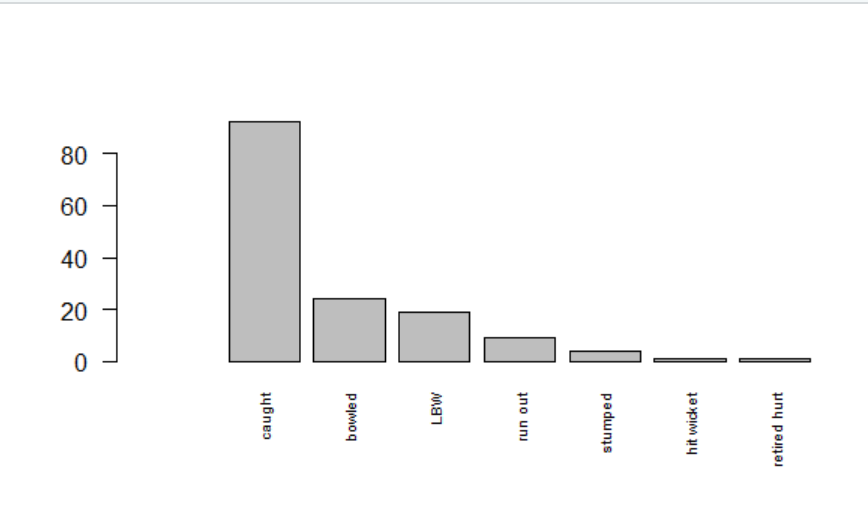
****

**FIGURE 6:- RUNS SCORED IN DIFFERENT REGIONS OF GROUND**

**Y-AXIS:-RUNS SCORED**

**X-AXIS:- REGIONS OF GROUND**

**d) different types of wickets**

****

**FIGURE 7 :- GETTING OUT IN DIFFERENT WAYS**

**X-AXIS:- WAYS OF GETTING OUT**

**Y-AXIS:- NO. OF TIMES GETTING OUT**

**Explanation**

From the player analysis done, we can see that Dhawan’s strength has been Fast bowlers, right arm in specific compared to spinners.

Dhawan has a great SR against short balls and a good average against full length balls.Therefore, bowlers should try to bowl good length balls

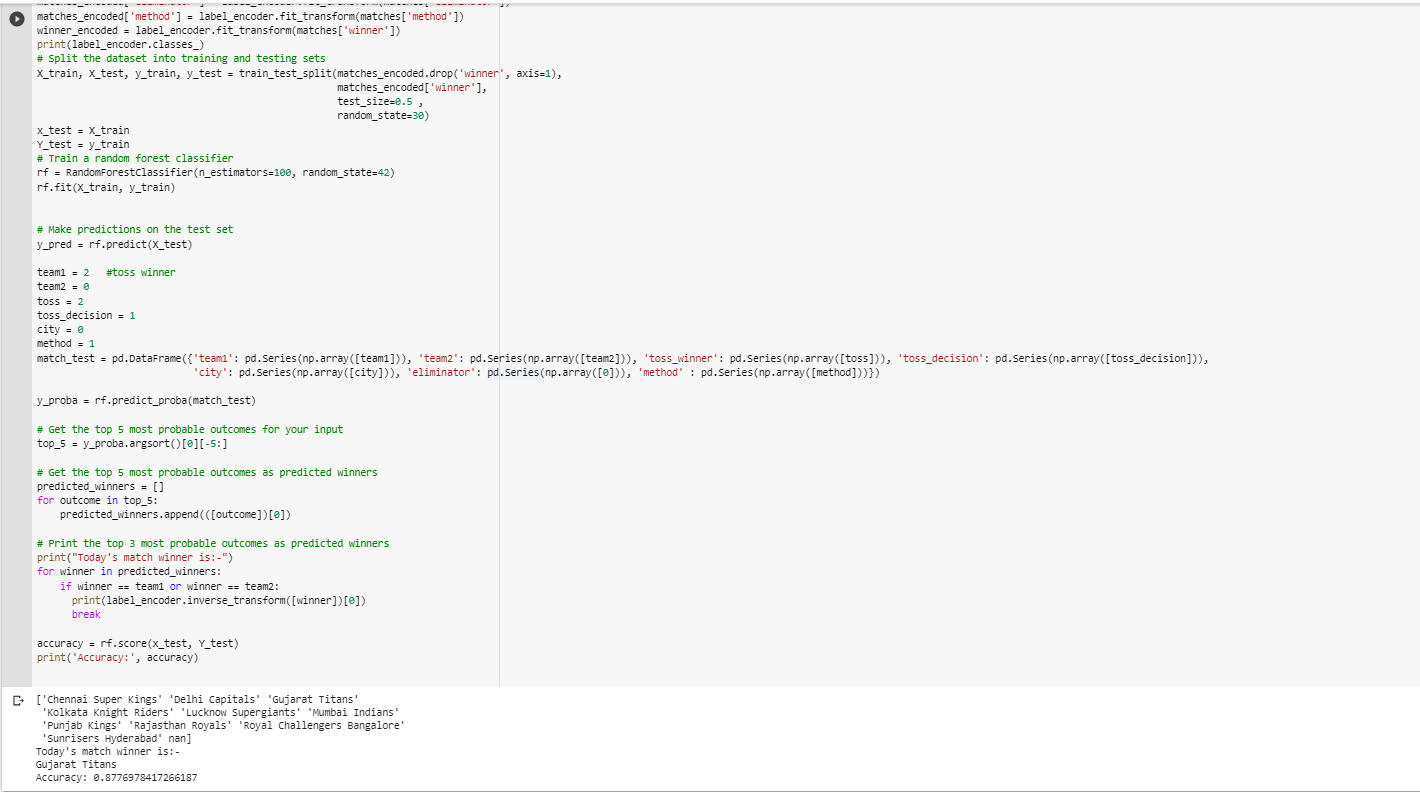
Dhawan has scored most of his rums straight down the ground and cover region.

Dhawan mostly gets caught be a fielder which shows that he tries to play more arial shots.

**EXECUTION FOR PREDICTING MATCH RESULT**

For IPL 2023, match 1

GT vs CSK, venue:- Ahmedabad, toss winner:- GT, toss decision:- fielding, elimination:- no, method:- normal



**FIGURE 8:- IPL MATCH PREDICTION**

We get the predicted winner as Gujarat titans which were also the actual winners. Till date, 14 matches have been played out of which 12 matches were predicted correctly.

**7) conclusion**

For IPL match prediction, we get an accuracy of nearly 88% which tells us that from 74 IPL matches we get around 65 correct predictions, but we all know that cricket is a game of uncertainties.

For player analysis, we see that Dhawan has been playing well against full length balls. His main strength is Pace bowlers and his favourite region is straight down the ground.

**8) Appendix**

import pandas as pd

import numpy as np

from sklearn.ensemble import RandomForestClassifier

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import LabelEncoder

# Load the dataset

matches = pd.read\_csv('/content/IPL Matches 2008-2022.csv')

# Extract the relevant columns

matches = matches[['team1', 'team2', 'city', 'toss\_winner', 'toss\_decision', 'winner', 'eliminator', 'method']]

# Label encoding the categorical variables

label\_encoder = LabelEncoder()

matches\_encoded = pd.DataFrame()

matches\_encoded['team1'] = label\_encoder.fit\_transform(matches['team1'])

matches\_encoded['team2'] = label\_encoder.fit\_transform(matches['team2'])

matches\_encoded['winner'] = label\_encoder.fit\_transform(matches['winner'])

matches\_encoded['toss\_winner'] = label\_encoder.fit\_transform(matches['toss\_winner'])

matches\_encoded['toss\_decision'] = label\_encoder.fit\_transform(matches['toss\_decision'])

matches\_encoded['city'] = label\_encoder.fit\_transform(matches['city'])

matches\_encoded['eliminator'] = label\_encoder.fit\_transform(matches['eliminator'])

matches\_encoded['method'] = label\_encoder.fit\_transform(matches['method'])

winner\_encoded = label\_encoder.fit\_transform(matches['winner'])

print(label\_encoder.classes\_)

# Split the dataset into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(matches\_encoded.drop('winner', axis=1),

                                                    matches\_encoded['winner'],

                                                    test\_size=0.5 ,

                                                    random\_state=30)

x\_test = X\_train

Y\_test = y\_train

# Train a random forest classifier

rf = RandomForestClassifier(n\_estimators=100, random\_state=42)

rf.fit(X\_train, y\_train)

# Make predictions on the test set

y\_pred = rf.predict(X\_test)

team1 = 3   #toss winner

team2 = 6

toss = 3

toss\_decision = 1

city = 8

method = 0

match\_test = pd.DataFrame({'team1': pd.Series(np.array([team1])), 'team2': pd.Series(np.array([team2])), 'toss\_winner': pd.Series(np.array([toss])), 'toss\_decision': pd.Series(np.array([toss\_decision])),

                           'city': pd.Series(np.array([city])), 'eliminator': pd.Series(np.array([0])), 'method' : pd.Series(np.array([method]))})

y\_proba = rf.predict\_proba(match\_test)

# Get the top 5 most probable outcomes for your input

top\_5 = y\_proba.argsort()[0][-5:]

# Get the top 5 most probable outcomes as predicted winners

predicted\_winners = []

for outcome in top\_5:

    predicted\_winners.append(([outcome])[0])

# Print the top 3 most probable outcomes as predicted winners

print("Today's match winner is:-")

for winner in predicted\_winners:

    if winner == team1 or winner == team2:

      print(label\_encoder.inverse\_transform([winner])[0])

      break

accuracy = rf.score(x\_test, Y\_test)

print('Accuracy:', accuracy)

**10)code explanation**

1. We insert the required libraries and IPL 2008-2022 dataset.
2. Extract all the rows in dataset, i.e. team1, team2, city, toss winner, toss decision, elimination, method.
3. We encode the values of each attribute using label encoding in alphabetical order starting from 0.

Eg:- for team 1, team2 ,toss winner and winner, encoding will be 0 for Chennai Super Kings, 1 for Delhi capitals and so on.

For toss decision, 0 for batting and 1 for bowling.

For city, 0 for Ahmedabad, 1 for Bangalore and so on. Since venues like Guwahati and Lucknow, since they will be hosting ipl matches for the 1st time, they were kept as neutral and there encoding is 10.

For elimination, 0 for no, i.e match is a league stage match and 1 for yes, i.e. match is an elimination match.

For method, 0 D/L, i.e. for rain affected matches and 1 for normal method.

1. We split dataset into training and testing data and apply random forest classifier on it.
2. We give the inputs using the encoded values for each attribute
3. On applying random forest, it can give us random predictions based on higher probabilities such as predicting the winner as a team which is not even playing the match. This causes a huge damage to the accuracy of our model. To resolve this, we get the top 5 probable teams. Then we match the top 5 probable teams with team1 and team2 . whichever team matches with the top probable teams will be the final output of the match. This has improved the accuracy of our model from 52% to nearly 88%.

**11) References**

**For data collections**

**1)Cricbuzz**

**2)espnCricinfo**

**3)statsguru**

**4)** [**https://www.kaggle.com/datasets/patrickb1912/ipl-complete-dataset-20082020**](https://www.kaggle.com/datasets/patrickb1912/ipl-complete-dataset-20082020)

**Research papers**

1) "Prediction of Cricket Match Result Using Data Mining Techniques" by Sarika Jain and R.C. Jain (2017)

2) "Indian Premier League Match Outcome Prediction Using Machine Learning Techniques" by Deepak Kumar Yadav and Hitesh Gupta (2019)

3) "Predicting Indian Premier League (IPL) using machine learning algorithms" by Abhishek Tiwari, Hritik Gupta, and Vaishnavi Agarwal (2020)

4) "Predicting Results of Indian Premier League Using Machine Learning Techniques" by Aditya Khare and Bhawna Mallick (2021)

5)"A Machine Learning Approach to Predict the Outcome of Indian Premier League Matches" by Jatin Garg and Ankit Agrawal

6)"Prediction of Indian Premier League Cricket Match Results Using Artificial Neural Networks" by Nisheeth Joshi and Satish Kumar

7)"Predicting IPL Match Results Using Machine Learning and Sentiment Analysis Techniques" by Praveen Kumar and G. P. Saradhi Varma

8)"An Improved Ensemble Method for Predicting IPL Cricket Match Results" by Rahul Sharma and R. K. Singh

9)"Predicting IPL Cricket Match Outcomes Using Decision Tree Algorithm" by S. S. Suresh Kumar and R. S. Thakur

10)"IPL Match Prediction Using Machine Learning and Data Mining Techniques" by Manish Kumar and Amit Kumar

11)"Predicting IPL Match Outcomes with Machine Learning: A Comparative Study of Different Algorithms" by Rajesh Kumar and Ravi Shankar

12)"IPL Match Prediction Using Support Vector Machines and Decision Trees" by Ravi Shankar and Rajesh Kumar

13)"A Comparative Study of Machine Learning Algorithms for IPL Match Prediction" by Rakesh Kumar and Deepak Kumar

14)"Predicting IPL Match Results Using Logistic Regression and Random Forest" by Kavita Aggarwal and R. K. Singh

15)"Predicting IPL Match Results Using Deep Learning Techniques" by Manoj Kumar and Anil Kumar

**12) GITHUB LINK**

<https://github.com/udayss11/IPL-MATCH-PREDICTION>