

VISUALIZATION AND PREDICTION OF THE INDIAN PREMIER LEAGUE USING MACHINE LEARNING

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Engineering degree in Computer Science and Engineering

by

Pratik Satpati (Reg.No.37110591)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING SCHOOL OF
COMPUTING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY (DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC I2B Status by UGC Approved by AICTE

JEPPIAARNAGAR, RAJIVGANDHISALAI, CHENNAI-600119

March- 2021



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC I12B Status by UGC Approved by AICTE

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai-600119

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of Pratik Satpati (Reg.No.37110591) and who carried out the project entitled

"VISUALIZATION AND PREDICTION OF THE INDIAN PREMIER LEAGUE USING MACHINE LEARNING" under my supervision from November 2020 to March 2021.

Internal Guide

Mrs.M.D.Anto Praveena M.C.A., M.E.,(Ph.D)

Head of the Department

Dr.S.VIGNESHWARI M.E.,Ph.D.,

Dr.L.LAKSHMANAN M.E.,Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **PRATIK SATPATI** (Reg.No.37110591) hereby declare that the Project Report entitled **“VISUALIZATION AND PREDICTION OF THE INDIAN PREMIER LEAGUE USING MACHINE LEARNING”** done by us under the guidance of **Mrs M.D.Anto Praveena M.C.A.,M.E.,(Ph.D)**.is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Computer Science and Engineering.

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr.T.Sasikala M.E., Ph.D., Dean, School of Computing and Dr.L.Lakshmanan M.E., Ph.D., and Dr.S.Vigneshwari M.E., Ph.D., Heads of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Mrs M.D.Anto Praveena M.C.A., M.E., (Ph.D). for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

Data Mining & Machine Learning in Sports Analytics is a blooming sector in the field of Computer Science. Cricket is one of the most popular team games in the world. With this project, we embark on predicting the outcome of Indian Premier League (IPL) cricket match which is the biggest carnival of T20 format in the world of cricket. This project aims at designing an effective result prediction system for a cricket match. The result of a T20 cricket match depends on lots of In-game and pre-game attributes, like venue, Past track-records and toss influence the results of the match predominantly. This project also aims to emphasize on exploratory data analysis, modelling and visualization of data regarding the Indian Premier League. Best possible outcome of a given match will be predicted using different supervised machine learning (Random Forest Classifier) and statistical approaches. For easy access and usage of the outcome, this will be hosted on a user-friendly web application that can run on any browser.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	v
	LIST OF ABBREVIATIONS	vii
	LIST OF FIGURES	viii
1.	INTRODUCTION	01
	1.1. INTRODUCTION	01
	1.2. OUTLINE OF THE PROJECT	02
2.	LITERATURE SURVEY	03
	2.1. RELATED WORK	03
3.	AIM AND SCOPE	04
	3.1. AIM OF THE PROJECT	04
	3.2. OBJECTIVE AND SCOPE	04
4.	SYSTEM IMPLEMENTATION	05
	4.1. SYSTEM ARCHITECTURE	05
	4.2. METHODS AND MODEL DETAILS	06
	4.2.1 IPL DATA ANALYTICS	06
	4.2.2 MATCH PREDICTION	08
5.	RESULTS AND DISCUSSION	14
	5.1. DATA ANALYTICS	14
	5.2. MATCH PREDICTION	19
6.	SUMMARY AND CONCLUSION	21
7.	APPENDIX	23
	A) SOURCE CODE	23
	B) REFERENCES	32

LIST OF SYMBOLS AND ABBREVIATIONS

ABBREVIATION	FULLFORM
IPL	INDIAN PREMIER LEAGUE
ML	MACHINE LEARNING
SVM	SUPPORT VECTOR MACHINE
KNN	NEAREST NEIGHBOUR
EDA	EXPLORATORY DATA ANALYSIS

LIST OF FIGURES

FIG.NO	FIG NAME	PAGE NO
4.1	ARCHITECTURE DIAGRAM	05
4.2	RANDOM FOREST CLASSIFIER	10
4.3	DECISION TREE	12
5.1	WELCOME PAGE	14
5.2	TEAMWISE PERFORMANCE	15
5.3	IMPACT ON TOSS	16
5.4	IMPACT OF TOSS DECISION	16
5.5	RUNS SPLIT OF A BATSMAN	17
5.6	WICKETS SPLIT OF A BOWLER	18
5.7	MOST MAN OF THE MATCH AWARDS	18
5.8	PRE-TOSS PREDICTION	19
5.9	POST-TOSS PREDICTION	20

CHAPTER 1

INTRODUCTION

1.1 INTRODUCTION

The game of cricket is played in various formats, i.e., One Day International, T20 and Test Matches. The Indian Premier League (IPL) is a Twenty-20 cricket tournament league established with the objective of promoting cricket in India and thereby nurturing young and talented players. The league is an annual event where teams representing different Indian cities compete against each other. It was started by the Board of Control for Cricket in India (BCCI) and has now become a giant, remunerative cricket venture. The teams for IPL are selected by means of an auction. Players' auctions are not a new phenomenon in the sports world. However, in India, selection of a team from a pool of available players by means of auctioning of players was done in Indian Premier League (IPL) for the first time. Due to the involvement of money, team spirit, city loyalty and a massive fan following, the outcome of matches is very important for all stake holders. This, in turn, is dependent on the complex rules governing the game, luck of the team (Toss), the ability of players and their performances on a given day. Various other natural parameters, such as the historical data related to players, play an integral role in predicting the outcome of a cricket match. A way of predicting the outcome of matches between various teams can aid in the team selection process. However, the varied parameters involved present significant challenges in predicting accurate results of a game. Moreover; the accuracy of a prediction depends on the size of data used for the same. The tool presented in this paper can be used to evaluate the performance of players. This tool provides a visualisation of players' performances. Using IPL T-20 variables related to statistics of batsmen and bowlers, a number of apt variables have been identified that have elucidative power over auction values. Further, several predictive models are also built for predicting the result of a match, based on each player's past performance as well as some match related data. The developed models can help decision makers during the IPL matches to evaluate the strength of a team against another.

1.2 OUTLINE OF THE PROJECT

Statistical Modelling and Data Mining tools are being used in Sports Analytics and prediction vividly now a days. This gives us an opportunity to analyse and predict the outcome of a game (like – Indian Premier League) using different visualization tools and machine learning algorithms. Cricket has been established as one of the most followed outdoor game in the world; over 1.5 billion people watch cricket worldwide including Asia, Australia, Europe, Africa etc. In India itself cricket has over 766 Million viewers who love to watch the sport. Cricket has had many evolutions over time; in 2005 Cricket saw the inception of it's shortest and the most entertaining format of the game called T20. The idea of Indian Premier League was conceived in 2007 after the first successful T20 World Cup with the objective of promoting T20 cricket in India and thereby nurturing young and talented players. It was started by BCCI (Board of Control for Cricket in India) and now has become a massive, remunerative annual venture and considered as the best of all the T20 Leagues in the world. In this tournament 8 different teams representing different provinces of India play in a Round Robin fashion for the ulterior motive of winning the prestigious trophy and a prize money of 10 crore Indian Rupees. Each IPL team consists of 11 players out of which 4 overseas and 7 local players play together. These players are bought in an annual auction from a pool of available players. IPL has a brand value of over 510 Crores where brands from India and all over the world invest in it. The outcome of the IPL matches is very important for all the stake holders due to the involvement of money, city loyalty, team spirit and massive fan following. The generated data about the players and the teams help in doing a proper SWAT analysis of strength and weakness of both the players and the teams. The outcome of a match depends on various factors like – Toss decision, ability of the players, the previous win -loss record against each other. This project thus aims to analyse the Team and Player data generated in IPL as well as predict the outcome of an IPL game by taking into consideration factors like toss, toss decision and venue.

CHAPTER 2

LITERATURE SURVEY

2.1 RELATED WORK

With the evolution of Cricket, it became a very hot topic for sports analysts. A lot of research has been made on cricket but due to inconsistent and complicated data sets, they could not get breakthrough in predicting match winner accurately. There are many techniques that has been used in predicting match winner like KNN, Logistic Regression, SVM, Naïve Bayes but nobody has achieved the accuracy. According to Ahmed & Nazir [1] they implemented different statistical approaches for formation of datasets and tried various classification techniques to predict the winner of One Day Cricket (50 over) match. He has predicted the winner with 80 % accuracy. Shah predicted One Day International match. In Features combination to predict the match outcome, is relative strength of Team B divided by relative strength of Team A is successful in measuring and comparing the strength of the playing teams. implemented Logistic Regression on this data and achieved accuracy in predicting the results by using data of ICC match ratings, ICC ranking points for batsmen and bowlers, home factor, ICC rating differences and ground effects on the match. The machine learning based approach used in [5] is reached at by an in-depth analysis of T20 cricket features. In order to indicate the players' performance, a novel index, namely Deep Performance Index (DPI) is derived using the characteristics specific to T20 cricket. The authors extract relevant features using the machine learning algorithm of Recursive Feature elimination for designing the DPI. It is demonstrated that DPI achieves better results in analysis of performance related data for batsmen as well as bowlers in comparison to some other ranking methods for T20 cricket. There exist some other approaches [6,7] which have specifically worked upon IPL data

CHAPTER 3

AIM AND SCOPE

3.1 AIM OF THE PROJECT

This project aims at designing an effective result prediction system for a cricket match. The result of a T20 cricket match depends on lots of In-game and pre-game attributes, like venue, Past track-records and toss influence the results of the match predominantly. This project also aims to emphasize on exploratory data analysis, modelling and visualization of data regarding the Indian Premier League. Best possible outcome of a given match will be predicted using different supervised machine learning (Random Forest Classifier) and statistical approaches. For easy access and usage of the outcome, this will be hosted on a user-friendly web application that can run on any browser.

3.2 OBJECTIVE AND SCOPE

To predict the outcome of an IPL match. It also aims to analyse and visualize data using various data visualisation techniques for better understanding. The data has to be pre-processed and fed to various supervised machine learning algorithms and analysed in accordance to their accuracies. The best possible outcome will be predicted using a perfect model and will be hosted in a user-friendly web application.

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 SYSTEM ARCHITECTURE:

The proposed system aims to analyse the data generated by IPL matches and predict the outcome of the match (one Pre-Toss and then Post-Toss). The steps followed are –

- Collecting data by scraping
- Data pre-processing
- Prediction using supervised learning algorithm (Random Forest Classifier)
- Data Visualization
- Deploying in a web application

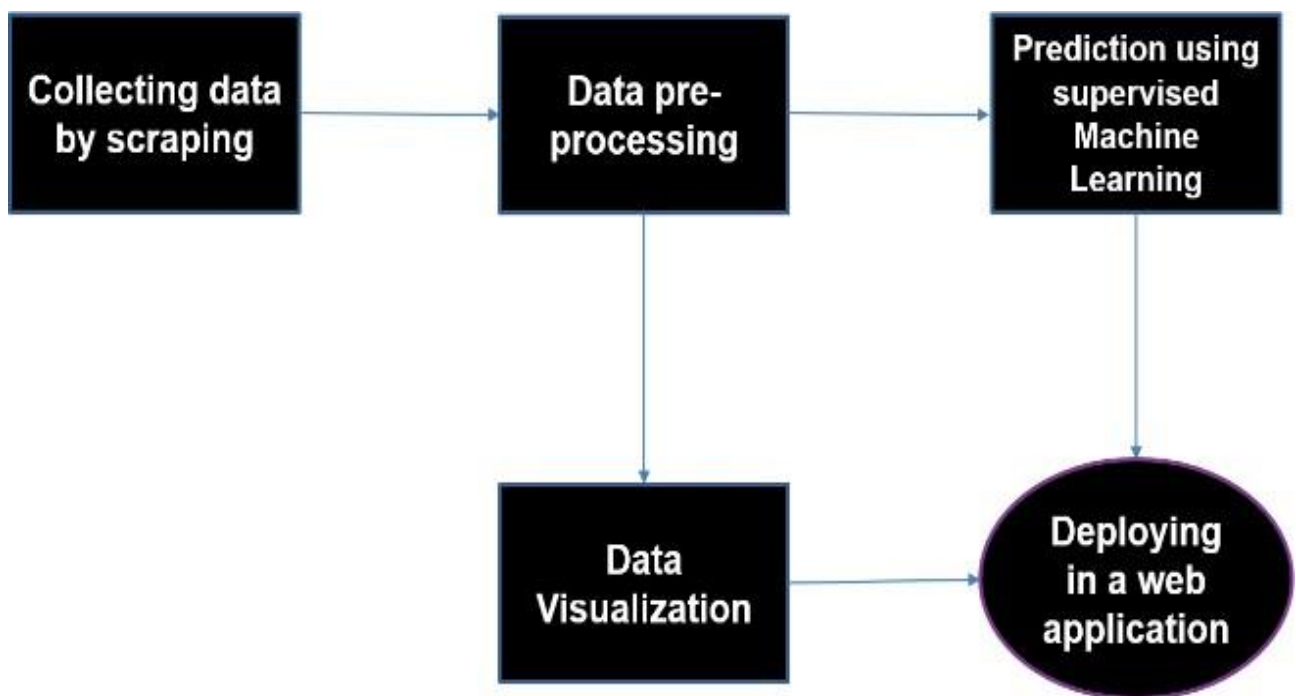


Figure 4.1: Architecture Diagram

4.2 METHODS AND MODEL DETAILS:

This project mainly has three parts:

- IPL Data Analytics (Team and Player Stats)
- Pre-Toss Prediction
- Post-Toss Prediction

4.2.1 IPL DATA ANALYTICS:

As the process of analysing raw data to find trends and answer questions, the definition of data analytics captures its broad scope of the field. However, it includes many techniques with many different goals. The data analytics process has some components that can help a variety of initiatives. By combining these components, a successful data analytics initiative will provide a clear picture of where you are, where you have been and where you should go. Statistics have always had a significant role in sports. As I mentioned above, sports analytics is on the rise and will continue to play a significant role in how teams operate, pick their players, how they play the game, etc. Cricket is no different. The runs scored by a batsman, the wickets taken by a bowler, or the matches won by a cricket team – these are all examples of the most important numbers in the game of cricket.

Maintaining a record of all such statistics has multiple benefits. The teams and the individual players can dig deep into this data and find areas of improvement. It can also be used to assess an opponent's strengths and weaknesses. Data analytics is a broad field. There are four primary types of data analytics: descriptive, diagnostic, predictive and prescriptive analytics. Each type has a different goal and a different place in the data analysis process. These are also the primary data analytics applications in business.

- Descriptive analytics helps answer questions about what happened. These techniques summarize large datasets to describe outcomes to stakeholders. By developing key performance indicators (KPIs,) these strategies can help track successes or failures. Metrics such as return on investment (ROI) are used in many industries. Specialized metrics are developed to track performance in specific industries. This process

requires the collection of relevant data, processing of the data, data analysis and data visualization. This process provides essential insight into past performance.

- Diagnostic analytics helps answer questions about why things happened. These techniques supplement more basic descriptive analytics. They take the findings from descriptive analytics and dig deeper to find the cause. The performance indicators are further investigated to discover why they got better or worse. This generally occurs in three steps:
 - Identify anomalies in the data. These may be unexpected changes in a metric or a particular market.
 - Data that is related to these anomalies is collected.
 - Statistical techniques are used to find relationships and trends that explain these anomalies.
- Predictive analytics helps answer questions about what will happen in the future. These techniques use historical data to identify trends and determine if they are likely to recur. Predictive analytical tools provide valuable insight into what may happen in the future and its techniques include a variety of statistical and machine learning techniques, such as: neural networks, decision trees, and regression.
- Prescriptive analytics helps answer questions about what should be done. By using insights from predictive analytics, data-driven decisions can be made. This allows businesses to make informed decisions in the face of uncertainty. Prescriptive analytics techniques rely on machine learning strategies that can find patterns in large datasets. By analysing past decisions and events, the likelihood of different outcomes can be estimated.

These types of data analytics provide the insight that businesses need to make effective and efficient decisions. Used in combination they provide a well-rounded understanding of a company's needs and opportunities. The primary goal of a data analyst is to increase efficiency and improve performance by discovering patterns in data. The work of a data analyst involves working with data throughout the data analysis pipeline. This means working with data in various ways. The primary steps in the data analytics process are data mining, data management, statistical analysis, and data presentation. The

importance and balance of these steps depend on the data being used and the goal of the analysis.

Data mining is an essential process for many data analytics tasks. This involves extracting data from unstructured data sources. These may include written text, large complex databases, or raw sensor data. The key steps in this process are to extract, transform, and load data (often called ETL.) These steps convert raw data into a useful and manageable format. This prepares data for storage and analysis. Data mining is generally the most time-intensive step in the data analysis pipeline.

Data management or data warehousing is another key aspect of a data analyst's job. Data warehousing involves designing and implementing databases that allow easy access to the results of data mining. This step generally involves creating and managing SQL databases. Non-relational and NoSQL databases are becoming more common as well.

Statistical analysis allows analysts to create insights from data. Both statistics and machine learning techniques are used to analyse data. Big data is used to create statistical models that reveal trends in data. These models can then be applied to new data to make predictions and inform decision making. Statistical programming languages such as R or Python (with pandas) are essential to this process. In addition, open-source libraries and packages such as TensorFlow enable advanced analysis.

The final step in most data analytics processes is data presentation. This step allows insights to be shared with stakeholders. Data visualization is often the most important tool in data presentation. Compelling visualizations can help tell the story in the data which may help executives and managers understand the importance of these insights.

4.2.2 MATCH PREDICTION:

The next part of the project is the prediction part where both the Pre toss and Post toss prediction is done using the Supervised machine learning algorithms such as Multiple Linear Regression and Random Forest Classifier algorithm.

Multiple Linear Regression: It's a form of linear regression that is used when there are two or more predictors. It is the most common form of linear regression analysis. As a predictive analysis, the multiple linear regression is used to explain the relationship between one continuous dependent variable and two or more independent variables. The independent variables can be continuous or categorical

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p$$

Here, Y is the output variable, and X terms are the corresponding input variables. Notice that this equation is just an extension of Simple Linear Regression, and each predictor has a corresponding slope coefficient (β).

The first β term (β_0) is the intercept constant and is the value of Y in absence of all predictors (i.e., when all X terms are 0). It may or may not hold any significance in a given regression problem. It's generally there to give a relevant nudge to the line/plane of regression.

Regression residuals must be normally distributed. A linear relationship is assumed between the dependent variable and the independent variables. The residuals are homoscedastic and approximately rectangular-shaped. Absence of multicollinearity is assumed in the model, meaning that the independent variables are not too highly correlated. At the centre of the multiple linear regression analysis is the task of fitting a single line through a scatter plot. More specifically the multiple linear regression fits a line through a multi-dimensional space of data points. The simplest form has one dependent and two independent variables. The dependent variable may also be referred to as the outcome variable or regressand. The independent variables may also be referred to as the predictor variables or regressors.

There are 3 major uses for multiple linear regression analysis. First, it might be used to identify the strength of the effect that the independent variables have on a dependent variable. Second, it can be used to forecast effects or impacts of changes. That is, multiple linear regression analysis helps us to understand how much will the dependent variable change when we change the independent variables. Third, multiple linear

regression analysis predicts trends and future values. The multiple linear regression analysis can be used to get point estimates. When selecting the model for the multiple linear regression analysis, another important consideration is the model fit. Adding independent variables to a multiple linear regression model will always increase the amount of explained variance in the dependent variable (typically expressed as R^2). Therefore, adding too many independent variables without any theoretical justification may result in an over-fit model.

Using Multiple Linear Regression in this project, the outcome of a match is predicted two times. Once, before the toss, without taking into consideration the toss decision (Pre-Toss). The model takes in the Team name as input and create a linear regression model (team names are encoded), to give the output of the prediction. On the other hand, the Post-Toss takes other factors like toss winner and toss decision into consideration for predicting the match outcome.

Random Forest Algorithm: Random forest algorithm is a flexible machine learning algorithm that produces great results even without hyper-parameter tuning. Apart from being simple to use, it is extremely accurate also. It is basically a supervised learning algorithm. A large number of decision trees operate together; an individual tree in a random forest model gives some prediction and finally the one with most votes becomes the prediction of the model. Random forest can be used for both classification as well as regression problems. A large number of relatively uncorrelated trees operating as a committee will perform better than individual constituent tree. Uncorrelated trees can produce better and more accurate results than individual decision trees, the reason for this is that the trees protect each other from their individual errors, thus even if some trees in the group are wrong, all the trees are able to move in correct direction given that many other trees will be right.

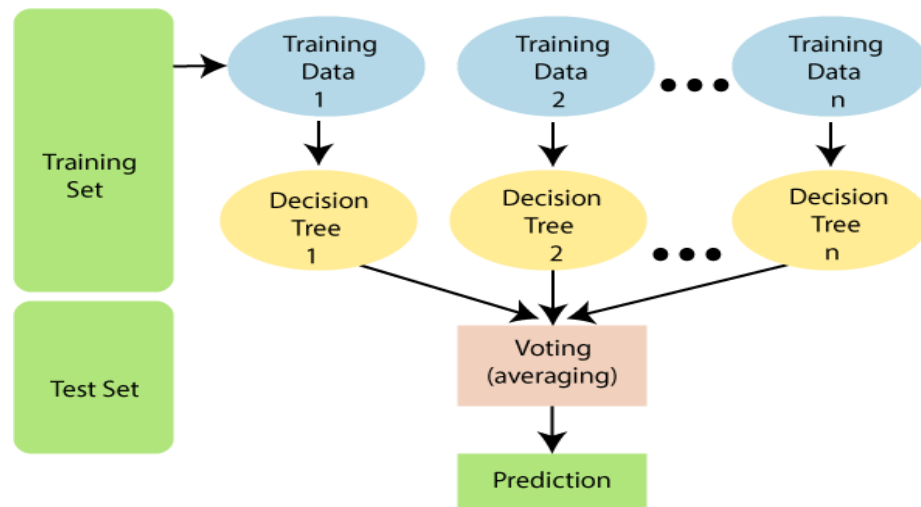


Figure 4.2: Random Forest Classifier

Random Forest works in four steps:

- Select random samples from a given dataset.
- Construct a decision tree for each sample and get a prediction result from each decision tree.
- Perform a vote for each predicted result.
- Select the prediction result with the most votes as the final prediction.

Decision Trees:

The classification technique is a systematic approach to build classification models from an input dataset. For example, decision tree classifiers, rule-based classifiers, neural networks, support vector machines, and naive Bayes classifiers are different technique to solve a classification problem. Each technique adopts a learning algorithm to identify a model that best fits the relationship between the attribute set and class label of the input data. Therefore, a key objective of the learning algorithm is to build predictive model that accurately predict the class labels of previously unknown records.

Decision Tree Classifier is a simple and widely used classification technique. It applies a straightforward idea to solve the classification problem. Decision Tree Classifier poses a series of carefully crafted questions about the attributes of the test record. Each time it receives an answer, a follow-up question is asked until a conclusion about the class label of the record is reached.

Build an optimal decision tree is key problem in decision tree classifier. In general, many decision trees can be constructed from a given set of attributes. While some of the trees are more accurate than others, finding the optimal tree is computationally infeasible because of the exponential size of the search space.

However, various efficient algorithms have been developed to construct a reasonably accurate, albeit suboptimal, decision tree in a reasonable amount of time. These algorithms usually employ a greedy strategy that grows a decision tree by making a series of locally optimum decisions about which attribute to use for partitioning the data. For example, Hunt's algorithm, ID3, C4.5, CART, SPRINT are greedy decision tree induction algorithms.

The decision tree inducing algorithm must provide a method for specifying the test condition for different attribute types as well as an objective measure for evaluating the goodness of each test condition.

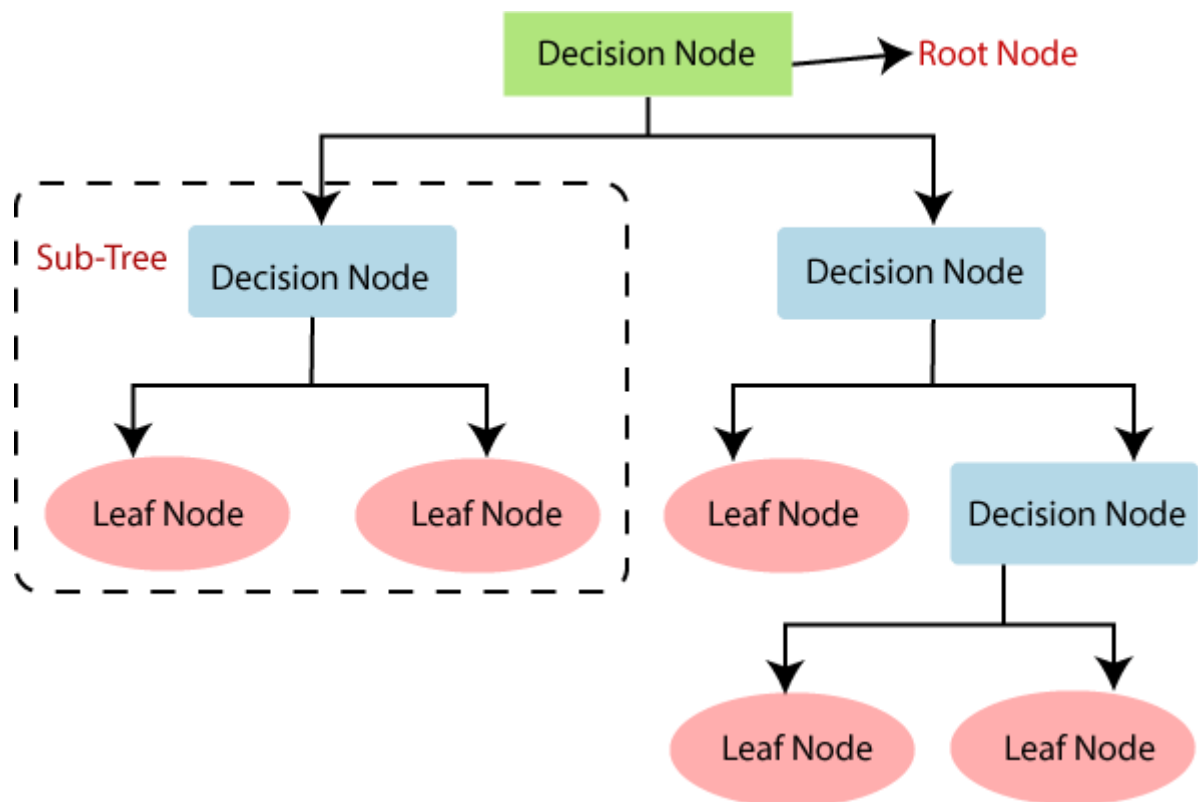


Figure 4.3:Decision Tree

First, the specification of an attribute test condition and its corresponding outcomes depends on the attribute types. We can do two-way split or multi-way split, discretize or group attribute values as needed. The binary attributes lead to two-way split test condition. For nominal attributes which have many values, the test condition can be expressed into multi way split on each distinct value, or two-way split by grouping the attribute values into two subsets. Similarly, the ordinal attributes can also produce binary or multi way splits as long as the grouping does not violate the order property of the attribute values. For continuous attributes, the test condition can be expressed as a comparison test with two outcomes, or a range query. Or we can discretize the continuous value into nominal attribute and then perform two-way or multi-way split.

Since there are many choices to specify the test conditions from the given training set, we need use a measurement to determine the best way to split the records. The goal of best test conditions is whether it leads a homogenous class distribution in the nodes, which is the purity of the child nodes before and after splitting. The larger the degree of purity, the better is the class distribution.

To determine how well a test condition performs, we need to compare the degree of impurity of the parent before splitting with degree of the impurity of the child nodes after splitting. The larger their difference, the better is the test condition. The measurements of node impurity/purity are:

- Gini Index
- Entropy
- Misclassification Error

$$\text{Gini index: } G(E) = 1 - \sum_{j=1}^C p_j^2,$$

$$\text{entropy: } H(E) = - \sum_{j=1}^C p_j \log p_j.$$

In this project, the outcome of a match is predicted two times. Once, before the toss, without taking into consideration the toss decision (Pre-Toss). The model (Random

Forest Classifier) takes in the Team name as input and creates an ensemble of decision trees usually trained with “bagging” method, to give the output of the prediction. On the other hand, the Post-Toss takes other factors like toss winner and toss decision into consideration for predicting the match outcome in a more accurate fashion.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 DATA ANALYTICS:

This paper focuses on predicting the outcome of an IPL match by taking factors like Toss, Toss Decision into consideration along with Data analytics and Visualization of teams and players.

Efficient prediction accuracy of about 84% is achieved in this model with the help of Random Forest algorithm.

All the results and outcomes of the project are hosted in a web application that is user friendly and can run on any web browser.

Welcome to IPL Predictions

Select Model

Select

⚙ Usage Heirarchy

- **IPL Data Analytics:**

- Team Stats
- Player Stats:
 - Batsman Stats
 - Bowler Stats

- **Pre Toss Prediction**

- In this section the ML model will predict Pre-Toss Sims between the selected teams

- **Post Toss Prediction**

- In this section the ML model will predict Post-Toss Sims between the selected teams

Figure 5.1: Welcome Page

Figure 5.1 represents the Home Page of the web application that can be used by the user for checking the outcome of a particular match as well as visualizing the team stats and player stats.

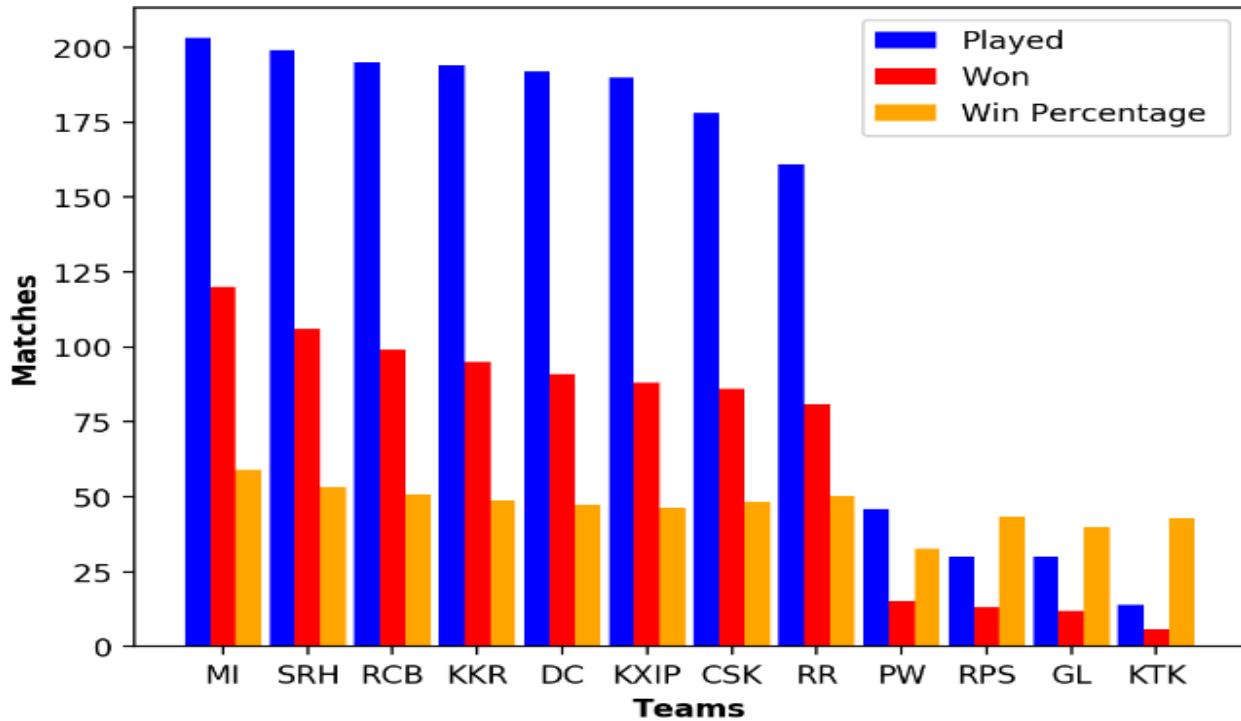


Figure 5.2:Teamwise Performance

The Figure 5.2 represents the teamwise analysis with number of matches played, matches won and win percentage of each team in the Y-Axis against the Team names in the X-Axis.

Teamwise analysis is very important when it comes to any team sports. The same is true for IPL. Here, through this analysis we can see that MI is the most successful team in IPL. It has played the most no of matches throughout the IPL. The yellow bar represents that MI has the highest win percentage as well. Similarly, Kochi Tuskers Kerala have played the least matches in IPL, this data is also gives us this insight.

Toss winners' percentage of Winning the match

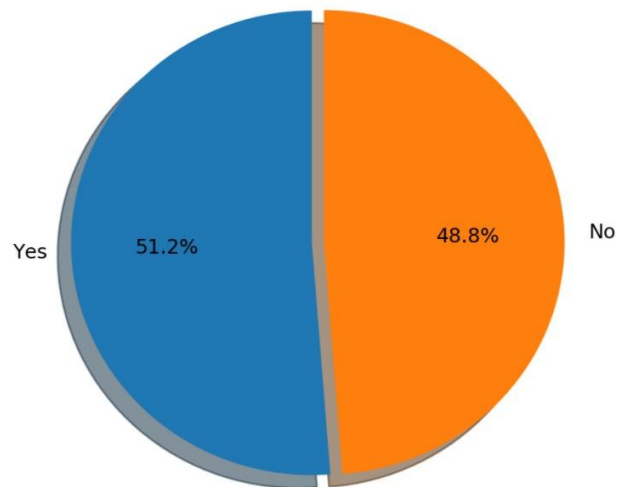


Figure 5.3:Impact on toss

The Figure 5.3 represents teams that win the toss has 51.2% record of winning the match whereas teams that lose the toss has 48.8% record of winning the match since IPL 2008.

Toss decision impact on Winning the match

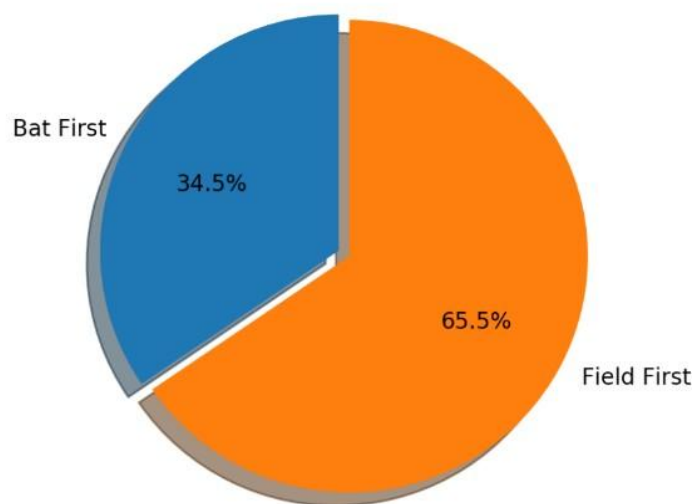


Figure 5.4:Impact of toss decision

The Figure 5.4 represents teams that win the toss and elect to bat first has 34.5% record of winning the match whereas teams that win the toss elect to field has 65.5% record of winning the match since IPL 2008.

Toss or flip of the coin is one of the most important factors in a cricket match. Unlike other sports Toss plays a huge role in determining the final outcome of the match. Toss is so important that sometimes the result of whole game is depending upon the Toss and the team that wins the toss wins the match as well(provided that the captain made the correct decision after winning the toss)

The teams mostly choose the option that is best suited to them (unless the pitch conditions are entirely different) after winning the toss. For example, a team whose strength lies in batting will opt to bowl first after winning the toss most of the times. If the team has a destructive bowling line-up then the toss can be decisive factor in the match.

V Kohli's Runs scored

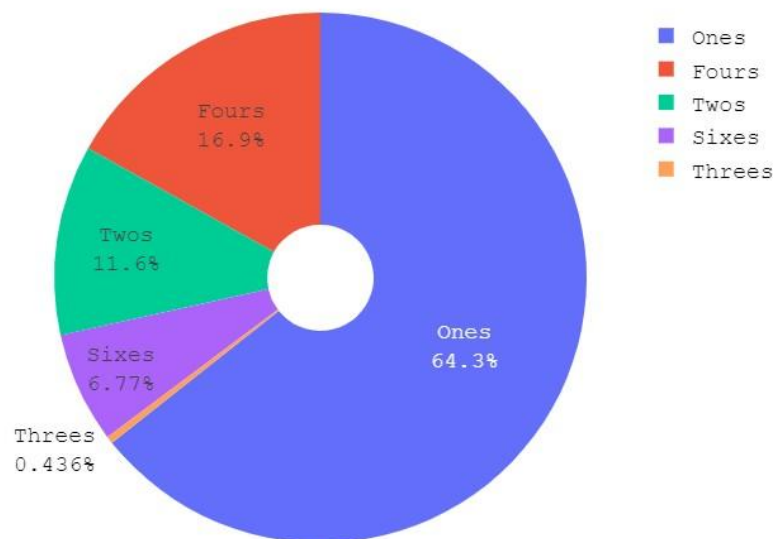


Figure 5.5:Runs split of a batsman

The Figure 5.5 represents the runs split of a particular batsman throughout his IPL career (till 2020). The example mentioned here represents the runs split of V Kohli from 2008 to 2020.

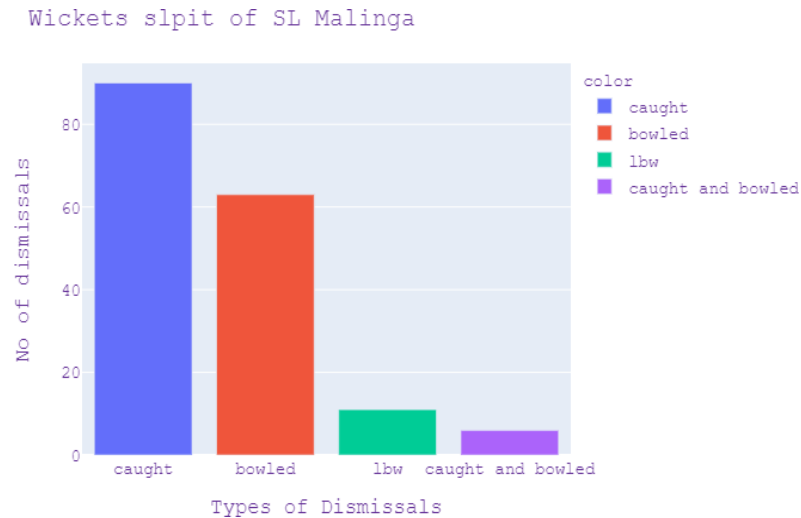


Figure 5.6:Wickets split of a bowler

The Figure 5.6 represents the wickets split of a particular bowler throughout his IPL career (till 2020). The example mentioned here represents the wickets split of SL Malinga from 2008 to 2020.

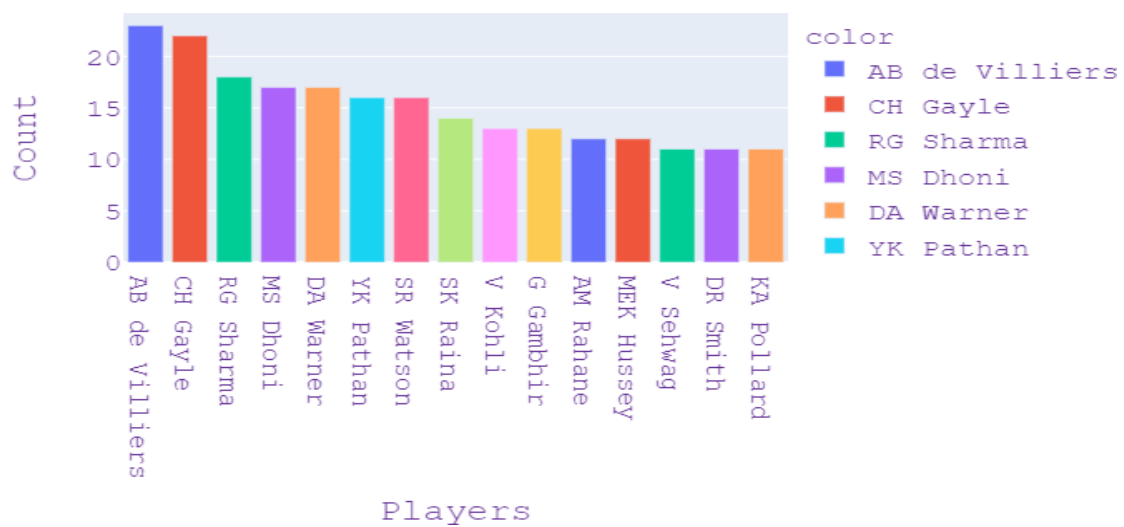


Figure 5.7:Most Man of the Match awards

The Figure 5.7 represents Most man of the match awards received by players throughout their IPL career (till 2020). The example mentioned here represents first 15 players from 2008 to 2020.

5.2 MATCH PREDICTION:

Pre toss and Post toss prediction is done using the Supervised machine learning algorithm known as Random Forest Classifier algorithm. even though the toss plays a huge role and affects the results of the game however the bowlers and the batsmen have to perform well, because if they don't then winning or losing the toss doesn't make any difference.

Pre-Toss Sims

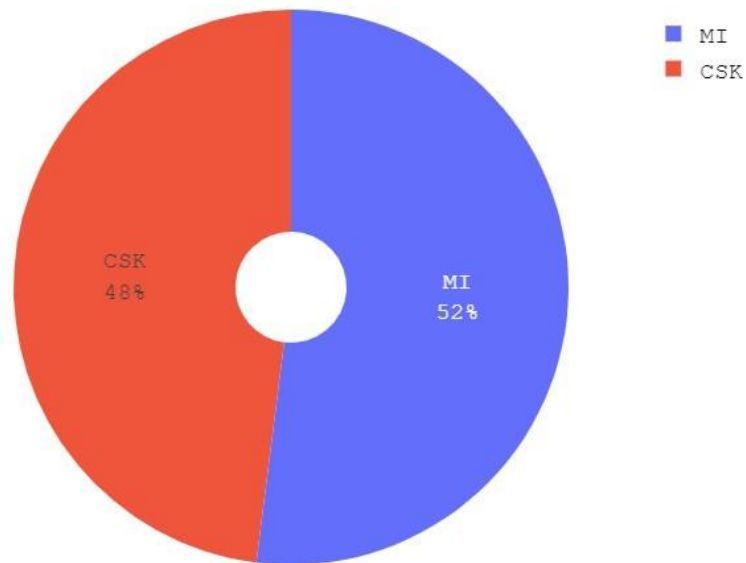


Figure 5.8:Pre-Toss Prediction

Figure 5.8 represents the simulation before the toss happens. In this particular example, Mumbai Indians (MI) has a winning chance of 52% whereas Chennai Super Kings (CSK) has a winning chance of 48%.

Post-Toss Sims

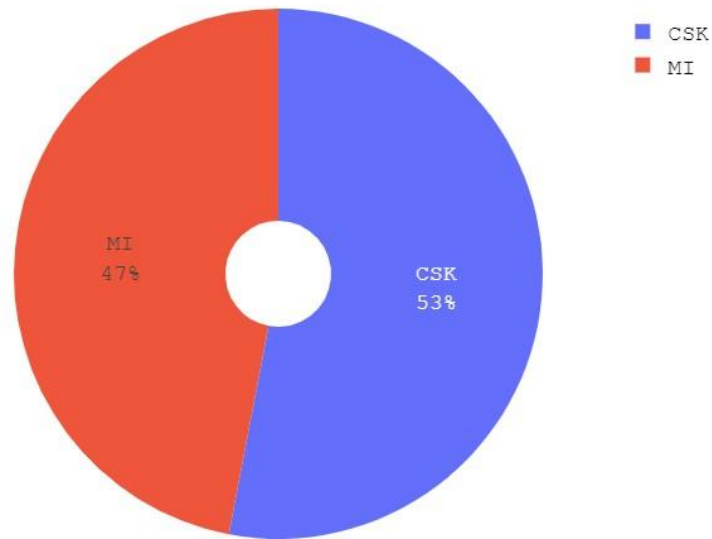


Figure 5.9:Post-Toss Prediction

Figure 5.9 represents the simulation after the toss happens. In this particular example, Chennai Super Kings (CSK) has won the toss and elected to field first, thus has a winning chance of 53% whereas Mumbai Indians (MI) has a winning chance of 47%, by batting first.

All the results and outcomes of the project are hosted in a web application that is user friendly and can run on any web browser.

CHAPTER 6

SUMMARY AND CONCLUSION

6.1 CONCLUSION:

Statistical Modelling and Data Mining tools are being used in Sports Analytics and prediction vividly now a days. This gives us an opportunity to analyse and predict the outcome of a game (like – Indian Premier League) using different visualization tools and machine learning algorithms. This paper focuses on predicting the outcome of an IPL match by taking factors like Toss, Toss Decision into consideration along with Data analytics and Visualization of teams and players. To conduct the analysis and predicting the winner of IPL various branches of Data Science has been converged including Pre-Processing of data, Visualizations of data, preparation of data, feature selection and implementing different machine learning models for the predictions. SEMMA methodology has been selected for conducting the analysis of IPL T20 match winner dataset. Pre-processing has been done on the dataset to make it consistent by removing missing value, encoding variables into numerical format. Best features were selected by visualizing attributes of data with target variable. On selected features several machine learning models has been applied on the to predict the winner and the results were outstanding.

First of all, after the data is cleaned and pre-processed, that data is used to do different data visualization like Team Statistics, Batsman Statistics, Bowler Statistics. The user gets to use the webpage to access any kind of data they need for IPL. The Data Analysis part is important as it gives insights about the data generated by Indian Premier League. The second part of the project deals with the prediction of the outcome of a match based on factors like previous win record, toss result, toss decision. Firstly, Multiple Linear Regression was used to predict the outcome of a particular match. Multiple linear regression (MLR), also known simply as multiple regression, is a statistical technique that uses several explanatory variables to predict the outcome of a response variable. The goal of multiple linear regression (MLR) is to model the linear relationship between the explanatory (independent) variables and response (dependent) variable. After using Multiple Linear Regression, the accuracy turned out to be around 30%, which was not good enough. Then we applied Random Forest model on the selected features and the predicted

the winner with 65% accuracy which was not good enough, so Random Forest Model was also tuned by parameter's tuning and results got better with 73 % accuracy.

Models	Accuracy
Multiple Linear Regression	30%
Random Forest Classifier	65%
Random Forest Classifier (Tuned)	73%

Thus finally, both the modules of this project Data Analysis and Outcome Prediction perform well and serve the objective it was supposed to.

This project is deployed in Heroku cloud platform [\[check it out here\]](#).

APPENDIX

A) SOURCE CODE:

```
// Posttoss.py

import streamlit as st

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

import pandas as pd

import plotly.express as px

import plotly.graph_objects as go

import random

import math

#@st.cache(suppress_st_warning=True)

def posttoss(t1,t2,tw,td):

    old_matches = pd.read_csv('matches.csv')

    #old_matches

    print("Data Frame read")

    sample1=old_matches.drop(['id','season','city','date','result','dl_applied','win_by_runs','win_by_wickets','player_of_match','venue','umpire1','umpire2','umpire3'],axis=1)

    #sample1

    print("dropping other rows")

    x=['Sunrisers Hyderabad', 'Mumbai Indians', 'Gujarat Lions',

        'Rising Pune Supergiant', 'Royal Challengers Bangalore',
```

```

    'Kolkata Knight Riders', 'Delhi Daredevils', 'Kings XI Punjab',
    'Chennai Super Kings', 'Rajasthan Royals', 'Deccan Chargers',
    'Kochi Tuskers Kerala', 'Pune Warriors', 'Rising Pune Supergiants', 'Delhi Capitals']

y =
['SRH','MI','GL','RPS','RCB','KKR','DC','KXIP','CSK','RR','SRH1','KTK','PW','RPS','DC']

sample1.replace(x,y,inplace = True)

#sample1

#sample

print("Renamed Teams")

sample1=sample1.dropna()

#sample1

#sample

sample1 = sample1[sample1.team1 != 'KTK']

sample1 = sample1[sample1.team1 != 'RPS']

sample1 = sample1[sample1.team1 != 'PW']

sample1 = sample1[sample1.team1 != 'GL']

sample1 = sample1[sample1.team1 != 'SRH1']

#sample1 = sample1[sample1.team1 != 'DC1']#KTK RPS PW GL

#sample1

sample1 = sample1[sample1.team2 != 'KTK']

sample1 = sample1[sample1.team2 != 'RPS']

sample1 = sample1[sample1.team2 != 'PW']

sample1 = sample1[sample1.team2 != 'GL']

sample1 = sample1[sample1.team2 != 'SRH1']

```

```

#sample1 = sample1[sample1.team2 != 'DC1']#KTK RPS PW GL

#sample1

print("Removed non existing teams")

fromsklearn.model_selection import train_test_split

fromsklearn.ensemble import RandomForestClassifier

fromsklearn.datasets import make_classification

sampl = pd.get_dummies(sample1, prefix=['Team_1', 'Team_2','toss_won','toss_dec'],
columns=['team1', 'team2','toss_winner','toss_decision'])

#sampl

X = sampl.drop(['winner'], axis=1)

y = sampl["winner"]

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=random.uniform(0.2,0.3),
random_state=42)

#X_train

rf1 = RandomForestClassifier(n_estimators=220, max_depth=40,oob_score=True
,class_weight='balanced',verbose=2,n_jobs=-1,random_state=62)

rf1.fit(X_train, y_train)

score = rf1.score(X_train, y_train)

scoree2 = rf1.score(X_test, y_test)

print(score)

print(scoree2)

#st.write(scoree2)

#print(rf1.oob_score_)

copy3=pd.read_csv('copytry.csv')

```

```

#copy3

x=['Sunrisershyderabad', 'Mumbai indians', 'Gujarat Lions',
  'Rising Pune Supergiant', 'Royal challengers bangalore',
  'Kolkata knight riders', 'Delhi Daredevils', 'Kings xi punjab',
  'Chennai super kings', 'Rajasthan royals', 'Deccan Chargers',
  'Kochi Tuskers Kerala', 'Pune Warriors', 'Rising Pune Supergiants', 'Delhi
capitals','Royal challengers']

y =
['SRH','MI','GL','RPS','RCB','KKR','DD','KXIP','CSK','RR','DCR','KTK','PW','RPS','DC','RCB']

copy3.replace(x,y,inplace = True)

#copy3

et1=list(copy3['Team'])
et2=list(copy3['Team2'])
et3=list(copy3['toss_winner'])
et4=list(copy3['toss_decision'])

copyy = pd.get_dummies(copy3, prefix=['Team_1', 'Team_2','toss_won','toss_dec'],
columns=['Team', 'Team2','toss_winner','toss_decision'])

#copyy

predicts=rf1.predict(copyy)

#print(predicts)

for i in range(224):

    if(et1[i]==t1 and et2[i]==t2 and et3[i]==tw and et4[i]==td):

print(predicts[i])

winner_is=predicts[i]

```

```

if(t1!=winner_is):

looser_is=t1

else:

looser_is=t2

st.success(predicts[i]+" will win")

    #from sklearn.model_selection import cross_val_score

    #RF_accuracies = cross_val_score(estimator = rf1, X = X_test, y = y_test, cv = 9)
#5,7,9,15

    #RF_accuracy=RF_accuracies.max()

    #print(RF_accuracy)

    scoree2=scoree2*100

print(scoree2)

    k=math.ceil(scoree2)

fig = go.Figure(data=[go.Pie(labels=[winner_is,looser_is],
textinfo='label+percent',values=[k,100-k], hole=.2)])

fig.update_layout(height=600, title_text="Post-Toss Sims", font=dict(family='Courier New,
monospace', size=18, color='#000000'))

st.plotly_chart(fig,use_container_width=True)

    #posttoss('MI','RCB','MI','bat')


// test.py


f=st.selectbox("Player Type",['Select','BatsmanStats','Bowler Stats'])

if(f=='Batsman Stats'):

```

```

nm=['Player','VKohli', 'SK Raina', 'DA Warner', 'RG Sharma', 'S Dhawan', 'AB de
Villiers', 'CH Gayle', 'MS Dhoni', 'RV Uthappa',

'G Gambhir', 'AM Rahane', 'SR Watson', 'KD Karthik', 'AT Rayudu', 'MK Pandey',
'YK Pathan', 'KA Pollard', 'BB McCullum',

'PA Patel', 'Yuvraj Singh', 'V Sehwag', 'KL Rahul', 'M Vijay', 'SV Samson', 'SE
Marsh', 'JH Kallis', 'DR Smith', 'SR Tendulkar',

'SPD Smith', 'F du Plessis', 'SS Iyer', 'R Dravid', 'RA Jadeja', 'RR Pant', 'AC
Gilchrist', 'JP Duminy', 'SA Yadav', 'AJ Finch', 'WP Saha', 'MEK Hussey']

ssn=['All',2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020]

g=st.selectbox("Select player",nm)

#zxc=st.slider("Season",min_value=2008, max_value=2020, step=1)

if(g=='Player'):

st.markdown("## Select Player")

else:

zx=st.select_slider("Season",options=ssn)

st.write(" ")

if(zx=='All'):

    #st.write("under cover")

player_name=g

kj=allballs[(allballs['striker']==player_name) & (allballs['innings']<3)]

#dhh = pd.DataFrame({' Run type': 'ones dots fours twos sixes threes'.split(),
'Value': kj['runs_off_bat'].value_counts().values})

#st.write(dhh)

total=kj['runs_off_bat'].sum()

#st.write("Total runs: ",total)

```

```

#try_dff=['Total runs','Innings Played', 'Balls
Faced','Ones','Twos','Threes','Fours','Sixes','StrikeRate','Average']

inn=allballs[(allballs['striker']==player_name)](allballs['non_striker']==player_name)]

mp=len(inn["match_id"].unique())

#st.write("Innings Played: ", mp)

bfwe=len(allballs[(allballs['striker']==player_name) & (allballs['innings']<3) &
(allballs['wides']>0)])

bf=len(allballs[(allballs['striker']==player_name) & (allballs['innings']<3)])

#st.write("Balls Faced :",bf-bfwe)

ones=len(allballs[(allballs['striker']==player_name) & (allballs['runs_off_bat']==1) &
(allballs['innings']<3)])

#st.write("ones: ",ones)

twos=len(allballs[(allballs['striker']==player_name) & (allballs['runs_off_bat']==2) &
(allballs['innings']<3)])

#st.write("twos: ",twos)

threes=len(allballs[(allballs['striker']==player_name) & (allballs['runs_off_bat']==3) &
(allballs['innings']<3)])

#st.write("threes: ",threes)

fours=len(allballs[(allballs['striker']==player_name) & (allballs['runs_off_bat']==4) &
(allballs['innings']<3)])

#st.write("fours: ",fours)

sixes=len(allballs[(allballs['striker']==player_name) & (allballs['runs_off_bat']==6) &
(allballs['innings']<3)])

#st.write("sixes: ",sixes)

#print("SR: ",(total/(bf-bfwe))*100)

```

```

#st.write("SR: ",(total/(bf-bfwe))*100)

out=len(allballs[((allballs['striker']==player_name)|(allballs['non_striker']==player_name)) &
(allballs['player_dismissed']==player_name) & (allballs['innings']<3)])

#print("Dismissed: ",out)

#st.write("Avg: ",total/(out))

#print(try_df)

#print(try_dff)

#st.write(try_df)

#st.write(try_dff)

#st.table(out)

lk2={'Total runs': total, 'Innings Played': mp, 'Balls Faced':bf-bfwe, 'Ones':ones,
'Twos':twos, 'Threes':threes, 'Fours':fours, 'Sixes':sixes, 'Strike Rate':((total/(bf-bfwe))*100),
'Average':(total/(out))}

dcv=pd.DataFrame(lk2,index=[0],dtype=float)

st.table(dcv)

#st.dataframe(dcv)

st.write(" ")

ssn1=[2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018,
2019, 2020]

scli=[]

fori in ssn1:

    kj1=allballs[(allballs['striker']==g) & (allballs['innings']<3) &
(allballs['season']==i)]

    scli.append(kj1['runs_off_bat'].sum())

#print(scli)

```

```
fig = go.Figure(data=go.Scatter(x=ssn1, y=scli,line_color='rgb(0,100,80)'))  
fig.update_layout(title="Runs split of "+g,axis_title="Season",yaxis_title="No of Runs",  
font=dict(family="Courier New, monospace",size=18,color="black"))
```

References:

- [1]. Daniel MagoVistro, Faizan Rasheed, Leo Gertrude David, "The Cricket Winner Prediction With Application of Machine Learning And Data Analytics" International Journal of Scientific & Technology Research (2019)
- [2]. Madan Gopal Jhanwar and VikramPudi, "Predicting the Outcome of ODI Cricket Matches: A Team Composition Based Approach" International Institution of Information Technology (2017)
- [3]. I. P. Wickramasingheet. al, "Predicting the performance of batsmen in test cricket," Journal of Human Sport & Exercise", vol. 9, no. 4, pp. (2017)
- [4]. R. P. Schumaker, O. K. Solieman and H. Chen, "Predictive Modeling for Sports and Gaming" in Sports Data Mining, vol. 26, Boston, Massachusetts: Springer, (2016)
- [5]. J. McCullagh, "Data Mining in Sport: A Neural Network Approach," International Journal of Sports Science and Engineering, vol. 4, no. 3 (2016)
- [6]. Bunker, Rory &Thabtah, Fadi. "A Machine Learning Framework for Sport Result Prediction. Applied Computing and Informatics". (2017)
- [7] Kulkarni, V. & Sinha, P., n.d. Effective Learning and Classification using Random Forest Algorithm. International Journal of Engineering and Innovative Technology (IJEIT).
- [8] Lokhande, A., Chawan, R. & Pramila, S., 2018. Prediction of Live Cricket Score and Winning. Computer and IT Dept, VeermataJeejabai Technological Institute, Mumbai, India, 5(4)(2394-9333).
- [9] Mitchel, M. T., 1997. Machine learning. Burr Ridge, IL: McGraw Hill, 45, 1997.
- [10] Murphy, K. P., 2006. Naive bayes classifiers. University of British Columbia.
- [11] Nasteski&Vladmir, 2007. An Overview of the Supervised Machine Learning Methods. Faculty of Information and Technology. Faculty of Information and communication Technologies.
- [12] Available at: <https://medium.com/machine-learning-101/chapter2-svm-support-vector-machine-theory-f0812effc72>
- [13] Shah, P. & Shah, M., 2015. Predicting ODI Cricket Result. ISSN (Paper) 2312-5187 ISSN (Online) 2312-5179 An International Peer-reviewed Journal, Volume 5.

- [14] Asare-Frempong, J. and Jayabalan, M., 2017. Predicting customer response to bank direct telemarketing campaign. In 2017 International Conference on Engineering Technology and Technopreneurship (ICE2T) (pp. 1-4). IEEE.
- [15] Yasir, M. et al., 2017. Ongoing Match Prediction in T20 International. IJCSNS International Journal of Computer Science and Network Security.
- [16] Python <https://www.python.org/>