MACHINE LEARNING PROJECT

FINAL REPORT

INDIAN PREMIERE LEAGUE (IPL) WIN PREDICTION USING MACHINE LEARNING METHODS

TEAM:3

SAI VASAVI PONNADA (1321765)

JOSEPH NIKHIL REDDY YERUVA (1321204)

TARUN SAI KRISHNA NARRA (1216062)

INTRODUCTION:

One of the most well-known and lucrative T20 cricket leagues in the world is the Indian Premier League (IPL), a professional competition. IPL is one of India's most important sports industries, attracting millions of fans annually. In recent years, the use of data and analytics has become increasingly important in the world of sports. With the advent of big data and machine learning techniques, teams and organizations can analyze player and team performance in real-time to make strategic decisions. This project demonstrates how data analysis and machine learning algorithms can be utilized in the sports industry to predict the outcome of matches and gain a competitive edge. By analyzing the IPL data, we can gain insights into team performance, player performance, and factors that influence the outcome of matches. This information can be used by teams, coaches, and analysts to make data-driven decisions and improve their chances of success. In this project, we have chosen to examine IPL data and try to forecast the winning team using the information at hand. This report's objective is to share our study of the IPL data and show how we used several machine learning algorithms to forecast which team will win.

In addition to being an exciting sporting event, the IPL also represents a massive commercial opportunity for sponsors, broadcasters, and advertisers. With a vast audience and high levels of engagement, the IPL presents an excellent platform for businesses to reach out to potential customers and build brand awareness. Therefore, accurately predicting the outcome of IPL matches can have a significant impact on the financial success of these businesses.

Moreover, the IPL is an excellent example of how technology and data analytics are transforming the sports industry. In recent years, the use of data analytics, machine learning, and artificial intelligence has become increasingly common in various sports. These technologies enable teams, coaches, and analysts to make data-driven decisions, improve player performance, and gain a competitive edge. This project demonstrates how machine learning algorithms can be used to analyze the IPL data and predict the outcome of matches.

Overall, this project highlights the importance of data analysis and machine learning techniques in the sports industry. By leveraging the available data and applying various machine learning algorithms, we can gain insights into team performance, player performance, and factors that impact the outcome of matches. These insights can be used by teams, coaches, and analysts to make strategic decisions and improve their chances of success.

METHODOLOGY:

DATA COLLECTION:

To carry out our analysis, we collected IPL data from Kaggle. The dataset includes the season played, toss win and opted, city played, venue, state, player of the match, win by runs, win by wickets, umpire, teams, result of the match, year played for all the IPL seasons from 2008 to 2019. We are using the dataset from Kaggle- IPL _Data_Set | Kaggle

DATA CLEANING:

The raw data we collected needed cleaning and preprocessing to make it suitable for analysis. We removed missing values, duplicates, and irrelevant columns from the dataset. We also performed feature engineering to create new features that could improve the accuracy of our predictions. We then split the data into training and testing sets.

DATA PREPROCESSING:

We have already taken the pre-processed data but we have made changes to the dataset like replacing the previous name of the teams with new team and also, we have pre preprocessed data separately for 1st innings and 2nd innings and created new data frames as per the requirements. We have also included feature engineering for specific fields. We also performed feature scaling according to the prediction requirements like current run rate and required run rate.

BASELINE:

In the context of the IPL match prediction project, the baseline model could be predicting the winner of the match to be the team that has won the most matches in the past. The baseline accuracy can then be calculated and compared with the accuracy of the machine learning models developed during the project to determine their effectiveness.

MODEL DESCRIPTION:

In this project, it is a binary classification model that predicts the outcome of a cricket match based on various features such as the current score, runs left, balls left, wickets left, batting team, bowling team, and city. Different algorithms, logistic regression, and random forest classifiers, XG boost were used to build the model. We also performed hyper-parameter training for this model. The

performance of the models was evaluated using accuracy score as the evaluation metric. The project includes data preprocessing steps such as handling missing values, feature engineering, and one-hot encoding of categorical variables. The project aims to explore the factors that influence the outcome of a cricket match and build a predictive model that can accurately predict the winner of a cricket match.

IMPLEMENTATION:

The below are the implementations we have done:

Data gathering, Data preprocessing, Data visualization, Model selection, Model training, Model evaluation, Hyper-parameter training, Model deployment.

COMPUTATIONAL REQUIREMENTS:

Since the project involves data preprocessing steps such as handling missing values, feature engineering, and one-hot encoding of categorical variables, the computational requirements for these steps would depend on the size of the dataset. As the dataset is large, it required a significant amount of computing power to preprocess the data.For the machine learning models used in the project, such as logistic regression, random forest classifiers, and XGBoost, the computational requirements depended on the complexity of the models and the size of the dataset. Training and optimizing these models required a significant amount of computing power and memory.

Additionally, hyperparameter tuning for the models required additional computational resources. The computational requirements for hyperparameter tuning depended on the number of hyperparameters being tuned and the size of the parameter search space.

REFERENCES:

A Gentle Introduction to XGBoost for Applied Machine Learning - MachineLearningMastery.com

What is Random Forest? | IBM

What is Logistic regression? | IBM

sklearn.model_selection.GridSearchCV — scikit-learn 1.2.2 documentation

Build Machine Learning Pipeline Using Scikit Learn (analyticsvidhya.com)