

DECLARATION

We hereby declare that the project entitled Cricket Analysis and Prediction is an authentic record of our own work carried out in the Electronics and Computer Engineering Department, Himalaya College of Engineering under the guidance of Bal Krishna Nyaupane during 7th and 8th semester (2018).

Date:

Roll.No	Name	Signature
18	Jeevan Pandey	-----
24	Madhu Nyoupane	-----
31	Pradeep Kiran Timsina	-----
33	Prakash Singh Madai	-----

Counter Signed by

Faculty Mentor:

Er. Bal Krishna Nyaupane
Assistant Professor
Department Of Electronic and Computer Engineering
Pulchowk Campus, IOE

ACKNOWLEDGEMENTS

We would like to express our sincerest appreciation to all those who provided us the possibility to complete this report. We express a special gratitude to The Department of Electronics and Computer Engineering, Himalaya College of Engineering for providing us the opportunity to explore our interest and ideas in the field of engineering through this project.

Without the support, patient and enthusiasm of the following people this paper would not have been completed. It is to them, we owe our sincere gratitude. We would like to acknowledge and express our big thanks to everyone for his/her support and encouragement which has inspired us for this project.

We would like to acknowledge to the HOD of computer and electronics department, Er. Ashok Gharti Magar. Furthermore, we would also like to acknowledge, with much appreciation, the crucial role of our Supervisor Er. Bal Krishna Nyaupane who assisted us during the research and feasibility study of the project. Special thanks go to the project coordinator Er. Narayan Adhikari Chhetri.

ABSTRACT

With the advent of statistical modeling in sports, predicting the outcome of a game has been established as a fundamental problem. We embark on predicting the outcome of a One Day International (ODI) cricket match using a supervised learning approach from a team composition perspective. We first preprocessed the data by filling the missing entries with appropriate values and then by changing the format of the specific features to make it suitable for model generation. We then generated newer features like home team advantage, strength of each team and its performance in past few matches. Our project suggests that the relative team strength between the competing teams forms a distinctive feature for predicting the winner. Modeling the team strength boils down to modeling individual player's batting and bowling performances, forming the basis of our approach. We use career statistics as well as the recent performances of each player to model him. Player independent factors like batting and bowling averages have also been considered in order to predict the outcome of a match. We then used various machine learning classifiers like decision tree, random forest, logistic regression and support vector machines to model our data. We then calculated the accuracy of each model by dividing the number of right predictions out of total predictions made. Out of all the models used we found out that the Logistic Regression gives us the better accuracy. We finally came to conclusion that along with factors like toss decision, venue of the match, prediction depends on their player's statistics like batting average, bowling average, their performance in past few matches.

Keywords

Prediction, cricket, data mining, visualization, preprocessing, model

Contents

1. Introduction.....	1
1.1 Project Overview.....	1
1.2 Problem Definition.....	3
1.3 Objectives.....	4
1.4 Project Milestone.....	4
2. Literature Reviews	5
3. Requirement Analysis and feasibility study	9
3.1 SRS Details	9
3.1.1 Introduction	9
3.1.2 Overall Description.....	9
3.1.3 System Features	10
3.1.4 External Interface Requirements	11
3.1.5 Other Nonfunctional Requirements.....	12
3.2 Software Requirement Specification.....	13
3.3 Feasibility Analysis of the Project	15
3.3.1 Economic feasibility:.....	15
3.3.2 Operational feasibility:	15
3.3.3 Schedule Feasibility:.....	16
3.4 Cost Analysis.....	16
3.5 Assumptions and Constraints	16
4. Design and Analysis	18
4.1 System Design.....	18
4.2 Activity diagram.....	19
4.3 Class Diagram	20
4.4 Data Flow Diagram (level 0).....	21
4.5 Sequence Diagram.....	22
4.6 Use case Diagram.....	23
4.7 Entity Relationship Diagram	25
4.8 System Flow	26
5. Methodology	27
5.1 Design Phases.....	30

5.2 Algorithm Implementation	37
5.2.1 Logistic Regression and Model Evaluation	37
6. Testing and Evaluations	43
6.1 Unit Testing	43
6.2 Integration Testing	43
6.3 Acceptance Testing	43
6.4 Inferences Drawn	43
7. Limitations and Future Enhancement	44
7.1 Future Direction	44
8. Conclusion	45
9. Project Metrics	46
9.1 Challenges Faced.....	46
9.2 Interdisciplinary Knowledge Sharing.....	46
9.3 Gantt Chart	47
9.4 Responsibility Assignment Matrix.....	47
10. References	48
Appendices.....	50
Some Snapshot of the project	50

TABLE OF FIGURE

Figure 4.1 System Design	18
Figure 4.2 Activity Diagram	19
Figure 4.3 Class Diagram	20
Figure 4.4 Data Flow Diagram (level-0).	21
Figure 4.5 Sequence Diagram.	22
Figure 4.6 Use Case Diagram.	23
Figure 4.7 ER Diagram	25
Figure 2.3 Model Evaluation	33
Figure 3.4.1: Graph Plot of Sigmoid Function	39
Figure A.1 : User Interface Home page	50
Figure A.2: Statistics Result Page	50
Figure A.3 Prediction Result Page	51
Figure A.4: Analysis Result Page	51
Figure A.5: Country Data Result Page	52
Figure A.6: Data search form	52
Figure A.7: Prediction form	53

LIST OF ABBREVIATIONS

API	: Application Programming Interface
App	: Application
BCT	: Bachelors of Computer Engineering
DHOD	: Deputy Head Of Department
GUI	: Graphical User Interface
HCOE	: Himalaya College Of Engineering
HOD	: Head Of Department
http	: hypertext transfer protocol
ID	: Identification
SDLC	: Software Development Life Cycle
SQLite	: Structured Query Language
SVM	:Support Vector Machine
UI	: User Interface
URL	: Uniform Resource Locator

1. Introduction

1.1 Project Overview

Statistical modeling has been used in sports since decades and has contributed significantly to the success on field. Cricket is one of the most popular sports in the world. Various natural factors affecting the game, enormous media coverage, and a huge betting market have given strong incentives to model the game from various perspectives. However, the complex rules governing the game, the ability of players and their performances on a given day, and various other natural parameters play an integral role in affecting the final outcome of a cricket match. This presents significant challenges in predicting the accurate results of a game. The game of cricket is played in three formats - Test Matches, ODIs and T20s. We focus our research on ODIs, the most popular format of the game. To predict the outcome of ODI cricket matches, we propose an approach where we first estimate the batting and bowling potentials of the 22 players playing the match using their career statistics and active participation in recent games. We then use these player potentials to render the relative dominance one team has over the other. Taking two other base features into account, namely, toss decision and the venue of the match, along with the relative team strength, we adopt supervised learning algorithms to predict the winner of the match.

We initiate our project by fetching data from sites like ESPN and Cricksheet. The data we obtained is in the form of csv file where each file describes each match details. Then we generated python scripts to club the data entries of each and every match. We then applied feature selection to select features like teams involved, match venue, toss decision and winner of the match. We then used feature generation to create more important features like strength of each team, performance of each team in past few matches, probability of team batting first and winning probability of particular team at a specific location based on all the previous matches played between those two teams at the specific venue. The strength and performances of each match are calculated in terms of relative strength and relative performances. The strength of each team in a

particular match is calculated based on the batting of each player in that match. Further the team batting average is calculated by taking the mean of batting averages of all the players of the team. After getting the strength of each team we subtract the strength of team A with team B to get relative strength of overall match. If relative strength is positive it means team A is stronger than team B. If negative it means team B is stronger than team A. Else both are equally strong. Along with strength we have also considered the past few match performances of each team as it tell us whether the team is in good form or not. The performances are calculated by taking mean batting average of past all matches of each team. After getting individual team performances we then calculate the relative performance of the team by subtracting team A performance with team B performance. If the relative performance is positive it means team A is in better form than team B. If negative then it means team B is in better form. Otherwise both are in equal form. The magnitude tells us the amount with which one team is better than other.

Relative Strength = Strength of Team A – Strength of Team B

Batting Average of Team = Σ (Bat average of each player)/No. of players

After preprocessing and feature generating we used the machine learning to generate models which can be used to predict the results. We used different models and calculated the accuracy from each model and finally selected the one with best accuracy. We used Random Forest, Logistic Regression and SVM. Out of all these we used Logistic regression as it gave us the better accuracy of 60% among all other models. After generating models we developed a GUI in python using flask module so that user can easily interact with our system rather than going to command prompt and typing all the fields.

1.2 Problem Definition

Cricket is the most popular sport in the subcontinent. Various natural factors affecting the game, enormous media coverage, and a huge betting market have given strong incentives to model the game from various perspectives. However, the complex rules governing the game, the ability of players and their performances on a given day, and various other natural parameters have not been taken into consideration while designing the prediction softwares which are already existing in the market. As a result their accuracy fails and prediction goes wrong most of the times. The most common and popular form of cricket is the One Day International (ODI), where over 50- overs per side are played. As is typical in games of sport, winning is the ultimate goal. Some studies, (De Silva, 2001), analyze the magnitude of the victory, but most consider the factors affecting winning. Kaluarachchi and et al (2010) takes into account various factors affecting the game including home team advantage, day/night effect and toss, etc., and uses the Bayesian classifier to predict the outcome of the match. Sankaranarayanan et al (2014) used machine learning approach to predict the result of a one day match depending on the previous data and in game data. Sohail Akhtar and Philip Scarf (2012) have forecasts match outcomes in test cricket in play, session by session. Match outcome probabilities at the start of each session are forecast using a sequence of multinomial logistic regression models. These probabilities can facilitate a team captain or management to consider an aggressive or defensive batting strategy for the coming session. But these probabilities can be increased by taking into consideration factors like team composition and their performance in past few matches, the batting averages and the bowling averages of each player in the team and the winning probability of team at a specific venue against a specific team.

1.3 Objectives

- To predict the results of the cricket match by not only taking factors like toss, venue, day-night but also by considering factors like team composition, the batting and bowling averages of each player in the team.
- To see the Statistics of a particular player in their past years and to review his performance stats.

1.4 Project Milestone

- Machine learning based model which will predict the ODI match results in advance by taking into consideration the factors like team composition, players batting and bowling averages and their performances in the past along with other parameters like toss decision and venue and home team advantage.
- Web based GUI for making the user friendly interaction with the model.
- Bar chart of accuracy attained with different models.

2. Literature Reviews

According to [2] factors contributing to winning games are imperative, as the ultimate objective in a game is victory. The aim of this study was to identify the factors that characterize the game of cricket, and to investigate the factors that truly influence the result of a game using the data collected from the Champions Trophy cricket tournament. According to the results, this cricket tournament can be characterized using the factors of batting, bowling, and decision-making. Further investigation suggests that the rank of the team and the number of runs they score have the most significant influence on the result of games.

[4] Embarks upon a very critical aspect that the team composition changes over time. It propose novel methods to model batsmen, bowlers and teams, using various career statistics and recent performances of the players. It propose a novel dynamic approach to reflect the changes in player combinations.

[5] Proposes a model to predict the winner at the end of each over in the second innings of an IPL cricket match. Our methodology not only incorporates the dynamically updating game context as the game progresses, but also includes the relative strength between the two teams playing the match. Estimating the relative strength between two teams involves modeling the individual participating players' potentials. To model a player, we use his career as well as recent performance statistics. Using the various dynamic features, we evaluate several supervised learning algorithms to predict the winner of the match. Finally, using the Random Forest Classifier (RFC), we have achieved an accuracy of 45.79% - 54.15% over the course of second innings, with an overall accuracy of 57.68%.

[6] Focuses on the prediction of likelihood of India winning or losing in One Day International (ODI) cricket match against Australia by fitting the logistic regression model. According to ICC ODI championship rating, dated 7th August 2015, India holds

2nd position with 5875 points and 115 rating by playing 51 matches. Data from actual recent matches with five independent variables and one dependent binary logistic variable are used throughout to illustrate the implementation of this successful use of mathematical and statistical principles to the solution of a practical problem in one-day international cricket match.

In [8], a machine learning model has been developed that predicts match result on every ball played. Using Duckworth- Lewis formula match outcome will be predicted for live match. For every ball bowled a probability is calculated and probability figure is plotted. For betting industry this model and the probability figure will be very useful for bettor in deciding which team to on and how much to bet.

In [12] a model has been proposed that has two methods, first predicts the score of first innings not only on the basis of current run rate but also considers number of wickets fallen, venue of the match and batting team. The second method predicts the outcome of the match in the second innings considering the same attributes as of the former method along with the target given to the batting team. These two methods have been implemented using Linear Regression Classifier and Naive Bayes Classifier for first innings and second innings respectively. In both methods, 5 over intervals have been made from 50 overs of the match and at each interval above mentioned attributes have been recorded of all non-curtailed matches played between 2002 and 2014 of every team independently. It has been found in the results that error in Linear Regression classifier is less than Current Run Rate method in estimating the final score and also accuracy of Naive Bayes in predicting match outcome has been 68%.

[13] deals with the evaluation of the Duckworth Lewis method, identifying its limitation, and devising a modification to address these limitations. The Duckworth Lewis method, or D/L method, was created by Frank Duckworth and Tony Lewis. The International Cricket Council (ICC) adopted D/L method in 1999 to address the issue

of delayed one-day cricket matches due to interruptions such as inclement weather conditions, poor light and floodlight failures, and crowd problems. They have attempted to identify the shortcomings in the existing Duckworth Lewis method using data mining algorithms such as Random Forests and C4.5. They have also shown that the p-values and other data mining techniques serve a dual purpose of not only evaluating whether systems such as D/L method have been exploited by taking advantage of their properties such as simplicity, but also devising alternate and robust approaches (or models). In the first part of their project, they have analyzed fifty one-day international (ODI) cricket matches, in which the Duckworth Lewis method has been applied, using tools such as WEKA and Microsoft Excel. They have observed that the Duckworth Lewis method has some limitations. As a result, using data mining methods they have shown that the Duckworth Lewis system has proven over time to be biased towards the team batting first and the team winning the toss -- a toss refers to the coin-flip at the beginning of the match used to decide who bats or fields. Bias in the context of the report is defined as taking advantage of the properties of systems such as the Duckworth Lewis method. They also showed that such an "exploitation" of the system permits prediction of the match winner with outcomes that are better than just chance. Using the analyses described above, they propose a modification to the existing Duckworth Lewis method to reduce this bias by considering the "venue" of the game as an additional resource along with the two existing resources-overs and wickets - to predict the target score. They have done a basic evaluation of the reduction in bias due to the proposed changes. The modification has helped not only to reduce the bias but also to alleviate the impact of factors such as toss and team batting first in predicting the target scores in limited-overs cricket matches.

[14] identifies rising stars in cricket domain by employing machine learning techniques. More precisely, it predict rising stars from batting as well as from bowling realms. For this intent, the concepts of co-players, team, and opposite teams are incorporated and distinct features along with their mathematical formulations are

presented. For classification purpose, generative and discriminative machine learning algorithms are employed, and two models from each category are evaluated. As a proof of applicability, the proposed approach is validated experimentally while analyzing the impact of individual features. Besides, model and category wise assessment is also performed. Employing cross validation, it demonstrate high accuracy for rising star prediction that is both robust and statistically significant. Finally, ranking lists of top ten rising cricketers based on weighted average, performance evolution, and rising star scores are compared with the international cricket council rankings.

3. Requirement Analysis and feasibility study

3.1 SRS Details

3.1.1 Introduction

Purpose

The purpose of this project is to provide software having version no. 1.0 that takes cricket match data as input in the form of team names, the toss decision, the toss winner, venue and output the predicted value as the winner of the match.

Document Conventions

The document follows a pre-specified Modern Language Association (MLA) format. Bold faced text has been used to emphasize section and sub-section headings. Every requirement statement written hereby is to have its own priority and no prior inheritance is assumed anywhere.

Intended Audience and Reading Suggestions

The document is intended for all the developers involved in the software and all the users who will be using the software, testers and documentation writers. The rest of the SRS specifies the functioning the match predictor software in detail.

Project Scope

The goal of the project is to predict the result of the match beforehand by taking into consideration the data and the results of the previous matches played between the two teams. The software is quite simple to use and can be used by any person with minimum training.

3.1.2 Overall Description

Product Perspective

The perspective of the product is to predict the result of ODI matches with the help of machine learning technology. The prediction will be of who will win the match.

Product Features

The major features of the product are summarized as follows:

- User will be able to enter predicting match details through a GUI made in flask module of python
- The product will predict which team is going to win.
- The product is based on supervised learning model. It will train previous data and will store the results in the database.
- It will take a dataset in the form of csv containing the history of all the ODI matches which include the teams' names, toss decision, toss winner, venue.

Design and Implementation Constraints

Some of the constraints that can limit the options available are:

- Accuracy: all the fields of the input form which is used to take predicting match details are required else prediction may go wrong.
- Language: English is kept as the only displayed language.

3.1.3 System Features

Input Predicting Match Details

Description and Priority

User will be able to enter match data a GUI interface built using flask library of python.

Stimulus/Response Sequences

- User first interact with the Home screen
- Home page contains three buttons namely statistics, prediction and analysis
- Statistics page will have two input fields player role and player name, by entering the valid data in those fields user can get statistics and graph representation of that player.
- Prediction page will have input form which requires match details which are team names, toss decision, toss winner, venue.
- Further processing will take place only after all the details have been entered else an error message will pop up.
- Analysis page simply shows the attribute graphs and their impact of features importance on the model.

Stimulus/Response Sequences of statistics

- The data collected from the form is stored in different variables.
- Then the player name is passed as attribute in database function which will search the players' data in database.
- For the graph section, players' role and name are passed as attribute and players' performance graph is created based on his performance in different years in the form of bar diagram.

Predicting the Result of the Match

Description and Priority

The product will predict which team will win the game.

Stimulus/Response Sequences of match prediction

- The data collected from the form is stored in a vector.
- Before predicting the match first the machine learning model is created based on previous data.
- Now the input details are sent to the predict function of our machine learning model class.
- According to that prediction will be done.

3.1.4 External Interface Requirements

User Interfaces

Python based library called flask is used for taking the input from the user which are (teams names, toss winner, toss decision, venue). Several exception are handled for creating this form, like if the user inputs incomplete information, or the user inputs some other field, the screen will display a message "Details are incomplete, please fill the complete details" and "The input of following field is invalid" respectively.

Hardware Interfaces

Display hardware device for user interface and displaying the result, mouse and a keyboard to interact with the tool are all required to be connected. Data will be maintained by a System administrator. The data will be in a csv format.

Software Interfaces

Tools used are python and flask. Scikit learn, Pandas, Matplotlib are the library used.

3.1.5 Other Nonfunctional Requirements

Performance Requirements

To generate the result with optimized efficiency and process it faster, a fast processor will be required so as to train the machine learning model quickly.

Safety Requirements

The software is an attempt to predict the result with maximum accuracy possible. Since the result is based on the current player performances, how the teams will play and probabilities from the test and train data, so 100% accuracy is impossible to achieve. Since the possibility of winning a game turns with every moment, which is what this software aims to predict accurately, the results can sometimes be not as the true outcome. We don't hold no responsibility if any harm or loss occurs due to the use of software results.

Security Requirements

The System Administrator will be responsible for keeping the database secured for access by only certain individuals. Any malicious link, or advertisements will not be shown and if it does, users are requested to click them at their own risk as they are a part of user's client app.

Software Quality Attributes

- a) Availability: The software will be available as a package.
- b) Maintainability: The software is easy to maintain and operate.
- c) Testability: The software is easy to use and test as previous records can be interpreted and they can be backtracked and tested.
- d) Correctness: The probabilities which are the output are accurate to certain level.

3.2 Software Requirement Specification

Project Name	Cricket Analysis and Prediction
Features	<p>For Prediction</p> <p>User can select the country name and opposite team.</p> <p>User should choose venue, toss winner, toss decision for prediction</p> <p>For playing XI selection, system.</p> <p>For player search,</p>
Operational Environment	<p>System should be operating in windows operating system.</p> <p>System will be developed in python programming language.</p> <p>System should operate on any browser.</p>
Assumption	<p>System should be error free.</p> <p>System should be user friendly so that it is easy to use for user.</p> <p>Data should be reliable and trustworthy.</p> <p>System should have accuracy as high as possible.</p>
Data Requirement	Data required for the system must contain recently updated data.
External Interface Requirement GUI	<p>System should provide good graphical interface for user.</p> <p>It allows the user to quick search about player's current data.</p>

Performance	<p>The purposed system that we are going to develop will be used to search the player's information as well. Therefore it is expected that the database would perform functionally all the time.</p> <p>The performance of the system should be fast and accurate.</p> <p>It shall handle expected and non-expected errors to prevent loss in information.</p> <p>The system should be able to handle large amount of data.</p>
Security Requirement	<p>System will use secure database.</p> <p>Normal users can just read data but they cannot edit or modify anything.</p>

3.3 Feasibility Analysis of the Project

A feasibility study is a preliminary study that investigates the information needs of prospective users and determines the resources requirements, costs, benefits, and feasibility of a proposed system. A feasibility study takes an account of various constraints within which the systems should implement and operate. The main objective of the feasibility study is to determine whether the project would be feasible in terms of economic, technical and operational feasibility. The results of this analysis are used in making the decision whether to proceed with project or not. Feasibility study includes:

3.3.1 Economic feasibility:

Economic feasibility attempts to deal with the costs of developing and implementing a new system, against the present day condition. This feasibility study gives the top management economic justification for the new system. The system can be used in any device which supports web browsing. This system can replace many other apps to use for predicting the outcome of a ODI cricket match. The creation of the application is also not costly.

3.3.2 Operational feasibility:

Operational feasibility is a measure, how well a proposed system solves the problems, and take advantage of the opportunities identified during scope definition. A system with easy interface will always help the user to easily use the system. Nowadays everyone knows how to surf the internet, so the system can be easily operated by the people without having high experience. Thus, the system is said to be operationally feasible.

3.3.3 Schedule Feasibility:

A project will fail if it takes too long to be completed before it is useful. Typically, this means estimating how long the system will take to develop, and if it can be completed in a given time period using the methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is. Given our technical expertise, are the project deadlines reasonable? Some project is initiated with specific deadlines. It is necessary to determine whether the deadlines are mandatory or desirable.

3.4 Cost Analysis

Apart from the laptop or the personal computer running the prediction software, there is no other hardware we have used in our project. All the python libraries used and python itself are freely available too on the internet.

For the choice of the laptop or any machine we recommend having a good processor (at least core i3). A processor below that may take much more time as in i3 itself, a neural network may take 30 minutes to train and 15 minutes for logistic regression.

3.5 Assumptions and Constraints

1. It is assumed that the python (3.6) is installed in the system alongside the following libraries -

- Pandas
- Scikit learn
- Flask
- Matplotlib
- Pickle
- CSV
- Sqlite3

2. There should be a System administrator whose job is to keep the dataset updated before a user runs the program.

3. The system should have sufficient processor speed so as to train the huge amount of dataset on different machine learning algorithms.

4. Design and Analysis

4.1 System Design

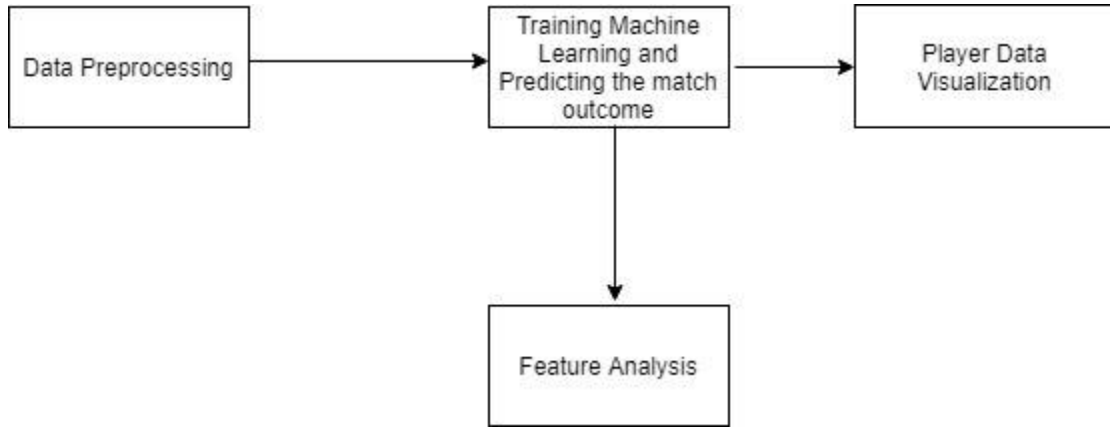


Figure 4.1 System Design

4.2 Activity diagram

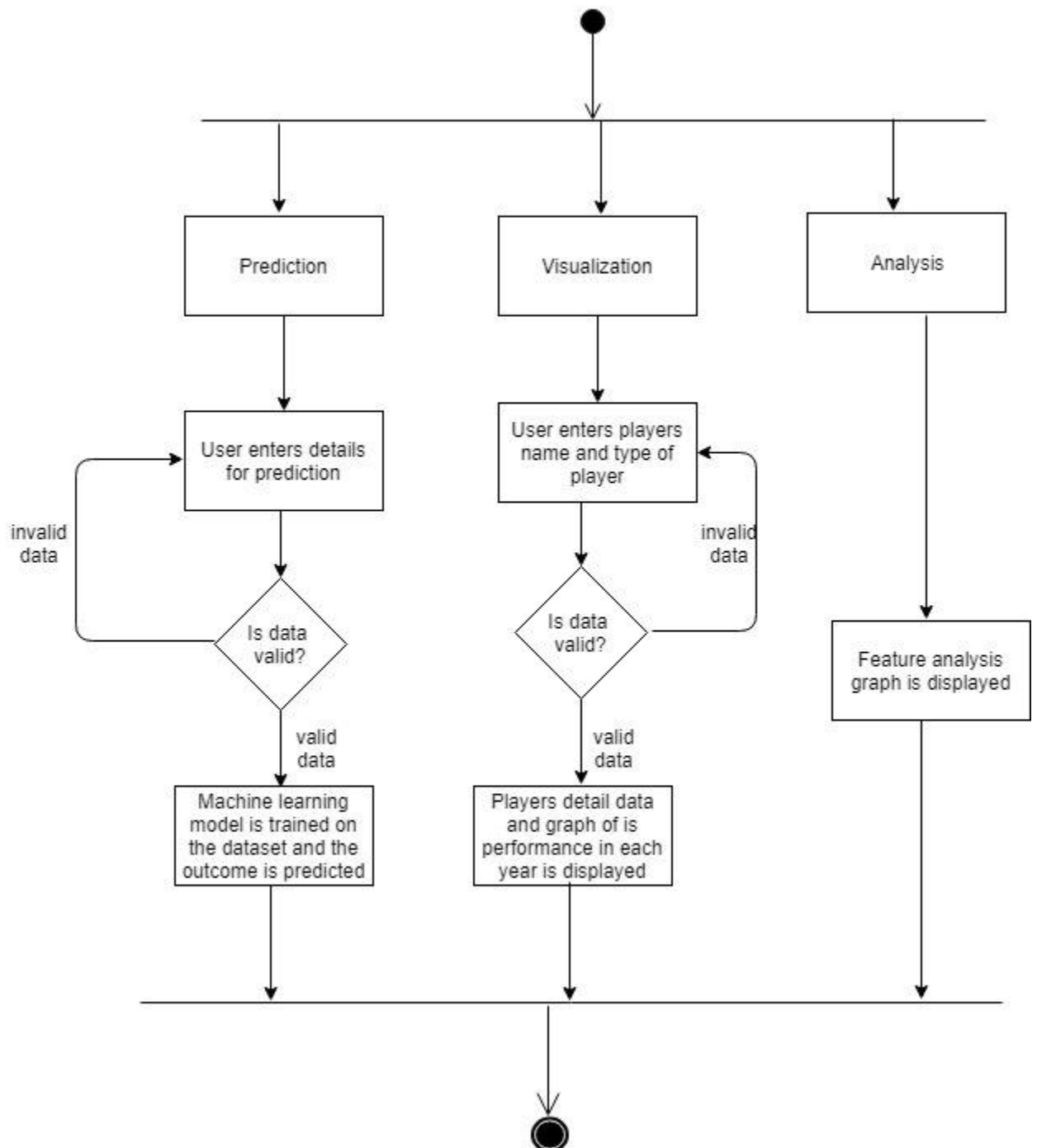


Figure 4.2 Activity Diagram

4.3 Class Diagram

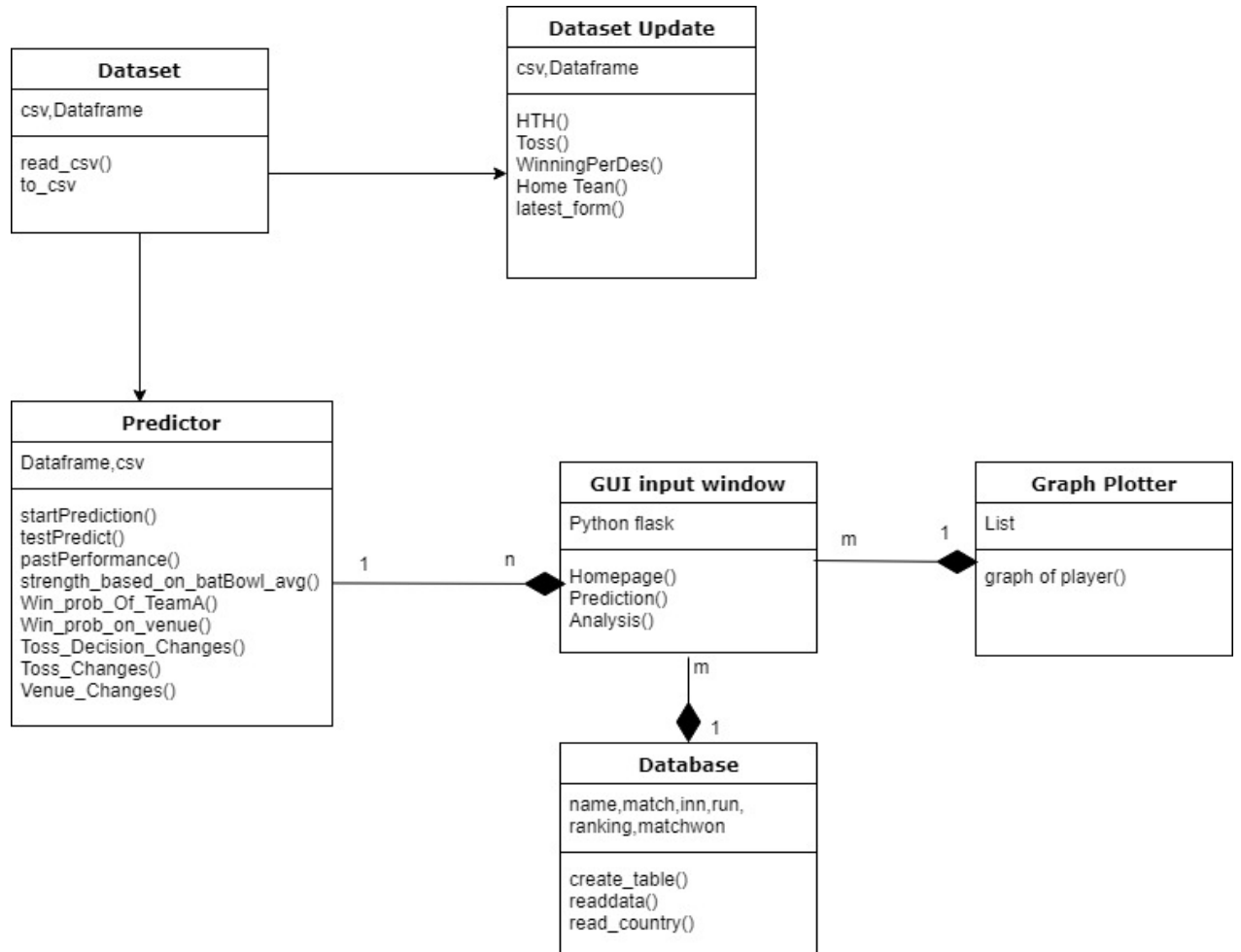


Figure 4.3 Class Diagram

4.4 Data Flow Diagram (level 0)

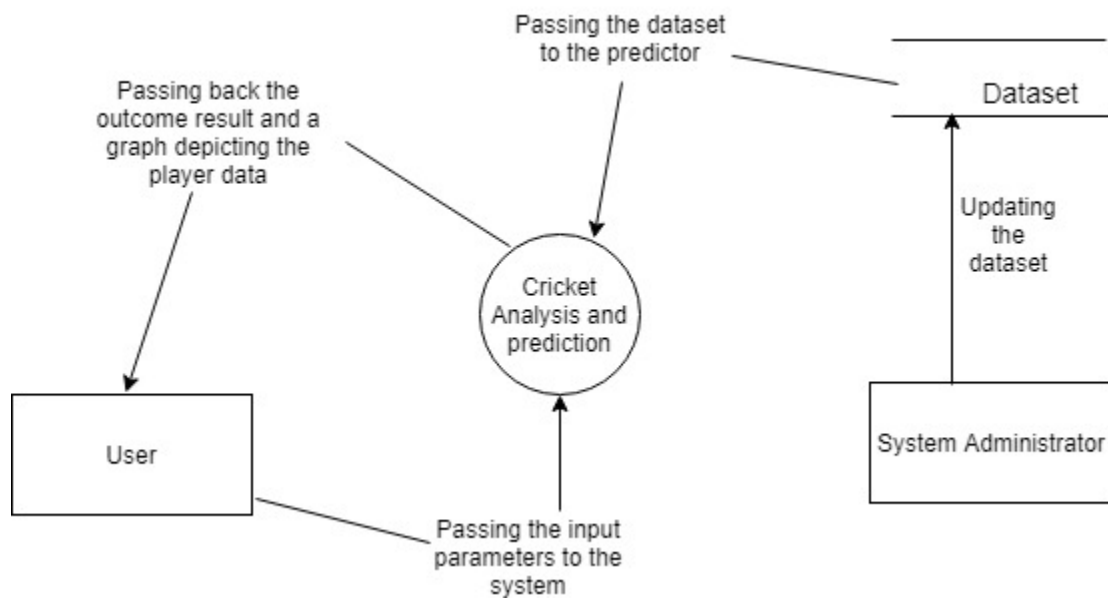


Figure 4.4 Data Flow Diagram (level-0).

4.5 Sequence Diagram

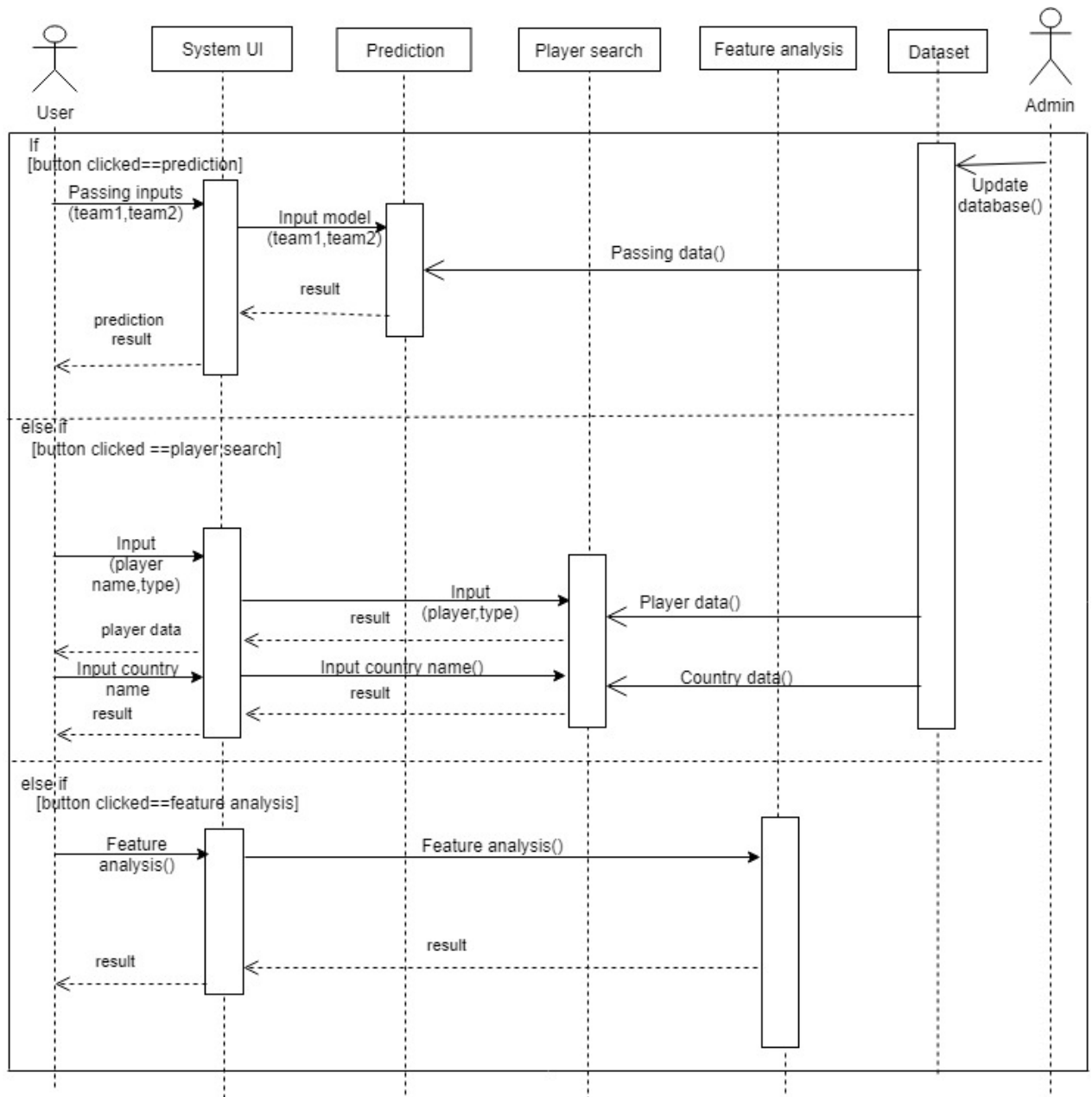


Figure 4.5 Sequence Diagram.

4.6 Use case Diagram

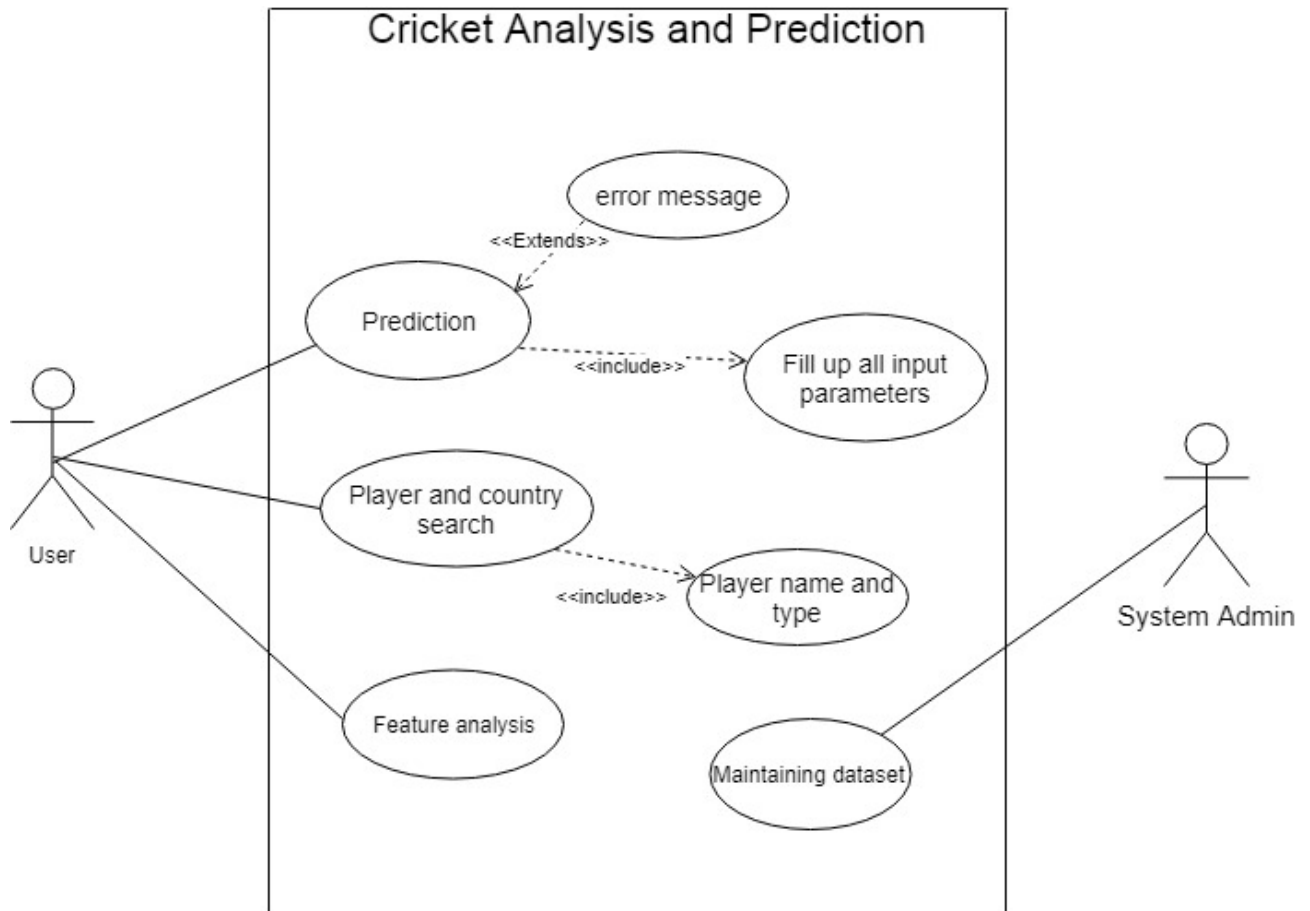


Figure 4.6 Use Case Diagram.

Actors:

User: They are the external person who can visit the webpage.

System Admin: It refers to the developer who can change or modify the system. Admin can also update dataset and database.

Use Cases:

model. If user enters invalid input system shows error message hence error message is an extend case for prediction. User must fill all input field to see the result thus it is an include case for prediction.

Player and Team search: User can search for players and teams data by providing players and country name. player name, type and country name are the necessary attributes for search hence it is an include case for search.

Feature analysis: User can see the different photos of feature analysis just by clicking on feature analysis button.

Dataset maintenance: System admin can modify,insert or remove data from database.

4.7 Entity Relationship Diagram

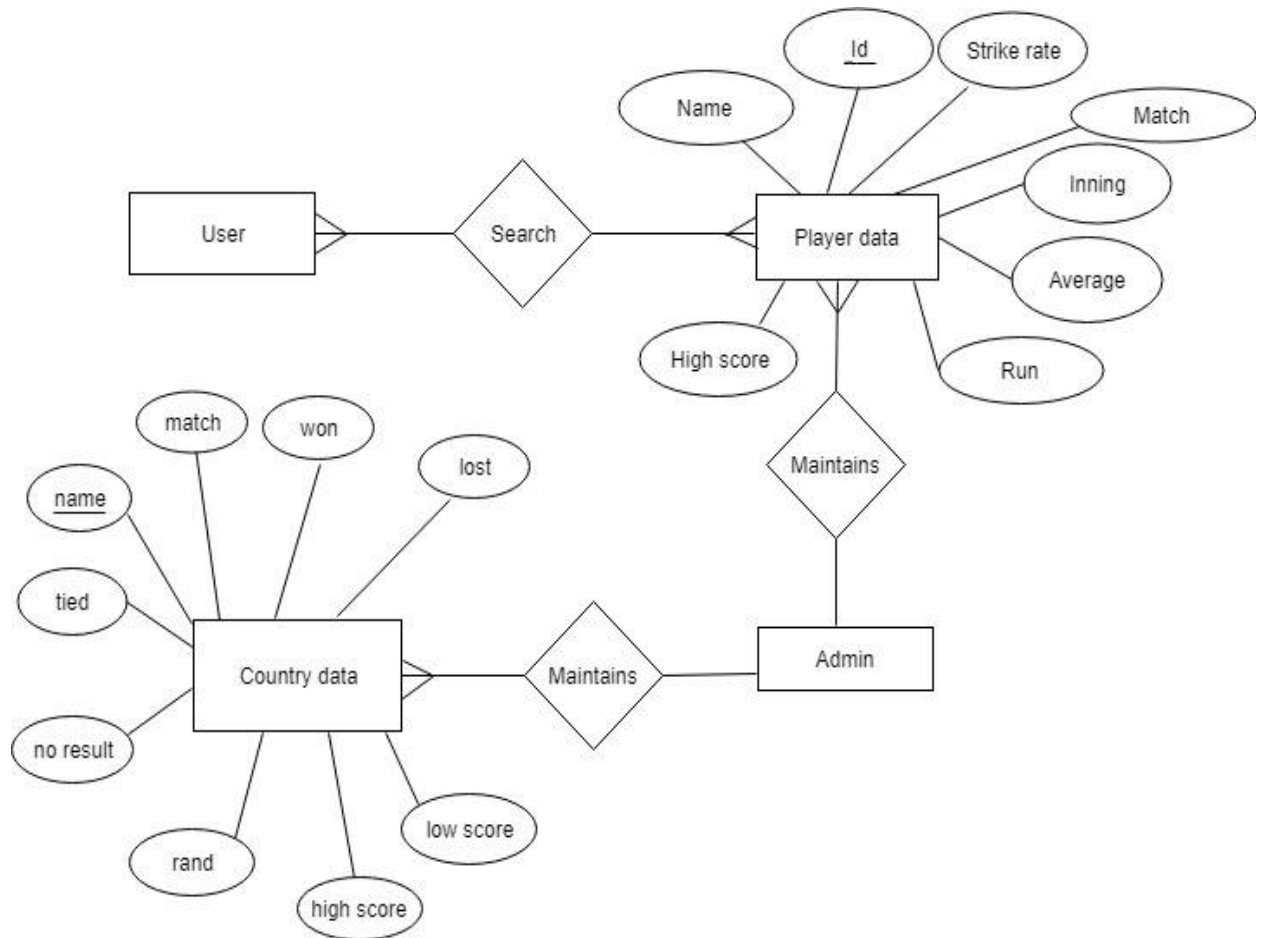


Figure 4.7: ER Diagram

4.8 System Flow

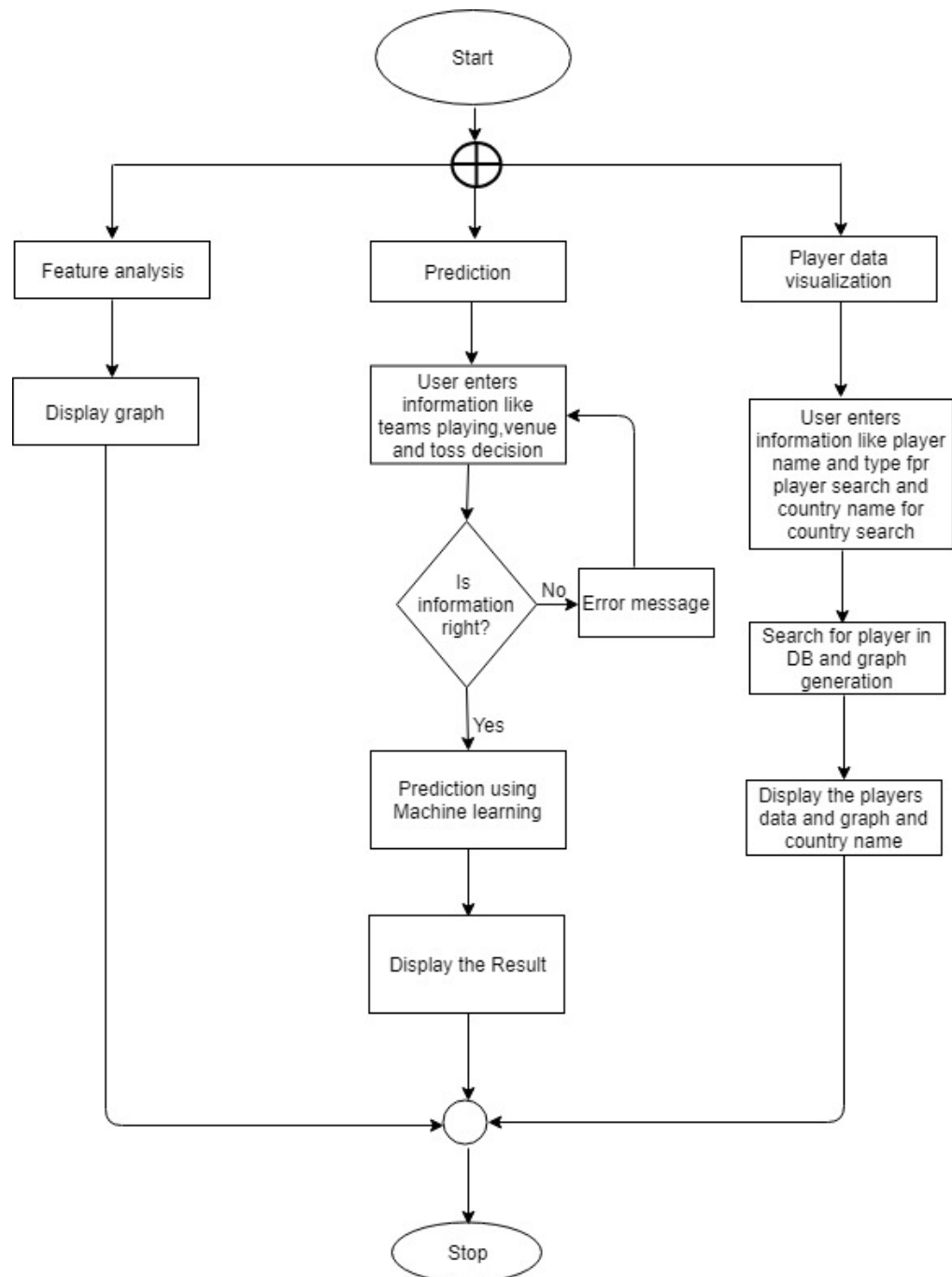


Figure 4.8 System Flow

5. Methodology

We initiate our project by fetching data from sites like ESPN and Cricksheet. The data we obtained is in the form of csv file where each file describes each match details. Then we generated python scripts to club the data entries of each and every match. We then applied feature selection to select features like teams involved, match venue, toss decision and winner of the match. We then used feature generation to create more important features like strength of each team, performance of each team in past, probability of team batting first and winning probability of particular team at a specific location based on all the previous matches played between those two teams at the specific venue. The strength and performances of each match are calculated in terms of relative strength and relative performances. The strength of each team in a particular match is calculated based on the batting of each player in that match. Further the team batting average is calculated by taking the mean of batting averages of all the players of the team and the. After getting the strength of each team we subtract the strength of team A with team B to get relative strength of overall match. If relative strength is positive it means team A is stronger than team B. If negative it means team B is stronger than team A. Else both are equally strong. Along with strength we have also considered the past few match performances of each team as it tell us whether the team is in good form or not. The performances are calculated by taking mean batting average of past all matches of each team. After getting individual team performances we then calculate the relative performance of the team by subtracting team A performance with team B performance. If the relative performance is positive it means team A is in better form than team B. If negative then it means team B is in better form. Otherwise both are in equal form. The magnitude tells us the amount with which one team is better than other.

After preprocessing and feature generating we used the machine learning to generate models which can be used to predict the results. We used different models and calculated the accuracy from each model and finally selected the one with best accuracy. We used Random Forest, Logistic Regression and SVM. Out of all these we

used logistic regression as it gave us the better accuracy among all other models. After generating models we developed a GUI in python using flask module so that user can easily interact with our system.

Software Development Life Cycle

In our project, we used iterative model as a SDLC. Iterative Model is too a part of Software Development Life Cycle. It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces. Our project can be illustrated by following figure as SDLC



Fig: Iterative model in our project

The 7 steps that together constitute in our project are:

- **Identify the Problem:**

In this phase we prepare SRS document to find out functional requirement of project, accuracy we want to achieve and data sources from where we want to achieve data set.

- **Identify available data sources:**

During this step we extracted data from different source using web scraping in python then we analyze the data to identify the quality of data. Finally we performed data cleaning to remove unnecessary data to fill missing value using pandas library.

- **Identify if additional data sources are needed:**

During second phase we only collected data for prediction only so after consulting with our supervisor we thought about player data visualization hence we also collected data for different players and data of different countries too.

- **Statistical analysis:**

In this phase we performed feature analysis to find out the best attributes that may affect the result of match outcome. We found that home and away team, toss winner, toss decision and venue are the most important factors that we should consider in our prediction model. We also analyze different machine learning algorithms like SVM, logistic regression and random forest to find out the best algorithm for our project and we reach to the conclusion to use logistic regression due to its best accuracy among three.

- **Implementation, development:**

Then finally we develop model for logistic regression and we train the model using our data set. Then we found out accuracy based on test data.

- **Communicate result:**

During this step we consult with our supervisor about the project then he suggest us to include player data visualization and team record also in the project.

- **Maintenance:**

Then we also include data visualization in our project. For this we created two tables for players and teams data in sqlite database. We also generate graph of players performance over different years which can be seen by user in player data visualization section of our project.

5.1 Design Phases

The whole project is divided into seven phases.

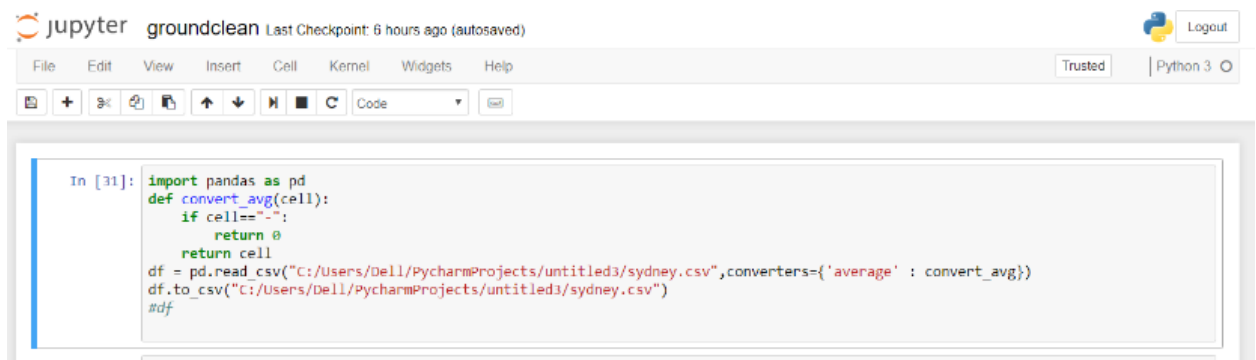
- Web Scraping
- Data Visualization
- Data Preprocessing
- Feature Generation
- Feature Analysis
- Model Generation and its testing
- GUI Development for better user interaction.

Web Scraping

Web scraping, data extraction method from web, is used to harvest data from espncriinfo.com for data of players performance in different years. We used beautiful soup library of python to scrap the data. At first we find out the runs scored by different players across different years in his career and then write those data in csv file. We only scrap data of 37 players from 10 different test playing nation.

Data Preprocessing

Initially we use pandas library to remove unnecessary data and fill the missing values which can be shown in following figure.



```
In [31]: import pandas as pd
def convert_avg(cell):
    if cell=='-':
        return 0
    return cell
df = pd.read_csv("C:/Users/Dell/PycharmProjects/untitled3/sydney.csv", converters={'average': convert_avg})
df.to_csv("C:/Users/Dell/PycharmProjects/untitled3/sydney.csv")
#df
```

The initial step is to perform the data preprocessing of the initial data. Basically our initial data consists of the following columns -

- Match ID
- Toss
- Toss Decision
- Team A
- Team B
- Date of the match
- Venue
- Winner

Of these data there are few missing entries which are solved by taking mean of all the values of that specific attribute. Then we apply feature generation techniques to add features which tells the relative strength and relative performance of team A. There are 4 features which are added.

TeamA_win_prob – This quantity is obtained by considering the 10 recent matches of the 2 teams in the original row and finding how many of these matches have team A won. After that, the matches won by team A is divided by the total 10 latest matches played by both the teams (probability of winning).

Relative Performance– The performance of a match is calculated by subtracting the average form of the individual teams over the last 10 recent matches. The quantity form basically represents the batting average of the whole match. The formula basically becomes:

$$\text{Relative Performance} = \text{Team A Batting Avg} - \text{Team B Batting Avg}$$

Strength – It basically tells the winning probability based on team composition in that particular match. It takes into account the batting averages of all the 11 players and bowling averages of all the 11 players and then subtract batting average with bowling average to give the strength of that team. Then we subtract team A strength with team B strength to get relative strength of team A.

$$\text{Relative Strength} = \text{Strength of Team A} - \text{Strength of Team B}$$

Training the prediction model - After our dataset is ready and input has been taken from the user, we apply machine learning models and determine the outcome. The additional 4 features are also calculated for the input match in the same way we calculated for each row in the match during data preprocessing. Then we form a dictionary of all the inputs and finally convert that dictionary to pandas dataframe so that it can be processed by predict function. Now out of all the models we have used logistic regression because as you can see in the following diagram how it gives the best result among all the other classifiers. We pass the test dataframe to logistic regression model with parametric value of random state set to one. We then get the winner of the match as a single entity. We display the other parameters in command line interface while we pass the winner value to our front end file where it stores its value in a label container and then displays that container. The user can change parametric values and know the results without restarting the application.

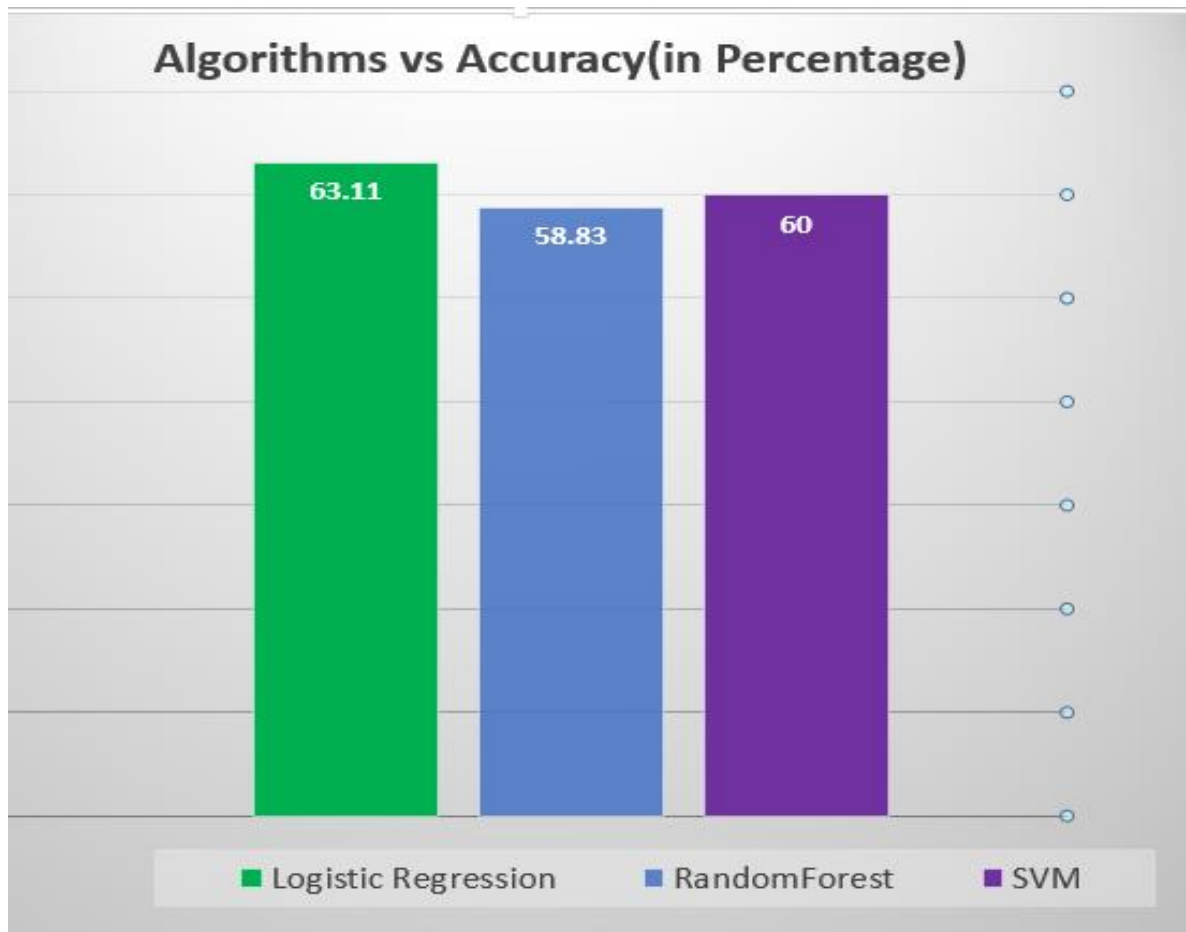


Figure 2.3 Model Selection

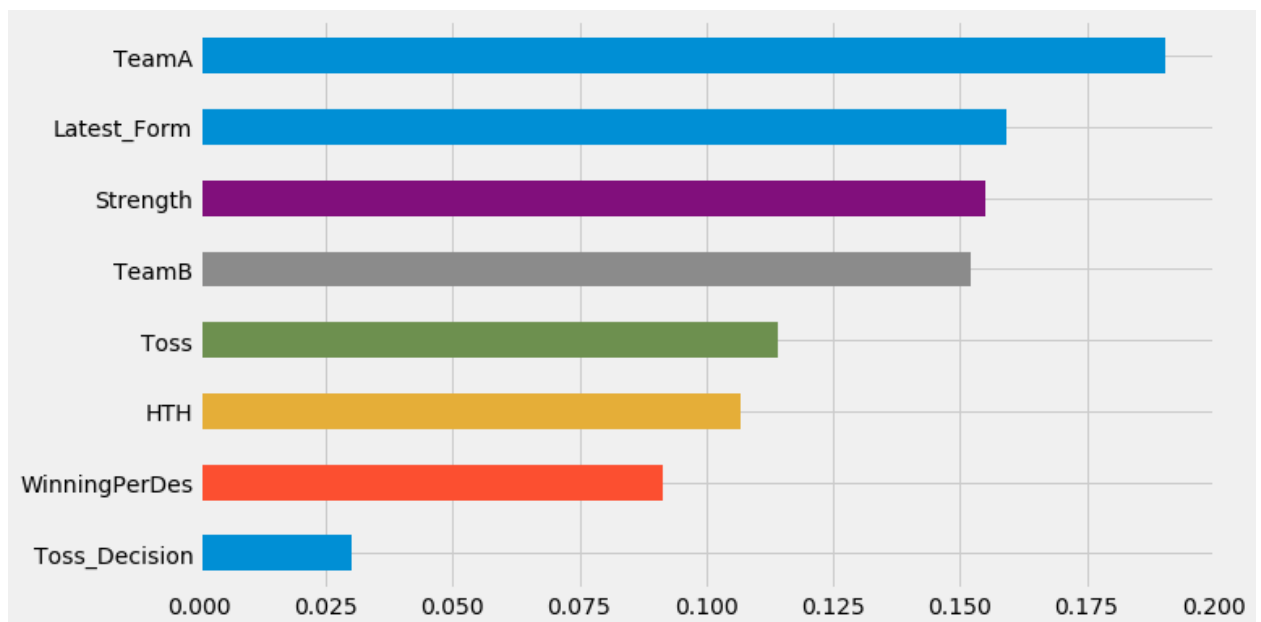
Feature Generation

We then used feature generation to create more important features like strength of each team, performance of each team in past, probability of team batting first and winning probability of particular team at a specific location based on all the previous matches played between those two teams at the specific venue. The strength and performances of each match are calculated in terms of relative strength and relative performances. The strength of each team in a particular match is calculated based on the batting of each player in that match. Further the team batting average is calculated by taking the mean of batting averages of all the players of the team and the. After getting the strength of each team we subtract the strength of team A with team B to get relative strength of overall match. If relative strength is positive it means team A is stronger than team B.

If negative it means team B is stronger than team A. Else both are equally strong. Along with strength we have also considered the past few match performances of each team as it tell us whether the team is in good form or not. The performances are calculated by taking mean batting average of past all matches of each team. After getting individual team performances we then calculate the relative performance of the team by subtracting teamA performance with teamB performance. If the relative performance is positive it means teamA is in better form than teamB. If negative then it means teamB is in better form. Otherwise both are in equal form. The magnitude tells us the amount with which one team is better than other.

Feature Selection and analysis

Feature selection is a process where you automatically select those features in your data that contribute most to the prediction variable or output in which you are interested. Three benefits of performing feature selection before modeling your data are it improves accuracy of model, reduces training time and increasing the models interpretability by revealing the most informative factors that drive the model's outcomes. In our dataset we performed feature selection and found that home team has more score followed by latest form of the teams. The final output of feature importance with outcome variable i.e winner is shown below in the graph.



Model Generation and its testing

After preprocessing and feature generating we used the machine learning to generate models which can be used to predict the results. We used different machine learning models and calculated the accuracy from each model and finally selected the one with best accuracy. We used decision tree, random forest, logistic regression and SVM. Out of all these we used logistic regression as it gave us the better accuracy among all other models. For future match prediction we first take the inputs from user via GUI which is created using flask. Then we first convert those inputs in numeric form like 1 for team A and 0 for team B and generate other important features using the modules we have used in phase 2. Then we convert these whole inputs into pandas data frame and pass it to model for prediction. The model does the processing and gives us the winner of the match.

Data Visualization

Data visualization is used for players data display and graph display purpose. For players data display section we stored data in sqlite3 database of python by creating a table playerinfo. Where the table contains attribute like players name, matches he played, innings, runs scored by that player, highest score, average score and strike rate

of one day international matches. Thus stored data is extracted from database and displayed to the user when he enters the player name. For graph plotting we used python library matplotlib and plotted the bar diagram of the player according to his performance across different years and finally displayed to the user.

GUI Development for better user interaction

In this phase we developed a simple user interface so that a user need not to go to command prompt for entering all the required inputs. The interface has been developed using python module flask. The interface contains three parts: Data visualization, Prediction and Feature analysis. In data visualization user needs to enter players name and his role i.e bowler or batsman then finally system displays player past data and his performance over different years. For prediction , user have to provide information like team1,team2,venue,toss winner, toss decision by filling the form then the form developed also validates if the provided input is valid or not. If the entered information is invalid it returns an error message. Also if the user doesn't give all the required inputs then it displays an error message to fill all the required entries. Thus it is easy and better way of getting all the valid inputs.

5.2 Algorithm Implementation

5.2.1 Logistic Regression and Model Evaluation

Logistic regression is a popular machine learning technique for analyzing a dataset in which there are one or more independent variables that determine an outcome. Logistic regression can be binomial, ordinal or multinomial. Binary logistic regression deals with situations in which observed outcome for a dependent variable can have only two possible types, "0" and "1".

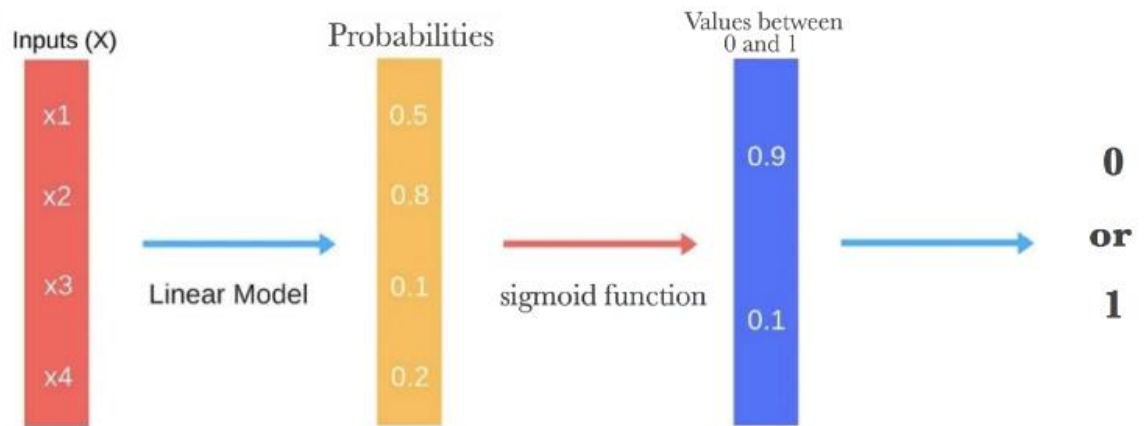
Suppose that we want to know if a student will pass the final university examination or not. The prediction of passing or failing of students depends on various independent variables or features like how many hours (s)he study per day, his/her performance on an entrance examination, his/her previous academic score and many more. In this case, success or pass of a student is coded as 1 and fail as 0.

How it works

Logistic Regression measures the relationship between the dependent variable (our label, what we want to predict) and the one or more independent variables (our features), by estimating probabilities using it's underlying logistic function.

These probabilities must then be transformed into binary values in order to actually make a prediction. This is the task of the logistic function, also called the sigmoid function. The Sigmoid-Function is an S-shaped curve that can take any real-valued number and map it into a value between the range of 0 and 1, but never exactly at those limits. This values between 0 and 1 will then be transformed into either 0 or 1 using a threshold classifier.

The picture below illustrates the steps that logistic regression goes through to give you your desired output.



If $X \in \mathbb{R}^n$ is a feature input vector with given parameters: weights $w \in \mathbb{R}^n$, threshold $b \in \mathbb{R}$, and h is estimated output (probability of getting correct result with given X) then $h = p(y=1|X)$ where $0 \leq h \leq 1$, y is correct output value, and n is number of features. The observed/estimated value can be calculated as follow as:

$h = g(W^T + b) = g(z)$ where $g[z]$ is a sigmoid function. Since $h = (W^T + b)$ is a linear function ($ax+b$), but we are looking for a probability constraint between $[0,1]$, so we used a sigmoid function which is bounded between $[0,1]$.

5.2.2 Sigmoid Function

A function takes inputs and returns outputs. To generate probabilities, logistic regression uses a function that gives outputs between 0 and 1 for all values of X . The equation is as follows

$$g(z) = \frac{1}{1 + e^{-z}}$$

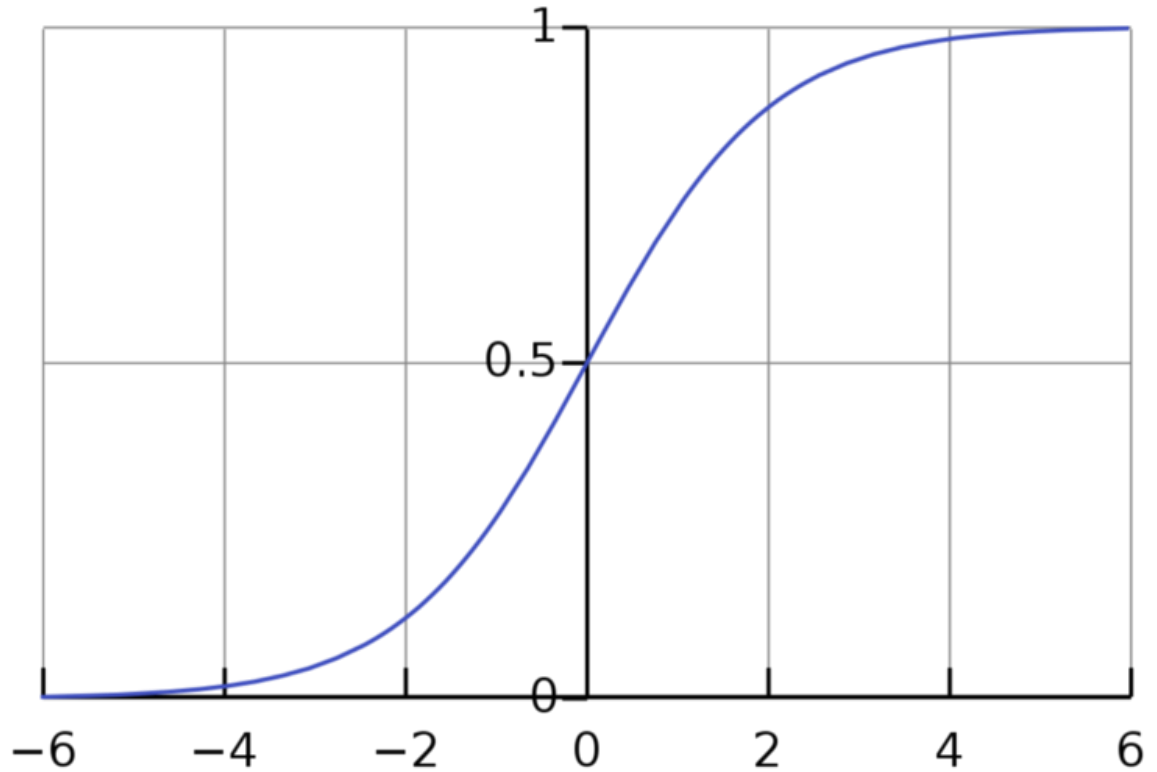


Figure 3.4.1: Graph Plot of Sigmoid Function

5.2.3 Loss function

In a simple way we can say that the loss function concerned with how good is our estimation with respect to the desired output. In other words, this function measures the discrepancy between the prediction estimated and the desired output.

$$L(h, y) = -(y^i \cdot \log h^i + (1 - y^i) \log(1 - h^i))$$

5.2.4 Cost Function

The cost function is the average of the loss function of the entire training set. We are going to find the parameters w and b that minimize the overall cost function. Functions have parameters/weights and we want to find the best values for them. To start we pick random values and we need a way to measure how well the algorithm performs using those random weights. That measure is computed using the cost function, defined as:

$$J(w, b) = \frac{1}{m} \sum_{i=1}^m L(h, y)$$

5.2.5 Gradient Descent

Our goal is to minimize the loss function and the way we have to achieve it is by increasing/decreasing the weights, i.e. fitting them. The question is, how do we know what parameters should be bigger and what parameters should be smaller? The answer is given by the derivative of the loss function with respect to each weight. It tells us how loss would change if we modified the parameters.

$$\frac{\delta J(W)}{\delta W} = \frac{1}{m} \times X^T (h - y)$$

Fig 4: Partial Derivative

We calculate w, b in such a way that cost function would minimize. We repeat the process until the cost function converges reach the global optimum.

$$W = W - \alpha \cdot dw$$

$$b = b - \alpha \cdot db$$

where α is learning rate.

Learning rate(α)

Learning rate, α controls how biggest step we take on each iteration. If the learning rate is low optimization process will take a lot of time because steps towards the minimum of the loss function are tiny. If the learning rate is high, then training may not converge or even diverge. Weight changes can be so big that the optimizer overshoots the minimum and makes the loss worse. That's why choosing best learning rate was the biggest challenge.

Algorithm

The general algorithm for logistic regression is briefly described below:

1. Define the model structure (such as number of input features)
2. Initialize the model's parameters
3. Loop:
 - i. Calculate current loss
 - ii. Calculate current gradient
 - iii. Update parameters

Model Evaluation

After implementing the different functions as listed above, we fitted training data in the model on for testing the accuracy of the model. This was done by fitting train predictor dataframe which contains features like Toss, Venue, latest form etc. as training data and train target dataframe which contains Winner as testing data. The trained data was saved to disk using pickle library .The trained data can be loaded next time without retraining the data. Picking a learning rate = 0.1 and number of iterations = 3000, the

algorithm classified almost all instances successfully. The accuracy of model was found to be 63.11%. with the following weights of the features used for prediction.

Test Accuracy is : 63.11926605504588

The weights of Toss, Toss_Decision, Venue, HTH, WinningPerDes, Strength, latest_form, Winner are :

[0.19435048 -0.00229934 -0.59509459 0.36124937 0.22027935 -0.67076883
-0.02803511 0.15410675]

Fig: Test Accuracy and Weights to the features used

6. Testing and Evaluations

6.1 Unit Testing

The system was developed in many units namely prediction model, data visualization, GUI development. After completion of development of each unit they were tested independently to check the correctness of each unit.

6.2 Integration Testing

After each unit were tested independently, they were then integrated to form the whole system. The integration was not straight forward. Prediction model was first developed as a windows application but while converting it to a web app we faced various problem related to programming language. Integration testing was performed in two phases:

- i) **Black box testing:** after performing some testing we concluded that the system requirements were met.
- ii) **White box testing:** We also came to know that the internal working logic of the system were correct and acceptable.

6.3 Acceptance Testing

This testing was performed to check whether the system meets the initial requirement or not. At first we tested the system ourselves then the system was tested by supervisor for acceptance.

6.4 Inferences Drawn

- The winning probability of a particular team greatly depends on its batting average, bowling average and its performance in past few matches.
- Logistic Regression algorithm gave the best accuracy among all the models.

7. Limitations and Future Enhancement

The major limitations of our web application are mentioned as follows:

- i) Our application focuses on data from the previous matches but the data is static and do not change with matches played every day.
- ii) Only the players from the current playing squads can be searched for player analysis.

7.1 Future Direction

The project currently takes into account the important factors from previous matches and then predicts the result before the match starts. But we can also take the ongoing match details to predict the result. So the future work will be to combine both the previous data with ongoing match data to have much more better results. Also the prediction can be extended to not only predict the winner of the match but also the expected runs to be scored by both teams.

8. Conclusion

With growing interest in the sport of cricket over the past few years, a need for tool which can predict the results of the match in advance has to be developed. Though there are lots of tools available in the market, there accuracy lacks in the way they take factors into consideration. To give an edge to those existing classifiers, our project aims to also take into consideration some important factors like team composition, performance of players in past, batting and bowling averages of the players in each team and winning probability of team batting first at a specific venue against a specific opponent. All these important factors along with toss and venue has taken into account and a classifier has been generated to give better results. Along with these a user interface has also been generated so that even a layman can interact with our system with ease.

9. Project Metrics

9.1 Challenges Faced

- In deciding the best parametric value for different classifiers while generating the model.
- The second was a minor problem of designing and coloring the GUI using flask
- Unknowingly Rearranging of index values while sorting the dataset on a particular attribute using pandas dataframe.
- Implementing algorithm from scratch in our dataset
- Testing accuracy of model

9.2 Interdisciplinary Knowledge Sharing

In this project, we have used the principles of machine learning which include preprocessing the data, applying machine learning models to predict the outcome and data visualization using graph. The programming language used is python in which we have used Data structures like lists, dictionaries and Dataframe. The following python libraries have been put to use in project -

- Scikit Learn (Applying machine learning models)
- Flask (making the GUI)
- Matplotlib (For data visualization)
- Pandas (For data analysis)
- Pickle (For saving data in the disk)
- CSV (for reading the dataset in form of csv)

Apart from that, principles of Software engineering have also been applied to make various UML diagrams like DFD, sequence diagram, class diagram, activity diagram, use case diagram.

9.3 Gantt Chart

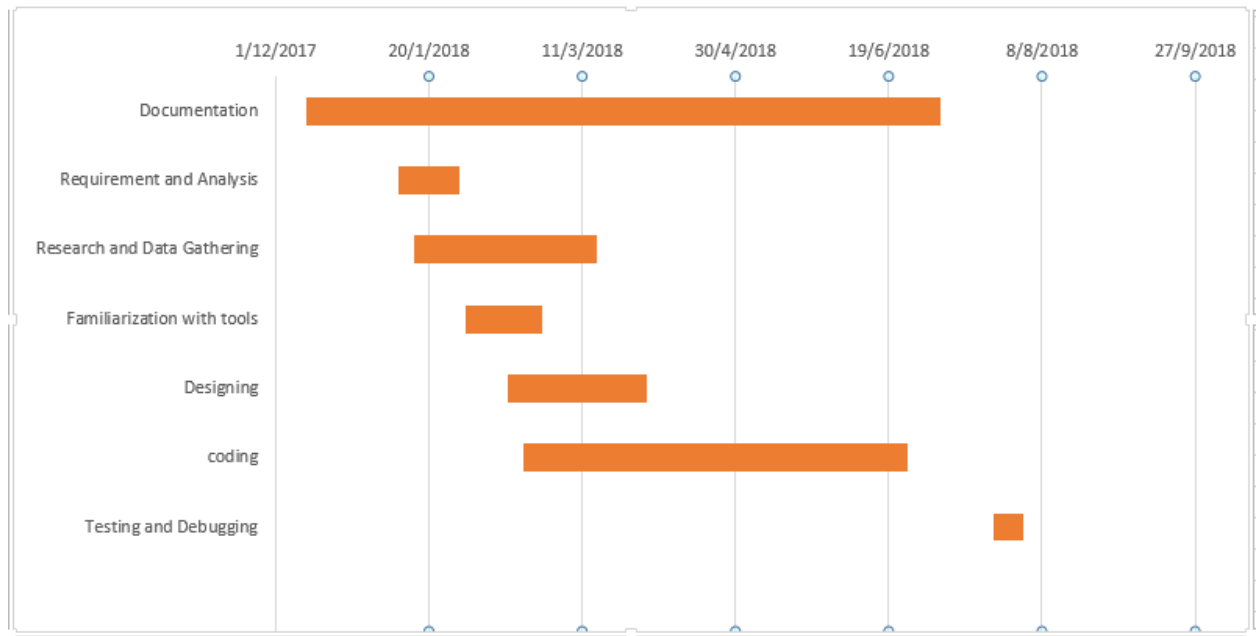


Fig: GANTT Chart

9.4 Responsibility Assignment Matrix

	Jeevan	Madhu	Pradeep	Prakash
Requirement Analysis	C, I	C	R, A	R, I
Designs	C, R	R, A	C, R	R, A
Coding	I	R, A	I	C,R
Testing and Integration	C,R	A,I	R	A
Deployment	R	A	I	A

Where:

R: Responsibility

I: Inform

C: Consult

A: Accountable

10. References

- [1] Ananda Bandulasiri Manage (2015). Predicting the Winner in One Day International Cricket, Sam Houston State University.
- [2] A Kaluarachchi, SV Aparna (2010).A classification based tool to predict the outcome in cricket.
- [3] Bandulasiri, A. (2008). Predicting the winner in one day international cricket. Journal of Mathematical Sciences & Mathematics Education, 3 (1), 6-17.
- [4] Madan Gopal Jhavar, Vikram Pudi (2016). Predicting the Outcome of ODI Cricket Matches: A Team Composition Based Approach. Report No: IIIT/TR/2016/-1.
- [5] Sasank Viswanadha, Kaustubh Sivalenka, Madan Gopal Jhavar,Vikram Pudi(2006). Dynamic Winner Prediction in Twenty20 Cricket: Based on Relative Team Strengths.
- [6] Monalisha Pattnaik, Anima Bag (2015). Fitting of Logistic Regression Model for Prediction of Likelihood of India Winning or Losing in Cricket Match.
- [7]. Munir, Fahad, Hasan, Md. Kamrul, Ahmed, Sakib, Md. Quraish, Sultan (2015). Predicting a T20 cricket match result while the match is in progress.
- [8] Parag Shah (2017). Predicting Outcome of Live Cricket Match Using Duckworth-Lewis Par Score.

- [9] Parag Shah, Mitesh Shah (2015).Predicting ODI Cricket Result.
- [10] Stephan Gray and Tuan Anh Le (2002). How to fix a one day international cricket match.
- [11] Swetha, Saravanan.KN (2017). Analysis on Attributes Deciding Cricket Winning.
- [12] Tejinder Singh, Vishal Singla, Parteek Bhatia (2015).Score and Winning Prediction in Cricket through Data Mining.
- [13] Viraj Phanse, Sourabh Deorah (2011). Evaluation and Extension to the Duckworth Lewis Method: A Dual Application of Data Mining Techniques.
- [14] Haseeb Ahmed, Licheng Wang (2017). Prediction of Rising Stars in the Game of Cricket.

Appendices

Some Snapshot of the project

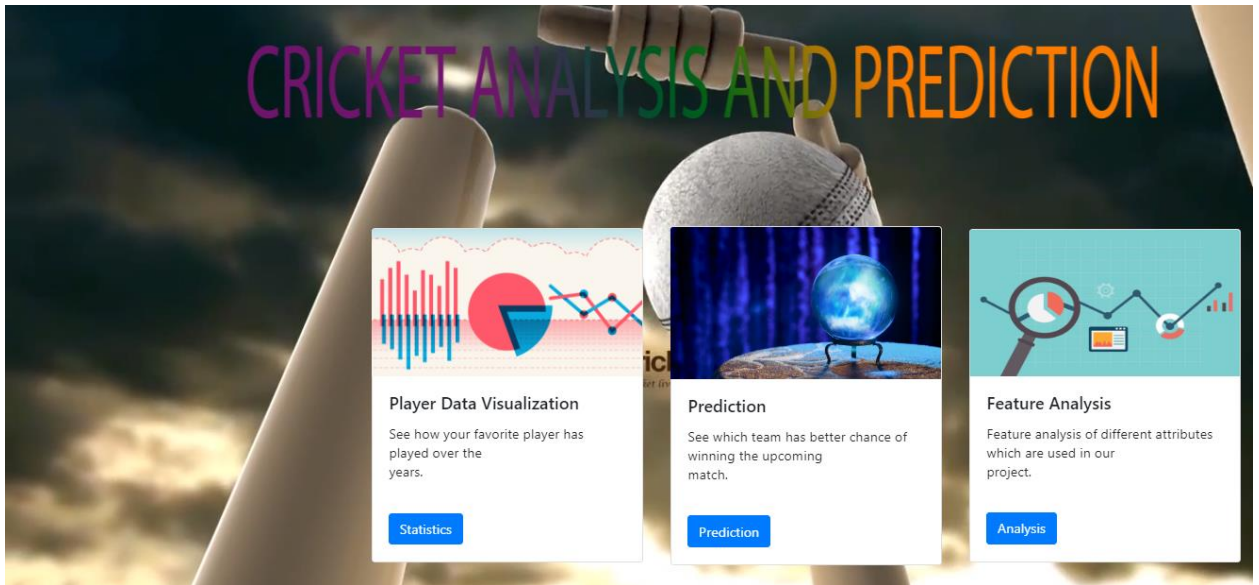
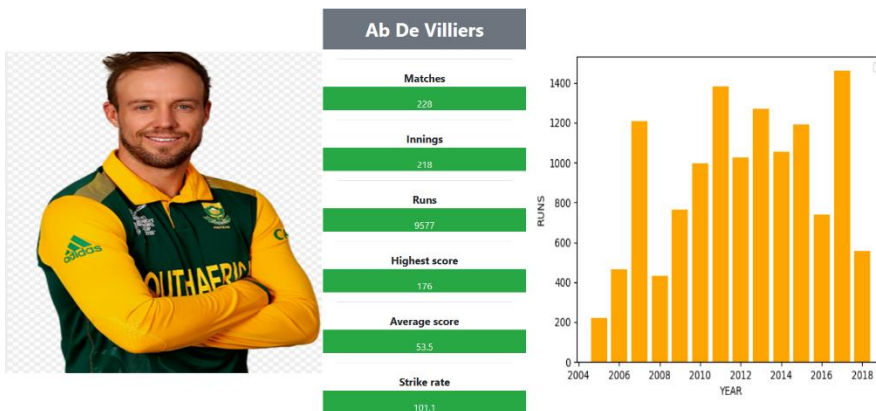


Figure A.1 : User Interface Home page



"A batsman of breathtaking chutzpah and enterprise. A cricketer with overflowing talent and the temperament to back it up. A fielder able to leap tall buildings and still come up with the catch - and who will happily move behind the stumps into the wicketkeeper's spot if needed. A fine rugby player, golfer, and tennis player. AB de Villiers has emerged as one of South Africa's greats. De Villiers is a 360-degree batsman who can hit any ball, anywhere, against any bowler. Indeed, his range of inventive shots has grown as his career has unfolded. He has been ranked among the top Test and ODI batsmen in the world and has established a cult-like following in T20 cricket, where his performances in the IPL have earned him a legion of Indian fans."

Figure A.2: Statistics Result Page

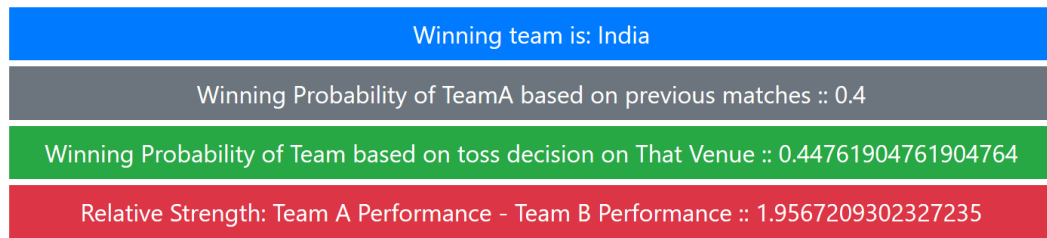


Figure A.3 Prediction Result Page

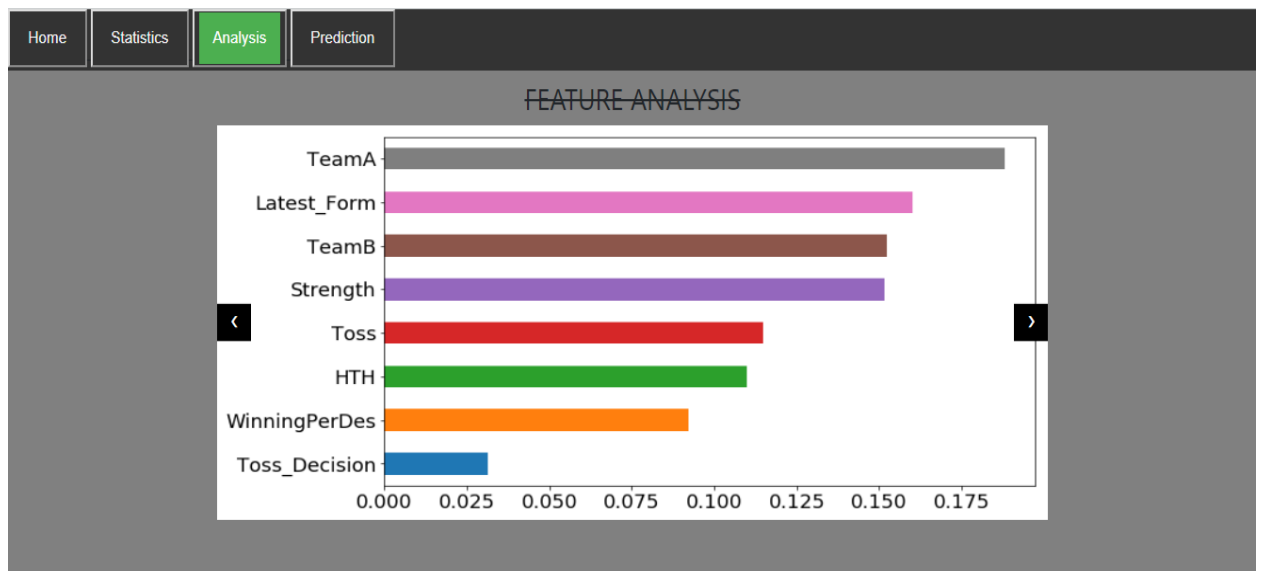


Figure A.4: Analysis Result Page



Figure A.5: Country Data Result Page

Home Statistics Analysis Prediction

ROLE
batsman

PLAYER NAME
[Redacted]

COUNTRY NAME
NEPAL


Submit

SubmitCountry

Cricket
Cricket lives here

Figure A.6: Data search form

Home Statistics Analysis Prediction

 CHAMPIONSHIP

CRICKET PREDICTION

FIRST TEAM *

AUSTRALIA

SECOND TEAM *

INDIA

TOSS

AUSTRALIA

VENUE

TOSS DECISION *

bat

Submit →

Figure A.7: Prediction form