

Subject: Super Predictor Of Indian Premier League (IPL)

1.1 Overview:

The project is all about a sports website which has a predictor and a dashboard. We have created dashboard using Cognos Analytics and these dashboards are similar and are useful in visualizing important insights of IPL. The Predictor is created using Jupyter Notebook and it mainly focuses on predicting first innings score of future IPL matches based on data of matches held in past. For this, we are using concepts of Machine Learning. The dashboard present on the website is created using chart.js. For frontend of website, HTML and CSS are used whereas for backend, Django is used.

1.2 Purpose:

IPL has a huge hype in India. High level of uncertainty and close encounters have urged fans to watch the matches. Fans not only watch matches ball by ball but are also eager to predict future results and sometimes they reach online fantasy sports platforms for the same. This predictor can be useful to such people. IPL teams can make or change their strategies by using this predictor.

2.1 Existing approaches or method to solve this problem:

In IPL matches, we often see past stats and stats showing the predicted score or may be the probability of team winning the match based on current match situation. For the same, we have created dashboard and predictor. Cognos Dashboard is created using past data of 2008-2019 and it is useful in visualizing

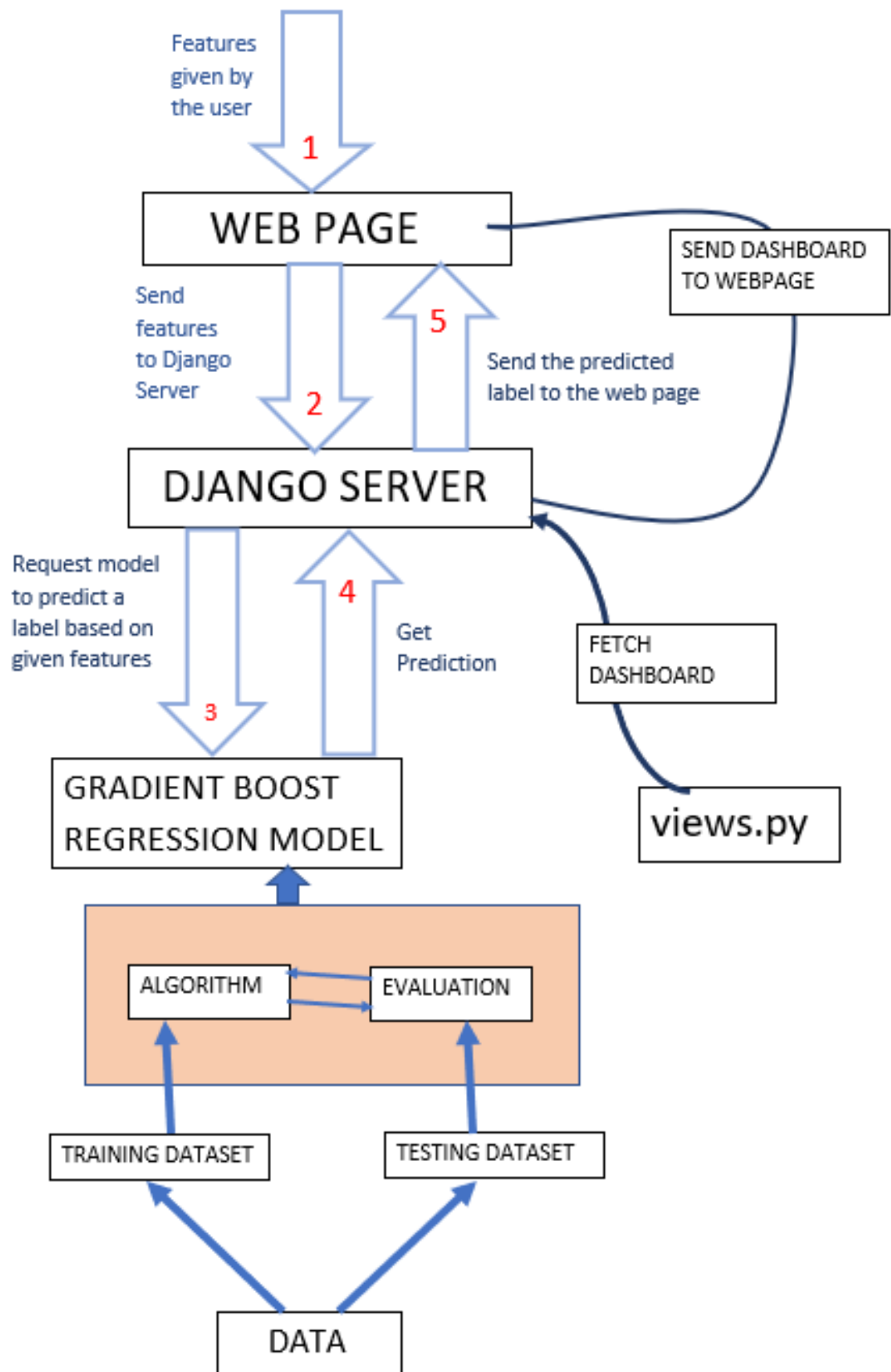
important insights as mentioned in the project. The dashboard visualizes following capabilities:

1. To find the team that won the most number of matches in the entire IPL.
2. To find the team that lost the most number of matches in the entire IPL.
3. Does winning a toss increase the chances of victory.
4. To find the player with the most player of the match awards in the entire IPL.
5. To find the city which that hosted the maximum number of IPL matches.
6. To find the most winning team for each season.
7. To find the on-field umpire with the maximum number of IPL matches.
8. To find the biggest victories in IPL while defending a total and while chasing a total.
9. Which team won the most matches while batting first.
10. Which team won the most matches while batting second.
11. List of teams which have won matches by most runs cumulatively.

2.2 Proposed solution

For making the predictor, first we searched for the proper dataset according to requirements. And then we applied gradient boost regressor on the data after analysing and applying proper steps on data. This predictor predicts the first innings score based on the features given by the user. After making predictor using Jupyter Notebook, we designed a website which has frontend made of HTML and CSS. HTML and CSS are used to design a form which takes features from user. This form is connected to Django Server (Backend) which is then connected to our predictor. In this way, website will give predictions. We also attached our dashboard on the website.

3.1 Block diagram:



3.2 Hardware/Software:

Software Designing of the Project:

❖ Frontend

- HTML, CSS, Chart.js
- VSCode, Cognos Analytics

❖ Backend

- Django
- PyCharm

❖ Predictor

- Python, Matplotlib, Seaborne, NumPy, Pandas
- Anaconda- Jupyter Notebook

Hardware Designing of the Project:

- intel CORE i7 10th Gen (Cognos Analytics)
- intel CORE i5 8th Gen (Jupyter Notebook)
- intel CORE i5 10th Gen (VSCode)
- intel CORE i5 8th Gen (PyCharm)

4. Experimental Investigations:

Experimental Analysis while working on the dashboard:

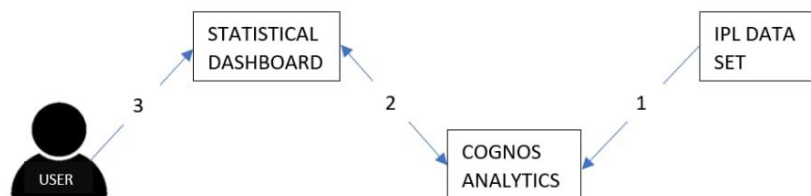
For Cognos dashboard, we looked for some sports dashboards on web to get some idea. Then we searched for match-by-match dataset which was required for making dashboard. Then we added some missing information. Then we cleaned the data (correcting some information about teams, umpires and cities). We also added a required column named "Looser". We also generated some required queries in Cognos Analytics to obtain the required output (like max, count, etc.). Then we created graphs using line graph, pie, column chart, etc.

Experimental Analysis while working on the predictor:

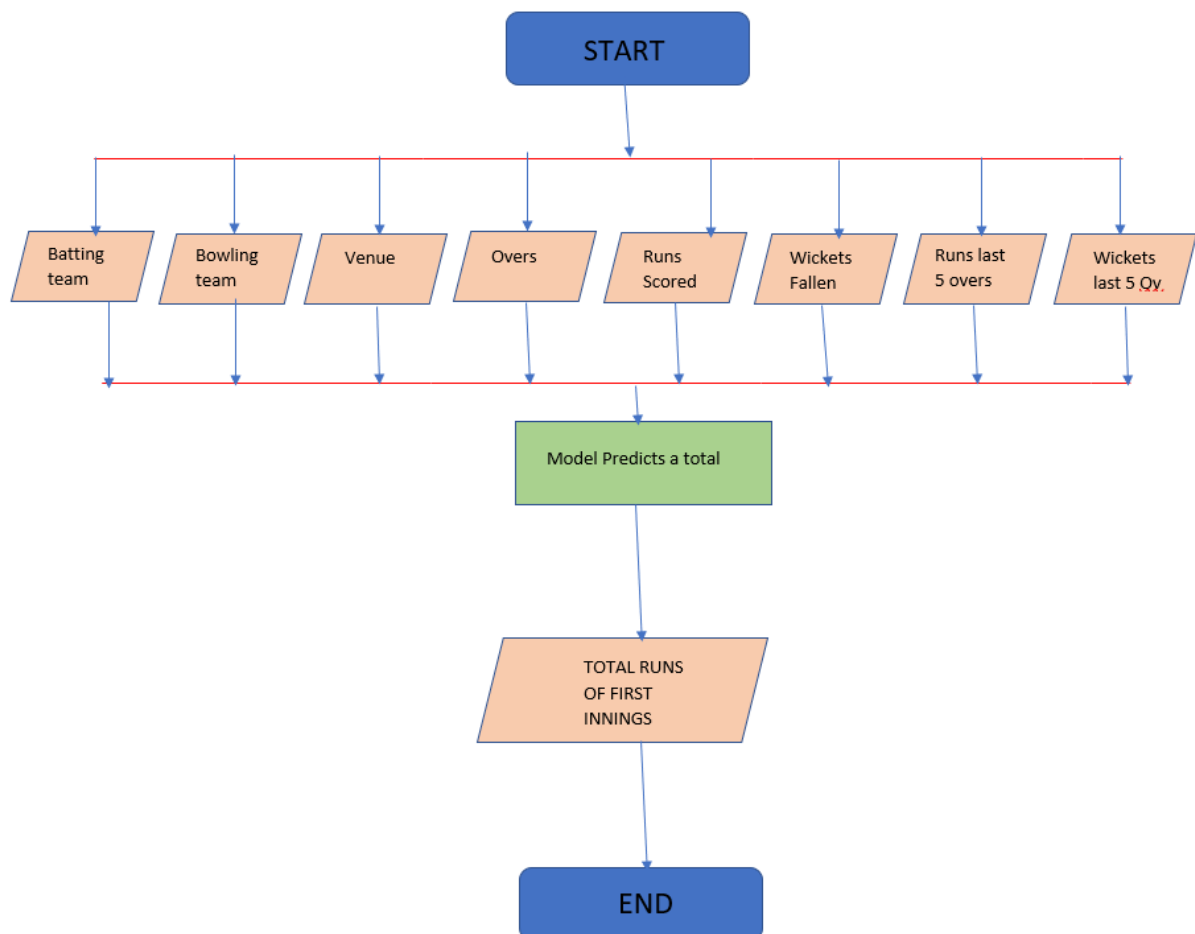
For predictor, we searched for the data of ball-by-ball of first innings having all the required fields. Then we understood the data by obtaining its shape. We then cleaned the data by doing some required steps. We corrected some values of teams and venues and removed some rows which were not useful. We then visualized some graphs to obtain correlation among the fields and dropped the columns which were less correlated. We then analysed which columns should be 'features', and which column should be 'label'. We then split the data into training and testing data. We then tried different models on our training data and tested on the testing data. We did cross validation on each model. And at last tuned the models for higher accuracy. We then chosen gradient boost regressor as our final model which was giving higher accuracy than other models.

5. Flowchart:

Flowchart of the solution(dashboard):

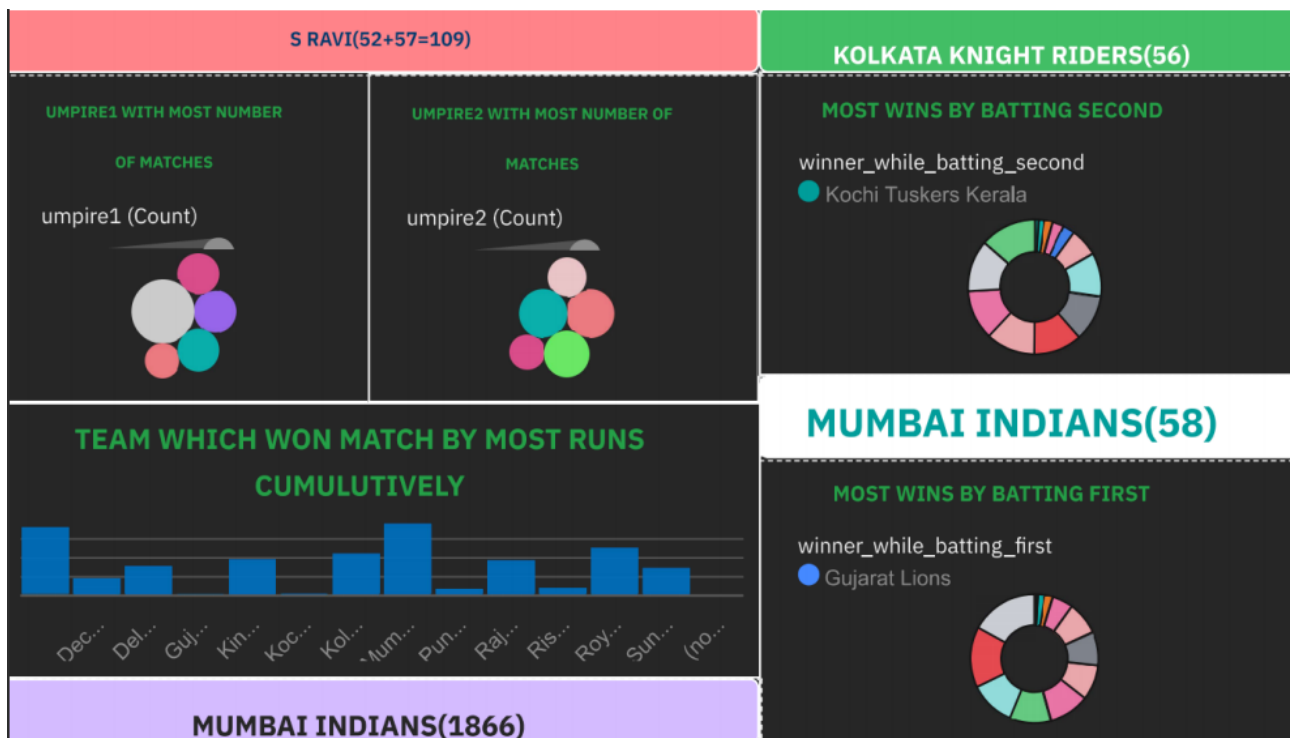
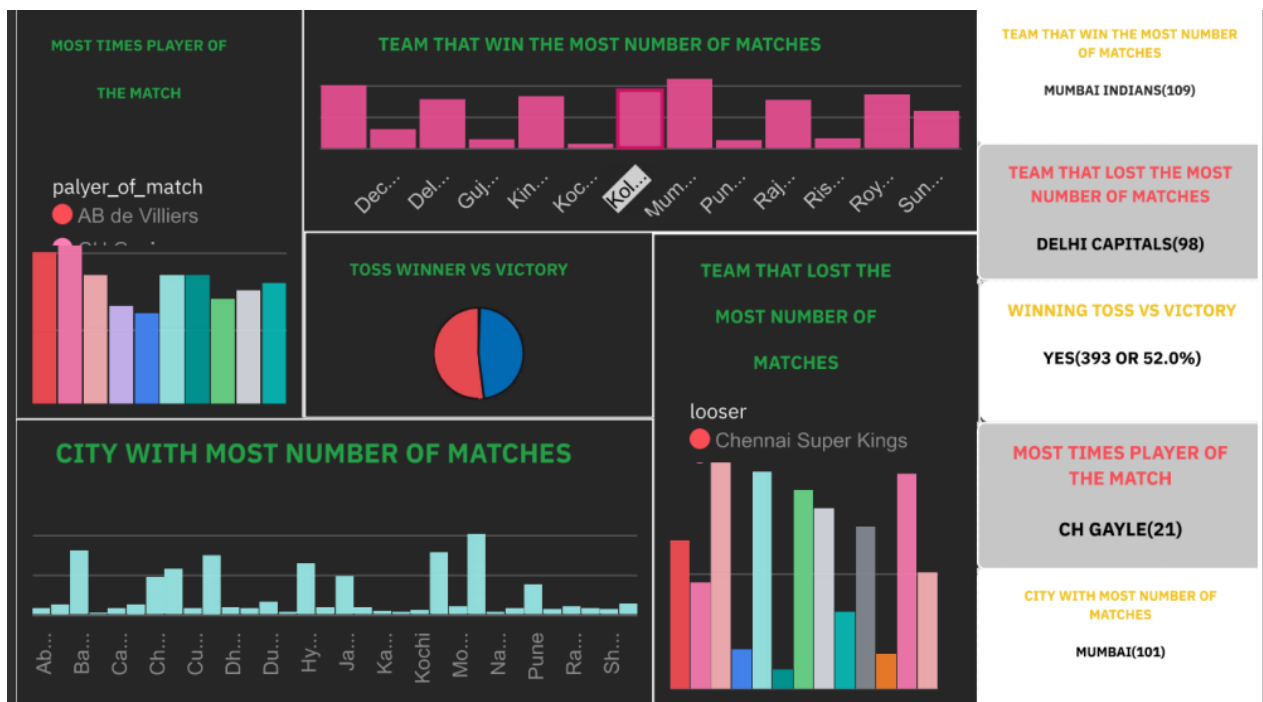


Flowchart of the solution(predictor):



6. Result:

Photos of Cognos dashboard:



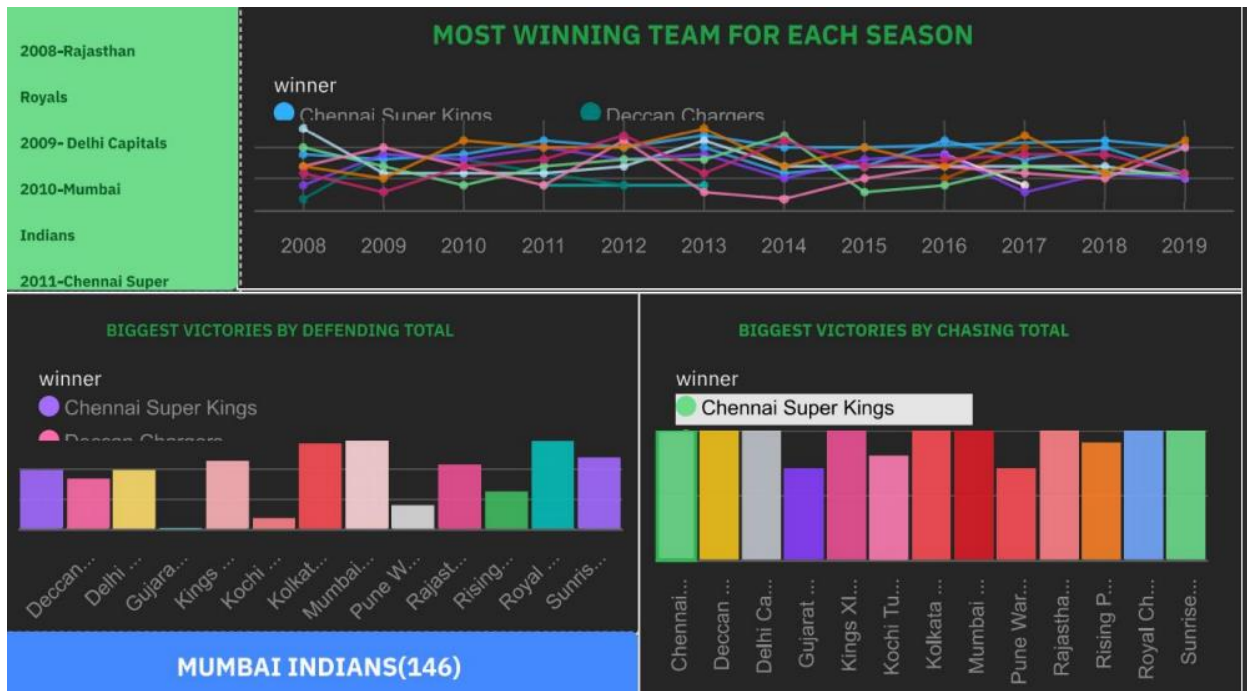


Photo of our project website(<https://ipl-cricfreak.herokuapp.com/>):

WhatsApp Strikers | Predictor

ipl-cricfreak.herokuapp.com/predictor/

Strikers Home Dashboard

Batting Team: Chennai Super Kings

Bowling Team: Mumbai Indians

Venue: MA Chidambaram Stadium, Chepauk

Overs: 7.0

Runs Scored: 55.0

Wickets Fallen: 2.0

Runs Scored in previous 5 Overs: 43.0

Wickets taken in previous 5 Overs: 2.0

Predict

Approximate score at the end of inning: 173

7. Advantages and Disadvantages of proposed solution (Gradient Boosting)

Advantages of Gradient Boosting are:

- Often provides predictive accuracy that cannot be trumped.
- Lots of flexibility - can optimize on different loss functions and provides several hyper parameter tuning options that make the function fit very flexible.
- No data pre-processing required - often works great with categorical and numerical values as is.
- Handles missing data - imputation not required.

Disadvantages of Gradient Boosting are:

- Gradient Boosting Models will continue improving to minimize all errors. This can overemphasize outliers and cause overfitting.
- Less interpretative in nature, although this is easily addressed with various tools.

8. Applications:

- Fantasy Sports Platform- Fans who are eager to predict future results can reach online fantasy sports platforms. This predictor can be useful to such people.
- IPL Teams- IPL Teams can use this predictor before their match against a certain opponent. They can predict the opponent's score and can make/change strategies accordingly.
- Sports Analysis- Our dashboard can be used to visualize and analyse past stats of IPL and predict future results.

9. Conclusion:

We finally connected all the domains (frontend, backend, dashboard and predictor) and created a website for IPL fans where one can find predictor and dashboard at one place. Cognos Dashboard is useful in visualizing important insights of IPL. Predictor takes features like batting team, bowling team, venue, overs, runs, wickets, runs_last5 and wickets_last5 and predicts a final total.

10. Future Scope:

- After every season of IPL, we can update the data by adding the data of that season to increase the accuracy of our model.
- We can add a feature of login/sign in to our website so that we can offer more functionality.
- We can integrate a chatbot so that users can interact with our website easily.

11. Bibilography:

References of websites referred:

- Chart.js Dashboard- <https://www.chartjs.org/docs/latest/>
- Django - <https://docs.djangoproject.com/en/3.2/>
- Cognos Analytics Dashboard-
<https://www.youtube.com/watch?v=SY8IVZFWJt0>
<https://www.ibm.com/docs/en/cognos-analytics/11.1.0?topic=stories-get-started-dashboards>

Dataset used in predictor- <https://github.com/srinathkr07/IPL-Data-Analysis/blob/master/matches.csv>

- Predictor- https://youtube.com/playlist?list=PLu0W_9lII9ai6fAMHp-acBmJONT7Y4BSG
https://youtu.be/p_tpQSY1aTs
<https://numpy.org/doc/>
<https://pandas.pydata.org/docs/>
<https://scikit-learn.org/stable/>

Dataset used in predictor-

<https://www.kaggle.com/rushikeshlavate/xgboost-for-predicting-ipl-s-1st-inning-score>

APPENDIX:

Source code:

IBM hack challenge git repo: <https://github.com/smartinternz02/SBSPS-Challenge-5331-Super-Predictor-of-Indian-Premier-League-IPL->

Live website git repo: <https://github.com/darsh295/IPL-score-predictor-and-dashboard>

Website URL: <https://ipl-cricfreak.herokuapp.com/>