Analyzing the Performance of IPL Teams and Predicting IPL Scores and League Rankings using Data Science Methods

Project Report for IITB DS 203, 2021

Shreedhar Malpani
Dept. of Mechanical Engineering
Indian Institute of Technology Bombay
Mumbai, Maharashtra
200020132@iitb.ac.in

Sumit Kumar
Dept. of Chemical Engineering
Indian Institute of Technology Bombay
Mumbai, Maharashtra
200020145@iitb.ac.in

Pranav Singla
Dept. of Mechanical Engineering
Indian Institute of Technology Bombay
Mumbai, Maharashtra
200040102@iitb.ac.in

Om Vishal Mihani
Dept. of Chemical Engineering
Indian Institute of Technology Bombay
Mumbai, Maharashtra
200020085@iitb.ac.in

***Abstract*—Abstract—Fantasy cricket is an online cricket team-building game in which one makes a virtual team of real cricket players based on their and the team's past performance and the points are scored depending on how those players perform in real-life matches. Fantasy cricket relies on Data Analysis rather than luck. One can score more points learning from past experiences, analyzing and basing teams on the Data gathered. With Data Science methods, one can effectively identify which team has a better chance of winning and is more likely to make more runs, leaving a bigger target to chase. Fantasy Cricket has gained a lot of traction in recent years with the introduction of new platforms like- Dream11, My circle 11, Gamezy, etc. On these platforms IPL- Indian Premier League is one of the biggest events and a major deciding factor, at the center of this whole charade.**

I. Introduction

The Indian Premier League(IPL) is a famous franchise based on T20 format. It is played between 8 teams representing different Indian states. It is organised by Board of Control for Cricket in India (BCCI). IPL is one of the most attended cricket leagues in the world. In the year of 2014 it was at the sixth position in terms of average attendance. Popularity of Indian Premier League can be estimated from the fact that it was the first sporting event to have a live broadcast on YouTube (2010). The brand value of Indian Premier League was reported a humongous ₹47,500 Crores in 2019. According to Board of Control for Cricket in India(BCCI), Indian Premier league contributed a total of ₹1150 Crores to GDP Of India in 2019. In 2020, IPL set a staggering record of viewership, a total of 31.57 million viewers clocked in with an overall consumption increase of 23% with respect to the year 2019. In Recent Years the market of fantasy cricket is growing in India with an appreciable increasing rate. The startup Dream 11 and Indian Premier League, which started with lots of scrutiny have developed from uncertain business models to successful startups. Every Cricket Fan knows about them and many famous players and celebrities have become ambassadors of different teams and played major roles in advertisement of these companies. Their advertisement Campaign dialogues are too famous that they are used by peoples in daily jokes and memes. When someone feeling tired due to work, you may not surprised when you hear that his friend telling him the famous dialogue of Dream 11 app "Ye mai karlunga, app Dream11 par team banao". Hence one can easily see how much success and popularity these startups have achieved in their business. Fantasy League has come a long way and predicting wins based on chance is not an option in such a competitive environment and hence players rely heavily on Data Analysis to get the most accurate prediction. We use random forest classifier and decision tree classifier to get the best model for prediction with the help of Mean absolute error, mean squared error, Root Mean Square. We also predict using our model what will be the runs at the end if user provide us wickets and total run at that point.

II. Background

- To understand our work one should understand T-20 cricket format. In this format of cricket, there are 20 over as its name suggests. The first 6 overs are of power play. Most of the runs of the inning came from the first and last over of the match so these were crucial for the match that team that has a good batting attack has an

advantage over a team which not have a good batting attack in these overs. The winning probability of a match depends on the toss meaning because if a team win a toss then it has independence on the decision to choose batting or bowling first so so the toss winning team can use its statics of previous matches on that ground against the opponent team to decide bowling or batting first against that team. This gave an extra advantage to the toss-winning team. From this, it is very much clear that the match-winning of a team depends on its toss winning, the ground, the City, opponent team, and while team bats first or bowl first?

- One should have a basic understanding of Python 3.0 and Google Colab. One should be familiar with the standard python libraries such as NumPy, SciPy, Matplotlib, skLearn, Seaborn, Pandas. We use Exploratory Data Analysis in the former part to get an understanding of the datasets used. One should also have a conceptual understanding of Statistics and Probability and how to read graphs such as Bar graphs, Pie-charts and Histograms. Once the Exploratory and Descriptive Data Analysis part is done we use models to predict the outcome of different match combinations using Classification with the help of Decision Tree Classifier and Random Forest Classifier. Hence one should have an understanding of how these models work. This can be explained with the help of branching and trees as the name of the classifiers suggests.

- Decision Tree classifier is a part of supervised Machine Learning where predetermined and well-defined steps are applied on inputs with the help of Linear Algebra to determine the outcome. However often a linear variation of input(from data-sets) is often not enough to encompass the desired output (or the target variable) to a high precision. Hence in our model we also use Random Forest Classifier which with the help of weights is able to include a variety of decision trees to predict the target variable. This process of combining the output of multiple individual models (also known as weak learners) is called Ensemble Learning. We also use regression for numerical data analysis in the latter part using SVM linear regressor and Random forest regressor. A random Forest Regressor works like a Random Forest Classifier. The main difference is that a classifier is used in classification problems when the data after data cleaning is of Categorical type and hence cannot be numerically added, compared or dealt with in contrast to a regressor which is used for numerical, integral, or continuous data analysis. A Support Vector Machine is ideally used for classification problems but the same concept is applied in the SVM regressor which with the help of hyper-planes (a linear separation between output data) by introducing extra dimensions is used to created decision boundaries using hyper-parameters.

III. Dataset & Feature Engineering

The dataset “IPL Matches 2008-2020.csv” is taken from Kaggle. It contains data about Indian Premier League matches played from 20008 to 2020. It has a total of 816 rows and 17 columns. 17 columns are named as follows: ‘id’, ‘city’, ‘date’, ‘player of the match’, “venue”, “neutral venue”, “team1”, “team2”, “toss winner”, “toss decision”, “winner”, “result”, “result margin”, “eliminator”, “method”, “umpire1”, “umpire2”. The meaning of these columns are following: id - it contain information about the id of the match, city - in this column name of the city where a match with corresponding match id is played, date - it takes care of date at which match was played, player of match - the name of player performed well by bat or ball or by both, player whose performance was important in match-winning, venue - the name of the stadium where the match was played, displayed in this column, neutral venue - if the field on which match is played is the home ground of any team then in this column, we have “1” as a representation of that, if this does not happen then this column contains “0”. team1team2 - both the columns contain the name of the teams playing in that match team1 refers to the team who bats in 1st inning and team2 stands for bowling team in the first inning, toss winner - in this column name of the team that won the toss is displayed, toss decision- it contains information about what toss the winning team chooses? Is it chosen to bat first or bowl? winner- as the name suggests it contains the name of winning team, result - it contains information about the winning team, does the winning team win by a run, or does it win by wicket? result margin- this column contains information on how much correspond to result in team win, for example, if the result the column contains run and result margin column contain 12 than winning team wins by 12 runs or if the result column contains wicket and result margin column contain 5 than winning team win by 5 wickets, eliminator- if the corresponding the match was an eliminator match than this column contains value “1” otherwise “0”, umpire1 and umpire2 - contains name of umpire of the match. In the data set, there was an error regarding a team name, the name of Rising Pune Supergiants mentioned as Rising Pune Supergiant which was producing error, by making a single team as two teams.

The dataset ”IPL 2008-2017 ball by ball.csv” is taken from Kaggle. It contains information about the ball by a ball match score of IPL matches conducted up to 2017. There are 76014 rows and 8 columns in this data set. 8 columns are in this data set named as follows: mid- contains match id, date- contains the date at which match was played, balling team - the name of the team who is bowling against batting team in that inning, batting team - this has the name of team batting in that inning, batsman - this column has the name of batsman who is playing, bowler- this column has the name of bowler who is bowling on the ball mentioned in the over. Run- this column contain the total run made by the batting team up to the ball mentioned in the Overs column. Wicket - contains the total number of wicket fall of batting team up to the ball mentioned in overs, overs

- this column tracks the record of the total bowl delivered by the playing team. Run last 5 - run made by batting team in last five overs from the current over. Wicket last 5 - total number of wickets taken by bowling team in last five overs from the current over. Total - in this column we have information about the total run made by the batting team in that inning.

IV. Computational Environment & Prediction Models

A. Goal Definition

Data Analysis and Interpretation: The goal of our Data The analysis is split into 3 parts: Exploratory Data Analysis, Descriptive Data Analysis and Predictive Data Analysis with the help of models.

Exploratory Data Analysis: We first loaded the data-set containing the outcome of the past IPL matches from the year 2008 to 2020. We read the data in a data frame and did some data cleaning processes. We then determined the different types of data types used and displayed some of the standard statistics of the same using the describe function in the Pandas library. We then moved on to defining the data into categorical, integral, or continuous types by determining the number of different values in each input variable. After finding out the null values in our data set we replaced the same with relevant data in some columns while dropping some of the others due to their irrelevance. After some more data cleaning, we moved on to Descriptive data analysis.

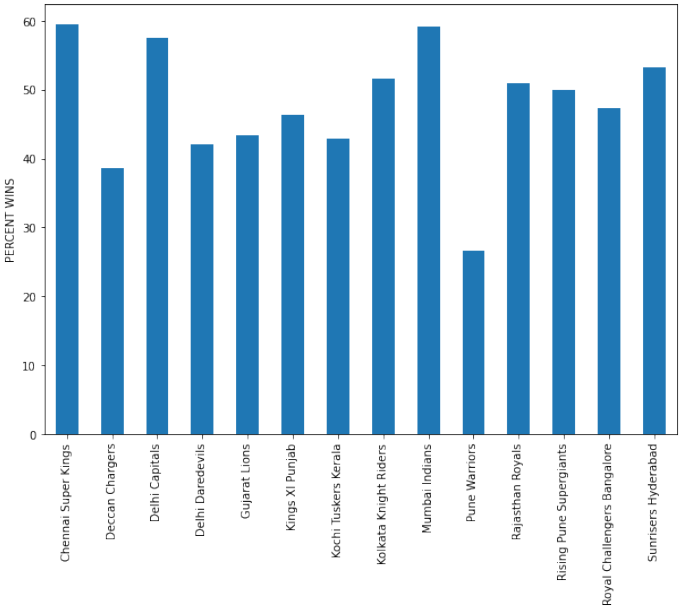

Fig. 1. percentage wins of different teams of total matches played

Descriptive Data Analysis: we implemented more result-oriented techniques with the help of libraries of python like -pandas, seaborn, matplotlib. pyplot e.t.c we draw a heat map to find the relation between different variables how different features are related to our targeted variable or result. We also define a function to know about such variables, the function is named correlation. We do this analysis to drop some features which are not much effective in deciding the result of the match. We draw graphs between various features to analyze links between them. We draw a bar graph between features toss to win and match win for each team to analyze the common statement "if a team wins the toss the team already wins half of the match". To get a correct analysis we also draw a graph between match wins v/s toss lost or toss win so that we can have better analysis to draw a statement on the effect of toss on a match and a particular team. We also draw a bar graph to find which team is the best performer by plotting total wins by a team. But this will give us a wrong result, as in IPL new teams induced in many seasons and some teams are removed so to achieve right conclusion we draw graphs between total wins and the total match played, and a graph of percentage wins. To check the accuracy of common belief in cricket about a ground " if a match is played on the home ground of a team ", we draw a graph between total wins by a particular team on a particular ground to check how the probability of win depends upon the home ground? Then we draw a graph between the wins of the team when it bats first and bowls first/ bats in the first inning and bowl in the first inning to analyze the performance of the team and decide which team is good at chasing? Which team's probability of win will be high if it bats first?

B. Computational Environment

In this project we have used five binary classification and regression models as implemented in the `scikit-learn` library in `Python`:

- Random Forest Classifier
- Decision Tree Classifier
- Linear Regression
- Support Vector Machine (SVM)
- Random Forest Regressor

C. Data Prediction Models

In statistics, **Linear regression** is a linear approach for modelling the relationship between a scalar response and one or more explanatory variables (also known as dependent and independent variables). In linear regression, the relationships are modelled using linear predictor functions whose unknown model parameters are estimated from the data. Such models are called linear models. Most commonly, the conditional mean of the response given the values of the explanatory variables (or predictors) is assumed to be an affine function of those values; less commonly, the conditional median or some other quantile is used.

The **Random Forest Classifier** consists of an ensemble of decision trees (DT) where each DT is a branching structure that represents a set of rules, distinguishing values in a hierarchical form. Depth of each DT and the number of such DTs to be used in our ensemble were the hyperparameters

which were selected after running a gridsearch on the training set for each ensemble,

Decision tree learning or induction of decision trees is one of the predictive modelling approaches used in statistics, data mining and machine learning. It uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves). Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctioof features that lead to those class labels

In machine learning, **Support Vector Machines** are supervised learning models with associated learning algorithms that analyze data used for classification and regression analysis. In Support Vector Regression, the straight line that is required to fit the data is referred to as hyperplane. The objective of a support vector machine algorithm is to find a hyperplane in an n-dimensional space that distinctly classifies the data points. The data points on either side of the hyperplane that are closest to the hyperplane are called Support Vectors. These influence the position and orientation of the hyperplane and thus help build the SVM.

Random Forest Regression is a supervised learning algorithm that uses ensemble learning method for regression. Ensemble learning method is a technique that combines predictions from multiple machine learning algorithms to make a more accurate prediction than a single model.

V. Result

The Data analysis help us to draw the following conclusions:

- The feature on which match-winning of a team1 depends on are following: City, neutral_venue, team2, toss_winner, toss_decision
- Toss winning highly affect the result of the match
- The highest win % is of Chennai Super Kings. This team wins about 60% of total match played.1
- The probability of a team winning a match will be high if the team wins the toss.5,3,4
- We use our data model to predict the ranking of teams and IPL2021 winner according to our model highest winning chances were of RCB and Then CSK, and we know Chennai Super Kings takes the Title. Our high prediction of RCB wins is because IPL unlikes a normal year not conducted in India completely, the major part of the tournament is conducted in Abu Dhabi so our result alters.

TABLE I
Error Score Card

	Evaluation Metrics		
Model	**MAE**	**MSE**	**RSME**
Random Forest Classifier	0.16	1.02	1.01
Decision Tree Classifier	0.33	1.80	1.34

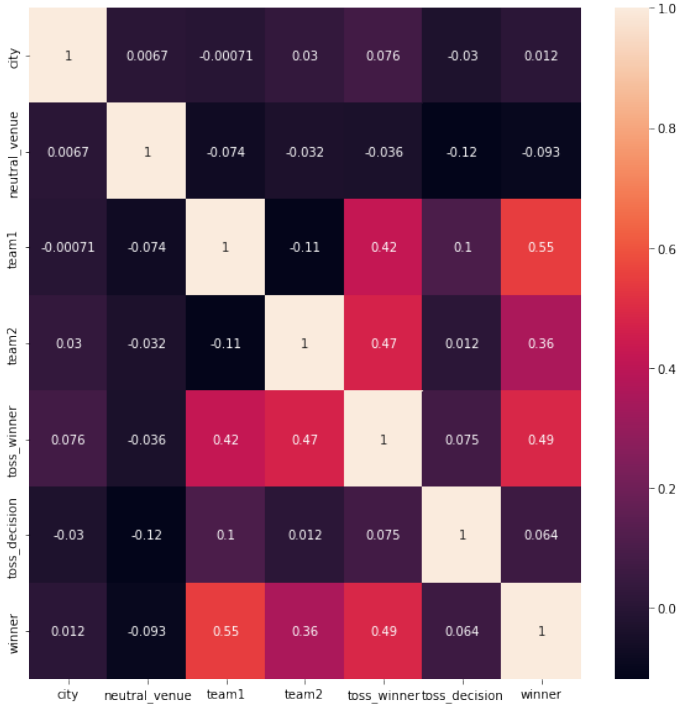

Fig. 2. Correlations between different features

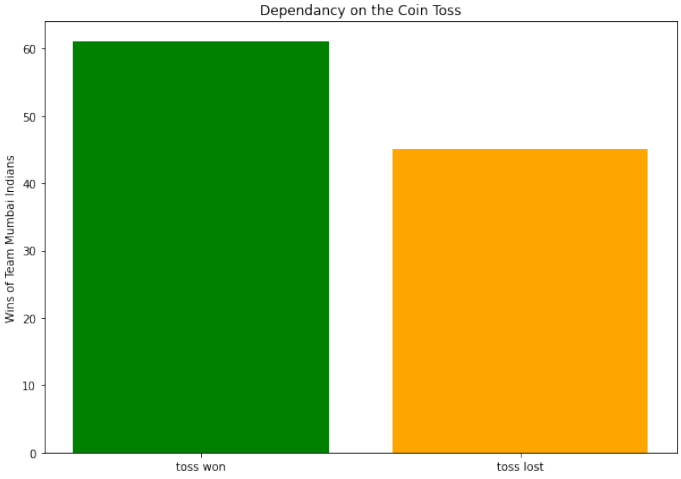

Fig. 3. Matches won by Mumbai Indians

- For predicting the winner out of the two teams when we have other features also available we use models: Random Forest Classifier, and Decision Tree classifier we get the following MAE, RMSE, and MSE results: hence On the basis of the above result, we decided to use Random Forest Classifier for predicting the winning team.
- For predicting score at a given point in the match we use Linear Regression, Support Vector Machine, Random Forest Regressor. Random Forest Regression, Support Vector Machine, and Linear Regression we get the following accuracies and MAE, RMSE, and MSE results : we conclude Random Forest Regression is best to train

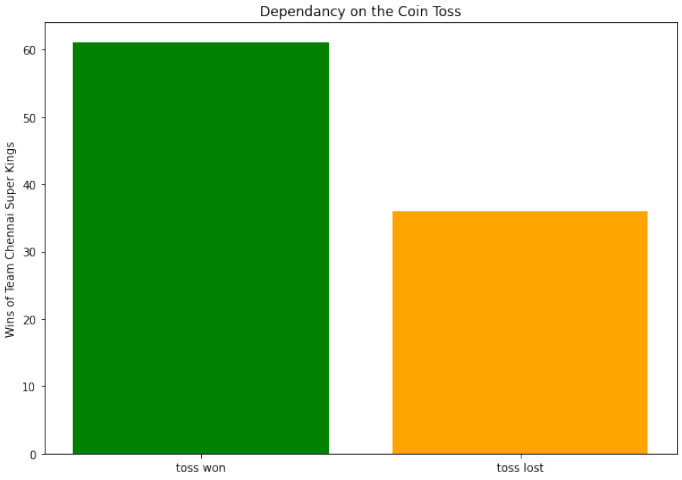

Fig. 4. Matches win by Chennai Super Kings

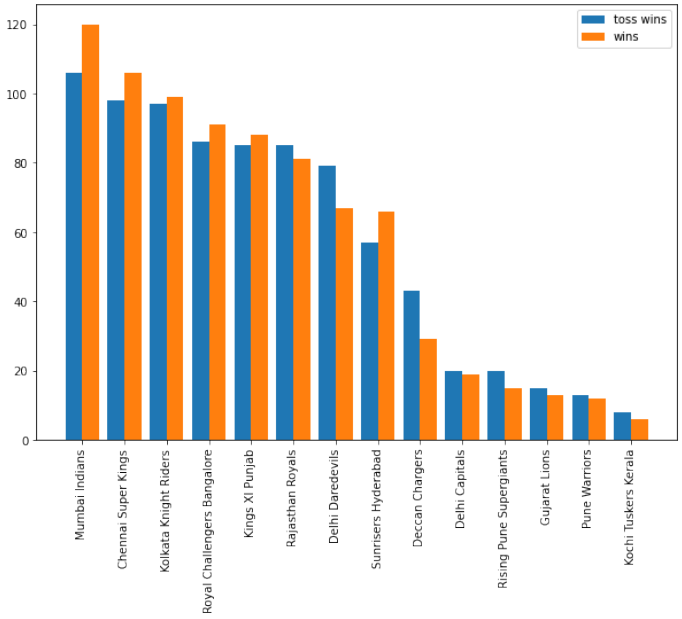

Fig. 5. Toss wins and match wins

TABLE II
ERROR SCORE CARD

Model	Evaluation Metrics		
	MAE	MSE	RSME
Random Forest Regressor	4.39	53.98	7.35
Linear Regression	12.90	292.68	17.10
Support Vector Machine	16.56	456.89	21.37

- The top 6 teams predicted by our model are the same as the top 6 teams in IPL 2021.6.7,8
- Also the 3 out of the top 4 teams which go into qualifiers are the same except for KKR
- Rajasthan Royals and Sunrisers Hyderabad were the poorest performing teams as predicted.7,8

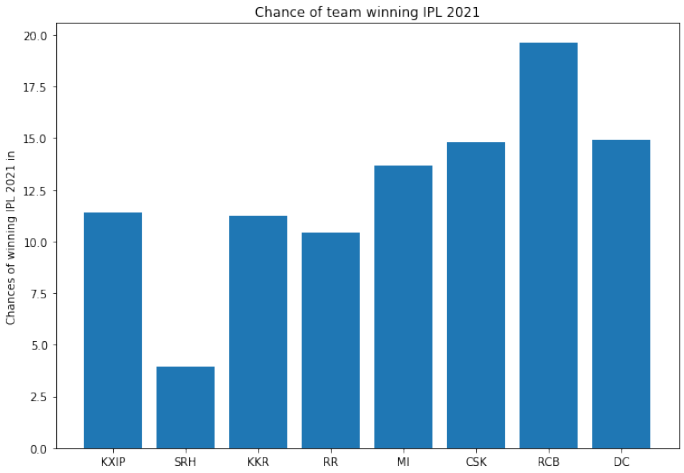

Fig. 6. Result of 20201 IPL Prediction by our model

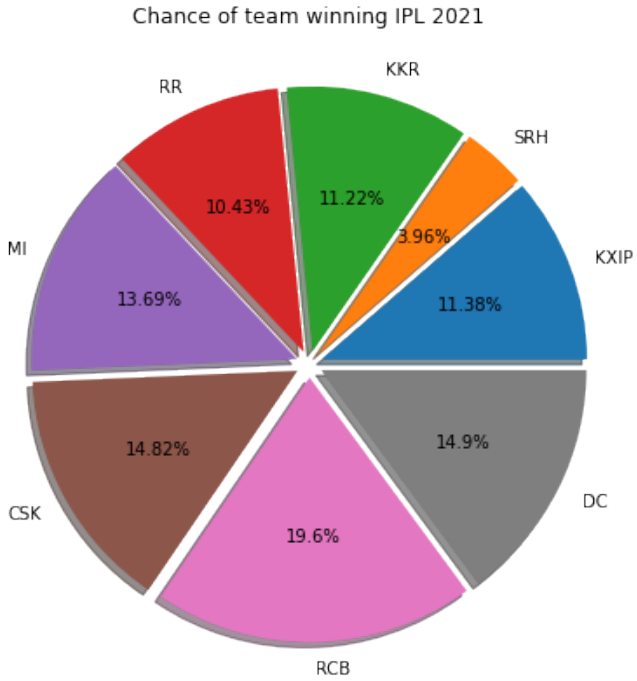

Fig. 7. Result of 20201 IPL Prediction by our model

VI. EXPERIMENTS

Our models use Random Forest Regressor, SVM linear Regressor, Random Forest Classifier, Decision Tree Classifier for prediction and while learning from the Data-set. Decision

IPL 2021 - Points Table

Teams	Mat	Won	Lost	Tied	NR	Pts	NRR
Delhi Capitals	14	10	4	0	0	**20**	+0.481
Chennai Super Kings	14	9	5	0	0	**18**	+0.455
Royal Challengers Bangalore	14	9	5	0	0	**18**	-0.140
Kolkata Knight Riders	14	7	7	0	0	**14**	+0.587
Mumbai Indians	14	7	7	0	0	**14**	+0.116
Punjab Kings	14	6	8	0	0	**12**	-0.001
Rajasthan Royals	14	5	9	0	0	**10**	-0.993
Sunrisers Hyderabad	14	3	11	0	0	**6**	-0.545

Fig. 8. Actual Ranking Result of 2021 IPL **[** credit:**Link]**

Tree classifier is a part of supervised Machine Learning where predetermined and well-defined steps are applied on inputs with the help of Linear Algebra to determine the outcome. However often a linear variation of input(from data-sets) is often not enough to encompass the desired output (or the target variable) to high precision. Hence in our model, we also use Random Forest Classifier which with the help of weights can include a variety of decision trees to predict the target variable. This process of combining the output of multiple individual models (also known as weak learners) is called Ensemble Learning. We also, use regression for numerical data analysis in the latter part using SVM linear regressor and Random forest regressor. A random Forest Regressor works like a Random Forest Classifier. The main difference is that a classifier is used in classification problems when the data after data cleaning is of Categorical type and hence cannot be numerically added, compared, or dealt with in contrast to a regressor which is used for numerical, integral, or continuous data analysis. A Support Vector Machine is ideally used for classification problems but the same concept is applied in the SVM regressor which with the help of hyper-planes (a linear separation between output data) introducing extra dimensions is used to create decision boundaries using hyper-parameters. After developing the model we ran some experiments using it to determine its accuracy to the outcome of an actual match. .

- This year (2021) the IPL match 1 was held between Mumbai Indians and Royal Challengers Bangalore. RCB won the toss and chose to field and eventually went on to win the match as well.
- This is also apparent from our Descriptive Data analysis and can be seen the team that wins the toss has a higher chance of winning the match, moreover, this can also be seen in the heatmap as the toss winner has a high correlation factor with the winner of the match.
- Our model predicts the right result, that is the team Royal Challengers Bangalore wins the match.
- We also used our model to predict the result of the match 18 of the same. Again Rajasthan Royals, the team which won the toss, against Kolkata Knight Riders went on to win the match.
- The model predicted the right outcome again for match 22, match 40 and match 53 of IPL 2021.
- For the IPL finals of 2021, which were held between Kolkata Knight Riders and Chennai Super Kings, Kolkata Knight Riders won the toss and decided to field. However, it was Chennai Super Kings which went on to win the IPL 2021 finals.
- Our model's prediction here was inconsistent but it was due to the match being an outlier case. Initially, Chennai Super Kings were at a disadvantage being given a huge the target of 193 runs but the conditions along with the chance and luck made Chennai Super Kings the eventual winner.

Using Random Forest regressor model (as it gives better accuracy than SVM Linear Regressor model) we have predicted the score values or the total runs made by the team and the predictions are in very close agreement to the actual value of runs. The heat-map also shows a very high correlation between runs and overs and this can also be seen from the experiments as the number of runs rises very rapidly in the power play overs, especially in the first and the last 5 overs of the match.

- We first ran the model for a match between Chennai Super Kings and Rajasthan Royals. The final runs made were 205 which was very close to the value 209 as predicted by the model.
- In the second experiment the final score was 132 of Sunrisers Hyderabad against Kings XI Punjab and our model predicted a score of 133.
- In the match against Chennai Super Kings, Delhi Capitals made a score of 165 which was again in close agreement to the score of 170 predicted by our model.

VII. Contributions and Knowledge Gained from the Project

This was a group project and hence we started by firstly exploring data-sets and similar models on Kaggle and other websites. The Exploratory Data Analysis part and data Cleaning was performed by Pranav Singla and Sumit Kumar. The Descriptive Data Analysis part and Predictive Data analysis part was performed by Shreedhar Malpani and Om Mihani. We all worked together on the report.

We got to learn a lot from this project as new functions were explored from different Python Libraries like Pandas, skLearn, Seaborn, Numpy, etc. We got to learn about probability and statistics from the course and how these can be applied in Data Analysis for tackling real-life problems. We also had fun exploring how different variables like the toss, weather, city, etc... get to play a deciding role in the outcome of a cricket match. We also learned how to apply our data science knowledge to solve real-life problems

VIII. Future Work

By doing player by team analysis one can identify which player will make the highest run against a particular team

in a particular city. This analysis also predicts the wickets taken by a bowler against a particular team on a particular occasion hence helping people to correctly predict the result of the match. We also include the effect on the performance of a team because of exchanges of players. Further to have the more accurate results we can also use data of the current session to predict the result of the match accurately.

IX. Acknowledgements

We would like to extend our deepest gratitude to our professors Prof. Amit Sethi, Prof. Manjesh K. Hanawal, Prof. Sunita Sarawagi and Prof. S. Sudarshan gave us the golden opportunity for giving this insightful project. We would also like to express our humble and special thanks to all the TA's who helped us with our doubts and henceforth in overcoming all the hurdles we faced without which we would not have been able to complete this project and the assignments related to this insightful project.

References

[1] https://towardsdatascience.com/predicting-ipl-match-winner-fc9e89f583ce
[2] https://www.geeksforgeeks.org/ipl-score-prediction-using-deep-learning/
[3] https://scikit-learn.org/stable/
[4] https://machinelearningmastery.com/columntransformer-for-numerical-and-categorical-data/
[5] https://github.com/aasis21/4th_umpire
[6] https://github.com/kuharan/IPL-ML-2018/tree/master/Dataset
[7] https://github.com/ajithnair20/IPL-Prediction/blob/master/IPL-Prediction.ipynb
[8] (IPL 2021) https://github.com/sankha1998/indian-premier-league-2021
[9] (IPL2008-2020)https://www.kaggle.com/patrickb1912/ipl-complete-dataset-20082020/activity
[10] (IPL2008-2016)https://github.com/12345k/IPL-Dataset/blob/master/IPL/data.csv
[11] https://pianalytix.com/ipl-match-prediction/
[12] alltime-https://www.kaggle.com/kanishk307/ipl-indian-premier-league-all-time-statistics?select=bat_most_fifties.csv
[13] All-kaggle-https://www.kaggle.com/datasets?search=IPL
[14] https://www.analyticsvidhya.com/blog/2021/10/building-an-ipl-score-predictor-end-to-end-ml-project/
[15] https://public.tableau.com/en-gb/search/all/

Appendix

The Python Notebook containing the entire code for the above analysis is present in **this** GitHub link. In this notebook, we've done a very detailed analysis of all the parameters along with explanations, conclusions and supporting graphs to help grasp a notion of the analysis.