

IPL STATISTICAL ANALYSIS

CSE3020 – DATA VISUALISATION

PROJECT BASED COMPONENT REPORT

Submitted by:

Naman Jain 19BCE2523

Arsh Temani 19BCE2591

Ishita Sachan 19BCE2188

Rajat Prasad 19BCE2533

School of Computer Science and Engineering



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

May 2021

DECLARATION

I hereby declare that the report entitled **“IPL STATISTICAL ANALYSIS”** submitted by me, for the CSE3020 DATA VISUALISATION (EPJ) to VIT is a record of bonafide work carried out by me under the supervision of Dr.S.VENGADESWARAN .

I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for any other courses in this institute or any other institute or university.

Place: Vellore

Date: 5 June, 2021

Naman

Arsh

Ishita

Rajat

Signature of the Candidate

1. Abstract

Sports play a very significant role in the development of the human persona. Getting involved in games like Cricket and other various sports help us to build character, discipline, confidence and physical fitness. Indian Premier League, IPL provides the most successful form of cricket as it gives opportunities to young and talented players to show case their talents on various pitch. Decision-makers are the utmost customers for all fundamentals in the sports analytics framework. Sports analytics has been a smash hit in shaping success for many players and teams in various sports. Sports analytics and data visualization can play a crucial role in selecting the best players for a team. This paper is about the player and match related analysis and the breadth of data visualization in supporting the decision makers for identifying inherent players for their teams.

2. Introduction to the Project

Indian Premier League, IPL provides the most successful form of cricket as it gives opportunities to young and talented players to showcase their talents on various pitches. Sports analytics and data visualization can play a crucial role in selecting the best players for a team and also help players increase their field performance. Our project is about the Toss Related analysis and the breadth of data visualization, supporting our goal in predicting the match-winner based on collective statistics from 2008 to 2019. The motive behind applying a visual analysis to it is to gain insight into what is likely to happen in the future.

2.1 Objective

We have applied visualization techniques to the datasets to provide deeper insight and pave way for recommendations to the player or team. The visual form of data is more easily understandable than numbers and text. And with the ease of obtaining and storing data, we have attempted to perform a Toss Related analytical and machine learning technique to engineer a model that can predict the match-winner based on past records from 2008 to 2019.

2.2 Problem Statement

With technology growing more and more advanced in the last few years, an in-depth acquisition of data has become relatively easy. With millions of people following the Indian Premier League (IPL), developing a model for predicting the outcome of its matches is a real-world problem. As a result, Machine Learning is becoming quite a trend in sports analytics because of the availability of living as well as historical data. Sports analytics is the process of collecting past matches data and analyzing them to extract the essential knowledge out of it, with the hope that it facilitates effective decision making.

SOLUTION

We have applied visualization techniques to the datasets to provide deeper insight and pave way for recommendations to the player or team. The visual form of data is more easily understandable than numbers

and text. And with the ease of obtaining and storing data, we have attempted to perform a Toss Related analytical and machine learning technique to engineer a model that can predict the match winner based on past records from 2008 to 2019.

2.3 Functional Requirements

Dataset

While dealing with data, Kaggle: Your Home for Data Science is the to-go platform.

We used the dataset <https://www.kaggle.com/vamsikrishna/exploratory-data-analysis-of-ipl-data> for visualizing the different parameters throughout the IPL season.

Which two different files:

1. matches.csv
2. deliveries.csv

Tools

R software, R dashboard, R Script to write the code, ggplot2 & readr, dplyr & gridExtra & treemap & RColorBrewer, tidyr as library

3. Data Abstraction

3.1 Visualizing the Datasets

- In the first phase, we filtered and cleaned the matches and delivery datasets. This was a major pre-processing done for the collected data as most of the entries were empty.
- The datasets containing the details of every match from 2008 to 2019 are represented in the form of charts and graphs to get a visual presentation of the data.
- The data is visualized to get a better and clear understanding about all the parameters of the season, the teams, all-rounders, batsmen and bowlers, average wins and toss results affecting the match winners.
- Different new features were introduced such as the number of Total Matches played by the Teams for all the 11 Seasons, Maximum Man of the Matches, Maximum Runs and Centuries scored by Batsmen, Maximum player of the Match Awards, Maximum count of Toss wins by Different Teams, Decision taken by each Team after Winning the Toss, etc.

3.2 Match Winner Prediction

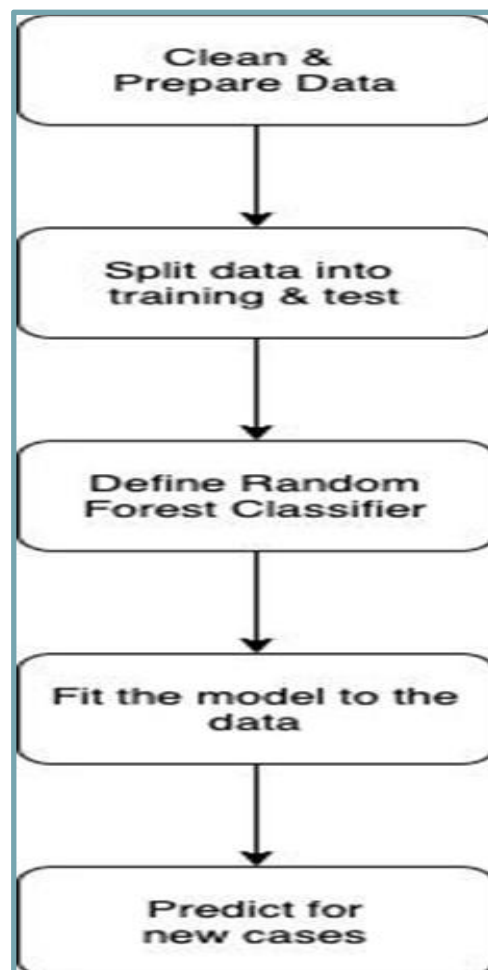
- We took points table from year 2008 to 2019 as an input, after dropping the redundant columns, we choose some custom parameters that are very crucial like 'toss_decision', 'field', 'bat', 'toss_win',

'team 1', 'team 2' etc and thus obtained a new points table.

- We then applied a Random Classifier and created a model to find out which team would win if a match was supposed to happen between the two teams.

4.Design of the proposed system

- From the visualization choose the most influencing factors (team points, run-rate, toss decision, and toss-win)
- Wrangle data and process into a new single data frame
- Split data into training and testing sets for training
- Define model (Random Forest Classifier here)
- Fit the model to the data
- Create a prediction function to reuse again and again.
- Finally, pass two team names, toss outcome and winner of toss to the prediction function, and the model will return the predicted winner.



5. Algorithm Design

- In the first phase, we filtered and cleaned the matches and deliveries datasets. This was a major pre-processing done for the collected data as most of the entries were empty. The datasets containing the details of every match from 2008 to 2019 are represented in the form of charts and graphs to get a visual presentation of the data.
- The data is visualized to get a better and clear understanding about all the parameters of the season, the teams, all-rounders, batsmen and bowlers, average wins and toss results affecting the match winners.
- Different features such as the number of Total Matches played by the Teams for all the 11 Seasons, Maximum Man of the Matches, Maximum Runs and Centuries scored by Batsmen, Maximum player of the Match Awards, Maximum count of Toss wins by Different Teams, Decision taken by each Team after Winning the Toss, etc can be accessed.

6. Task Abstraction

Tasks

1. Closeness of the matches when the team batting first wins
2. To find if home advantage is a real thing in IPL?
3. Top Batsmen
4. Batsmen with a top strike rate
5. Against which bowlers have the top run-getters performed? (run made by best batsmen against the bowler)
6. Best partnership of batsmen with other batsmen
7. Types of dismissal
8. Inning progression of the player

Actions

After the tasks are finalized, the next important step is to know how to fulfill a task, it can be done by planning actions. Actions are the user-defined goals. Our project is based on statistical analysis. We have to analyze-consume the existing data.

These are the following steps:-

❖ Analyze

We have consumed the existing data and analyzed it to create something new out of it for the better benefit of our users. In our project, we have tried to find out the best players, the best team from the attribute which is already given in the dataset. We have gotten outstanding results showing the best player and his run against different bowlers. Predicting the chance of winning and losing, or even give factual data about some myths like does winning the toss really helps.

❖ Design

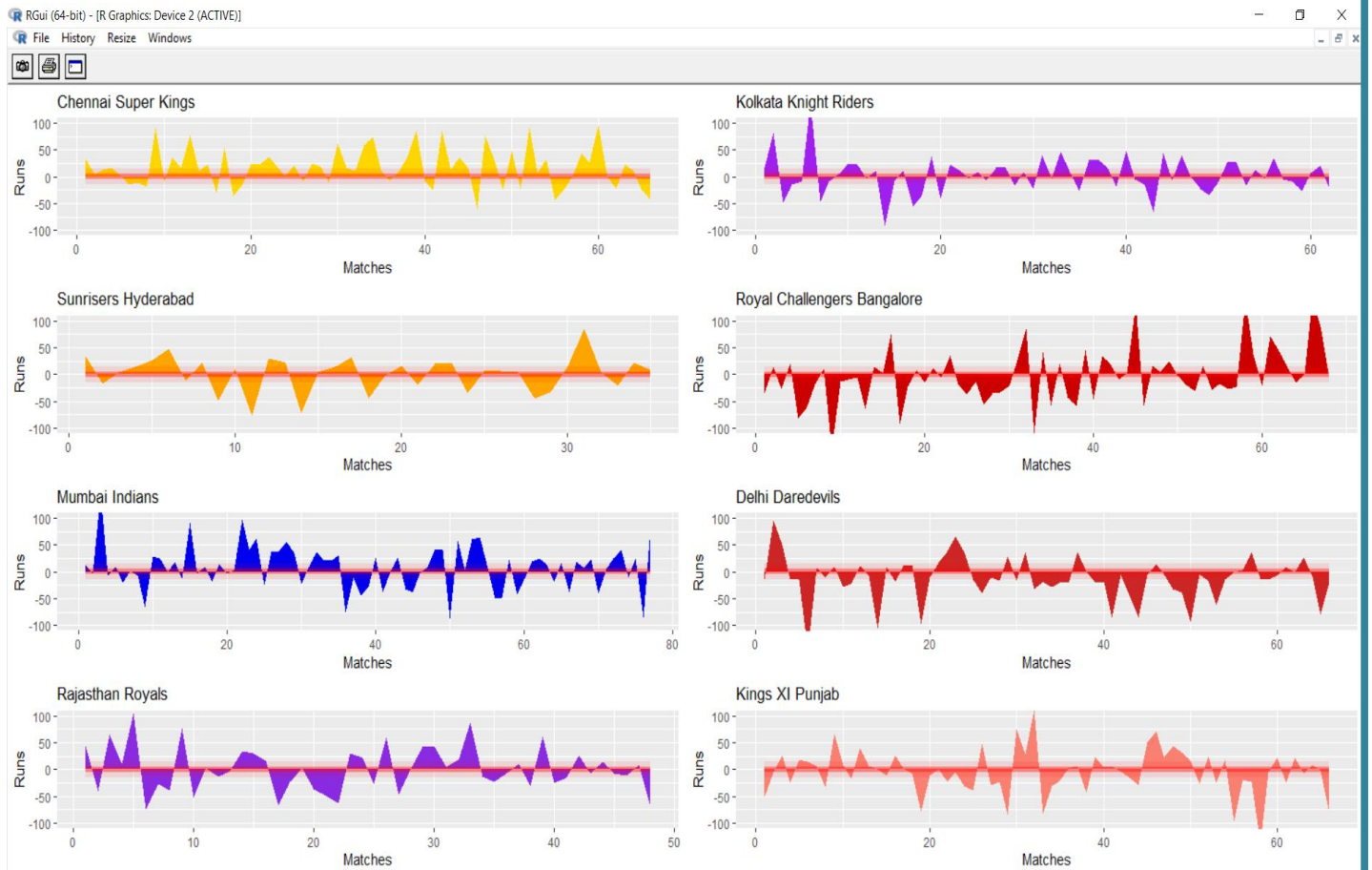
We have presented the data in user-friendly and understandable visualization, with appropriate marks and channels, and graphs that are most suitable so that it doesn't become hard to understand thus spoiling the whole concept of data visualization. Even if the users don't know the language used they won't find it difficult to understand and comprehend the data.

❖ Interesting feature

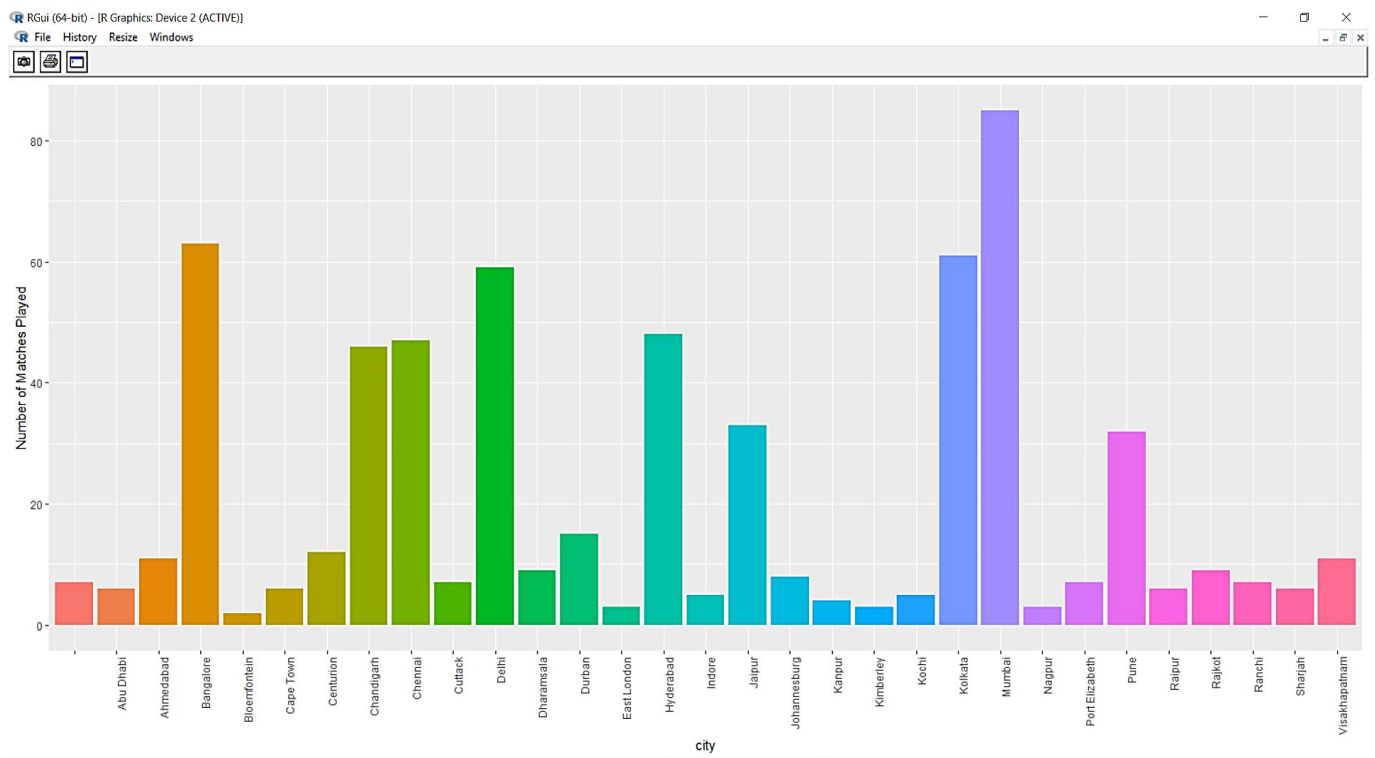
It is very important for a visualization to be attractive and enjoyable and looking at the project we already know we have many users. We have used different colours and palettes. Users will be interested in knowing who is best according to the statistics and even the type of dismissal of their favorite player.

7. Dashboard Implementation

The closeness of the matches when the team batting first wins



Number of matches played in different cities



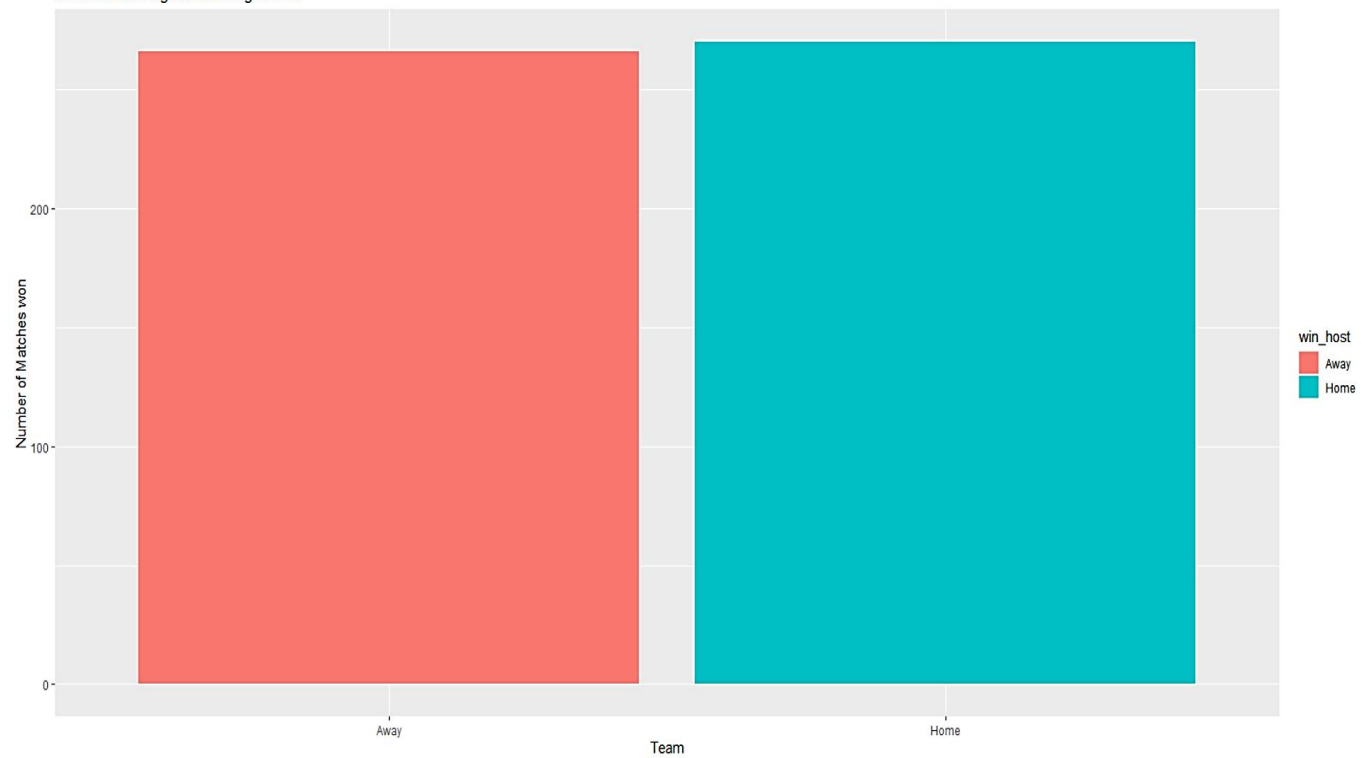
Is winning the toss really an advantage?



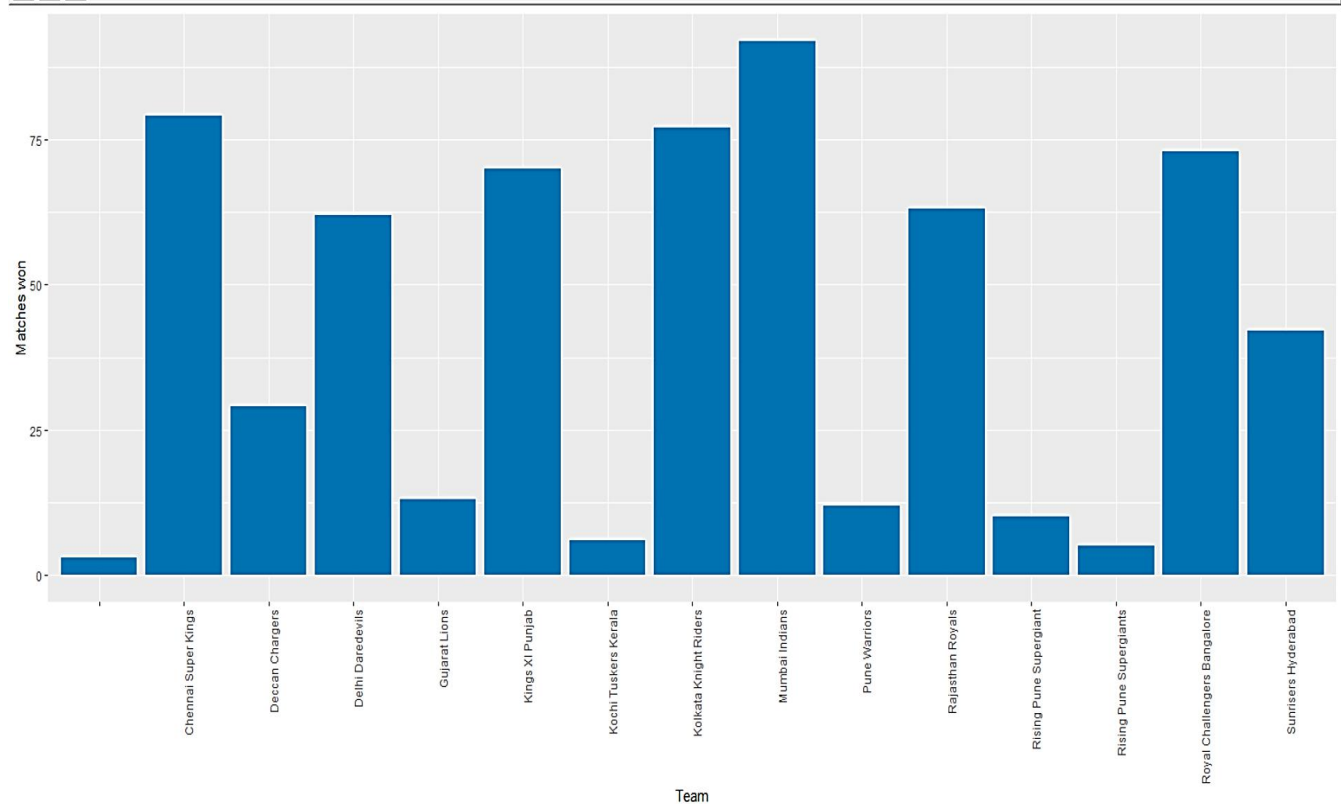
Is home advantage really a thing



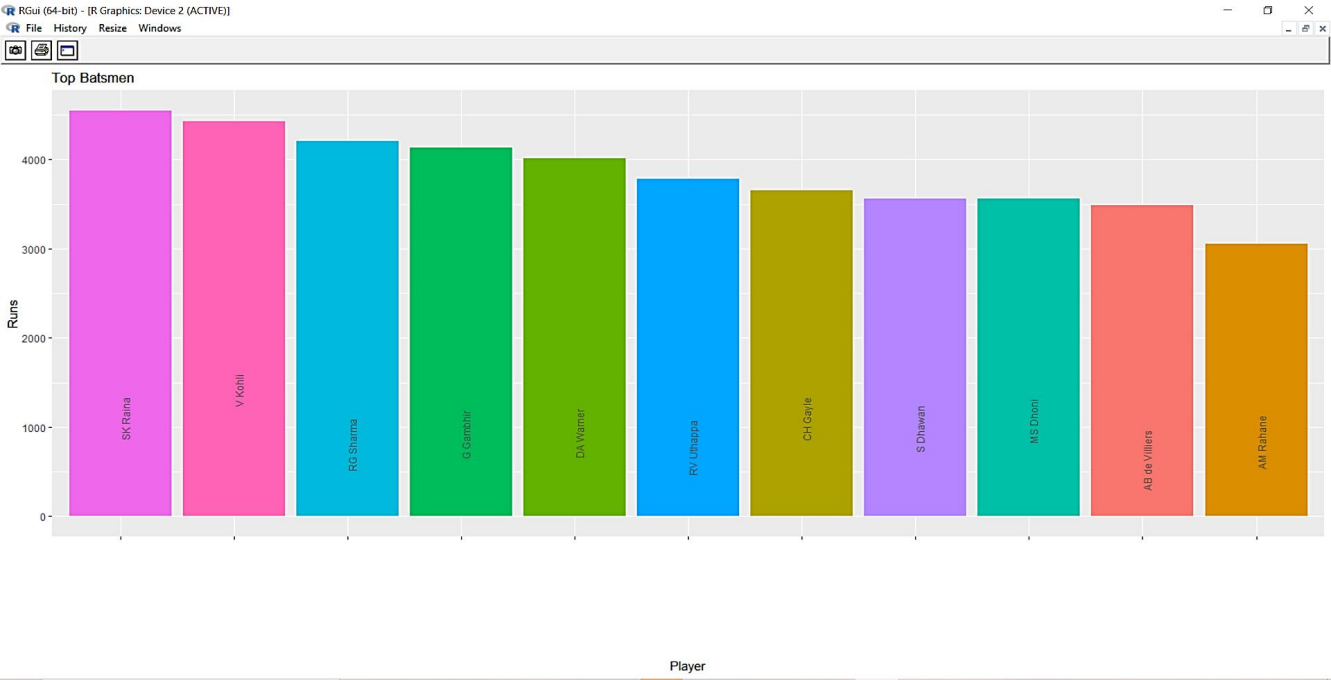
Is home advantage a real thing in IPL?



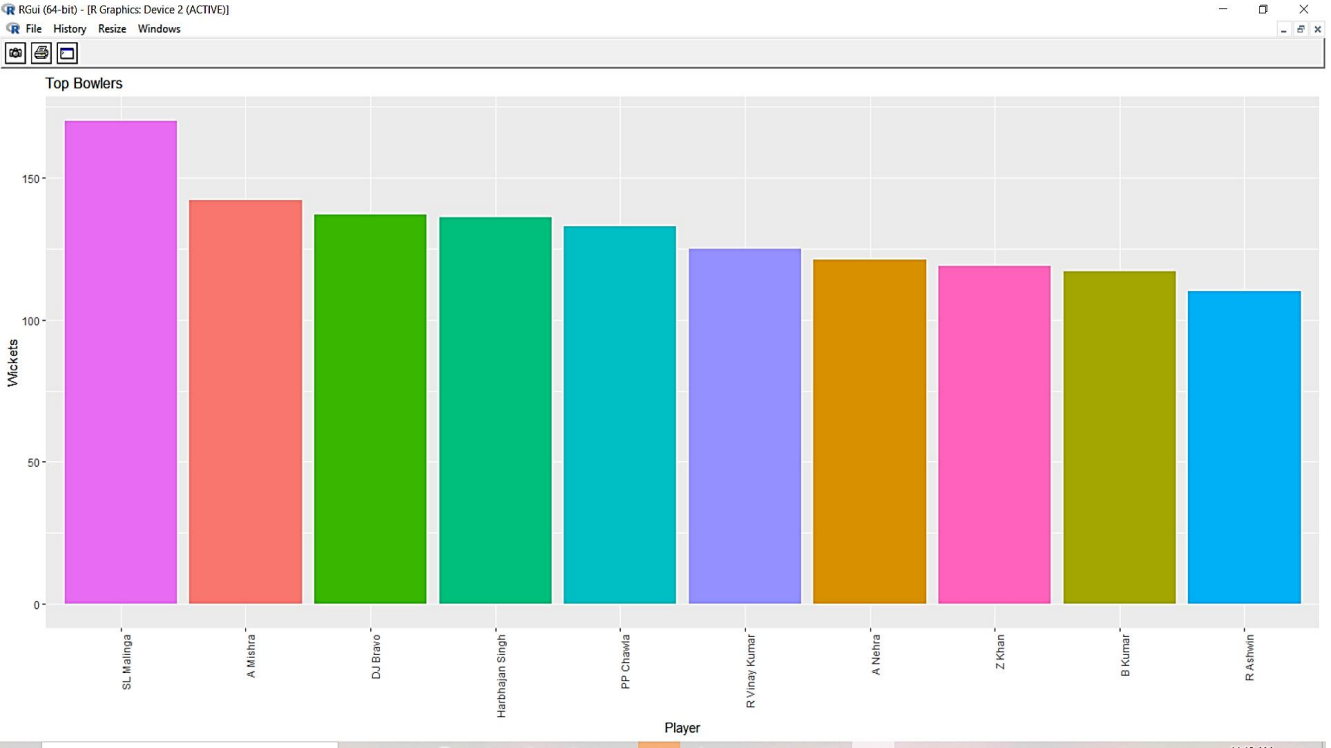
Number of Matches won by each team



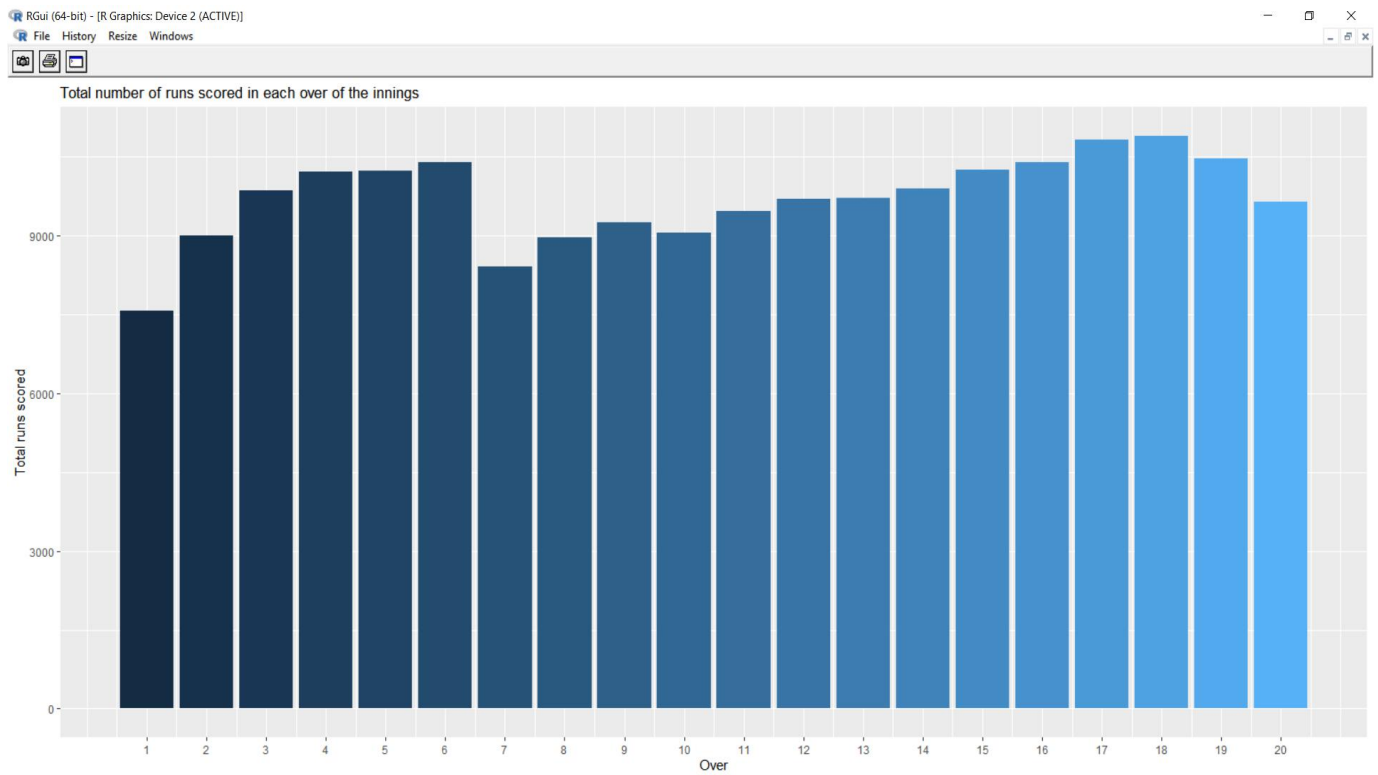
Top batsmen



Top bowlers



Total runs scored in each innings



Run against different teams and bowlers by certain batsmen:

First, we have found the top batsmen and used the data to find the above parameter but we can also find run by any batsmen against the bowlers

For example: DAVID WARNER

Runs by David Warner against different bowlers

P Kumar	Sandeep Sharma	SL Malinga	PP Chawla		AR Patel	J Yadav	
	HV Patel	AB Dinda	KA Pollard	SK Raina	R Ashwin	JA Morkel	
SP Narine	UT Yadav	NM Coulter-Nile	Z Khan	Anureet Singh	DW Steyn	RP Singh	
	KW Richardson	I Sharma	DA Warner	JJ Bumrah	VR Aaron	SK Trivedi	B Lee
Harbhajan Singh	PP Ojha	YK Pathan	Kuldeep Yadav	SR Watson	JH Kallis	JP Faulkner	KC Cariappa
MM Sharma	MJ McClenaghan	DJ Bravo	RA Jadeja	TG Southee	WD Parnell	CK Langeveldt	MC Henriques
	YS Chahal	MG Johnson	DS Kulkarni	A Mishra	AC Thomas	J Botha	P Awana
					Mohammed Shami	SW Tait	TP Sudhendra

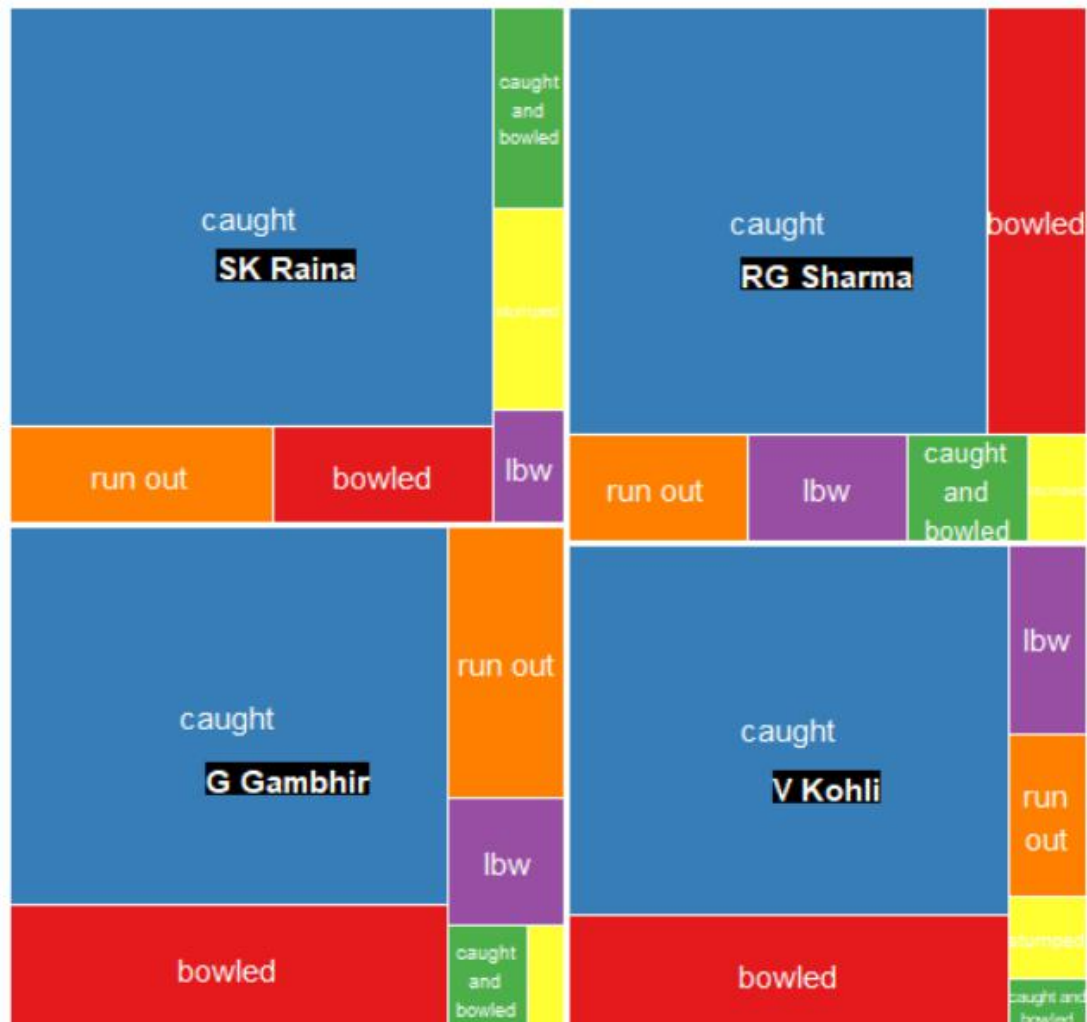
Next, we have: ROHIT SHARMA (HITMAN)

Runs by RG Sharma against different bowlers

PP Chawla	JH Kallis	Shakib Al Hasan		P Kumar		A Mishra		RA Jadeja	
	IK Pathan	DT Christian	MM Sharma		A Nehra		VR Aaron		AB Dinda
M Morkel	R Ashwin	A Kumble	Imran Tahir		DS Kulkarni		SK Trivedi		MC Henriques
		UT Yadav	S Nadeem	RG Sharma	DW Steyn	JD Unadkat	Joginder Sharma	P Negi	
AB Agarkar	SK Raina			M Kartik		MM Patel	B Kumar	P Awana	
R Bhatia	DJ Bravo	SK Warne	Yuvraj Singh	KV Sharma		L Balaji		Z Khan	DJ Hussey
		SR Watson	NLTC Perera	JP Faulkner		A Singh		I Sharma	
SP Narine	AD Russell	Gagandeep Singh	B Lee	MF Maharooof		CH Morris		RE van der Merwe	
								SB Jakati	

Types of dismissal

Type of Dismissals

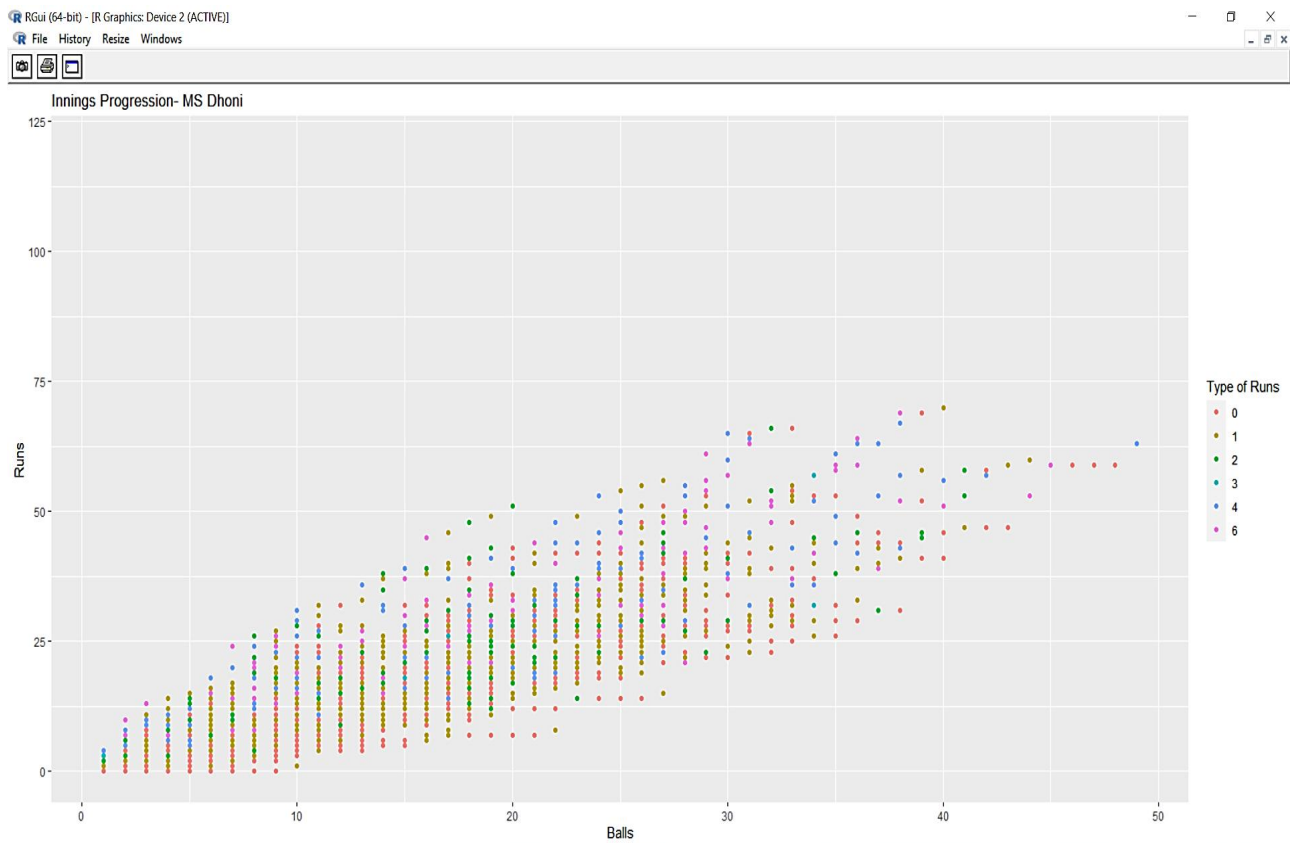


Best partnership

Runs with different players at the other end

RV Uthappa	MK Pandey		V Sehwag		SK Raina		RA Jadeja				
	G Gambhir McCullum		MS Bisla		MK Tiwary						
	S Dhawan		YK Pathan		CA Lynn		DA Warner				
JH Kallis					S Badrinath		DJ Bravo				
						MS Dhoni					
						ML Hayden		F du Plessis			
						MEK Hussey					
						JA Morkel		MK Tiwary			
						R Ashwin		SPD Smith			
S Dhawan		MC Henriques		V Sehwag							
		DA Warner		BT Chand		AJ Finch		MC Juneja			
		G Gambhir									
NV Ojha		KS Williamson		Y. Venugopal Rao		KL Rahul					
						DA Miller		HM Amla			
						GJ Maxwell					
						WP Saha		CA Pujara			

Inning progression of the player



8. Result Analysis

Results obtained through Visual Techniques applied to the given datasets

```
matches = pd.read_csv("matches.csv")
deliveries = pd.read_csv("deliveries.csv")
```

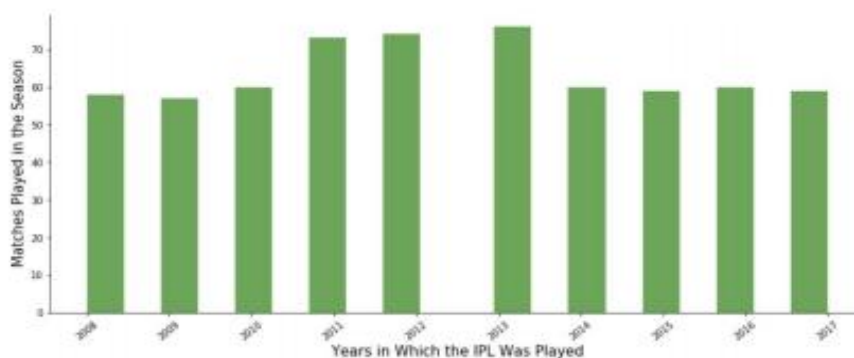


Fig 4.1.1 Histogram to show the number of matches played in each season of the IPL

Analysis: From Fig 4.1.1, we see that the year 2011 to 2013 having 9 IPL franchises have the maximum number of matches during those seasons, which brings the average matches played per season to 63.6.

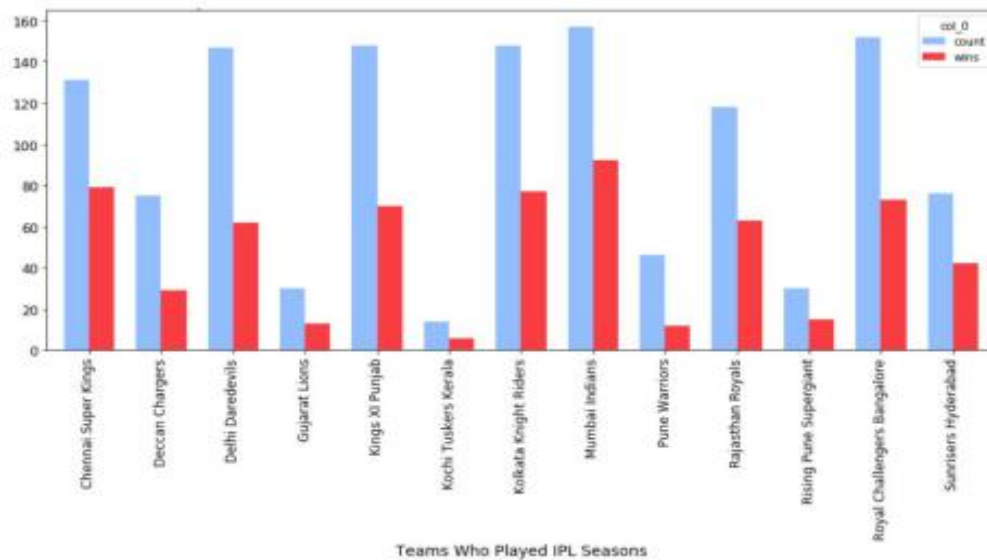


Fig 4.1.2 Grouped bar plot for each total matches and wins of each team in all seasons

Analysis: We found the total number of teams with the help of function `unique()`. We then calculated the number of matches played as team1 and team2 by the respective Teams using `crosstab()`. We then found the total matches won by the teams and grouped them accordingly in Fig 4.1.2

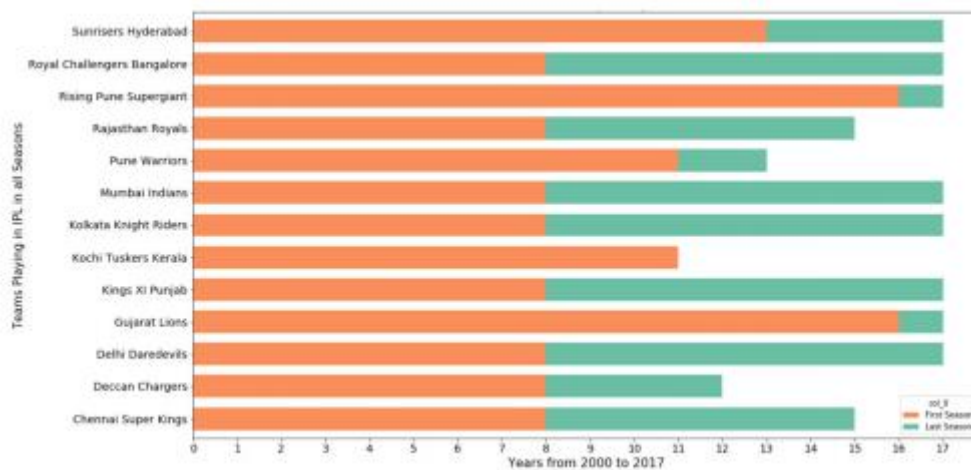


Fig 4.1.3 Horizontal Stacked Bar chart for seasons played by each IPL Team (Year v/s Time)

Analysis: We then sorted the teams according to the average matches played and won throughout the 11 seasons. And plotted a Stacked Bar chart for the same against the years played.

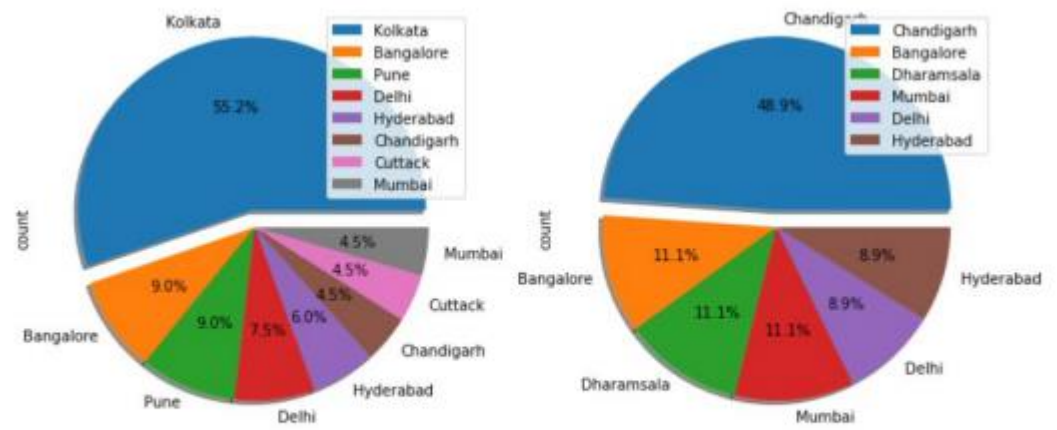


Fig 4.1.4 3D Pie Chart for Matches won by the teams I. Kolkata Knight Riders and II.Kings XI Punjab in different cities.

Analysis: We took a separate data array to store all the cities in which the matches were played. Removed any redundant values, sorted them accordingly to the number of matches played in the respective cities. We then plotted the cities in which each Team won in the form of a pie chart. Fig 4.1.4 shows the same for Teams KKR and KXIP.

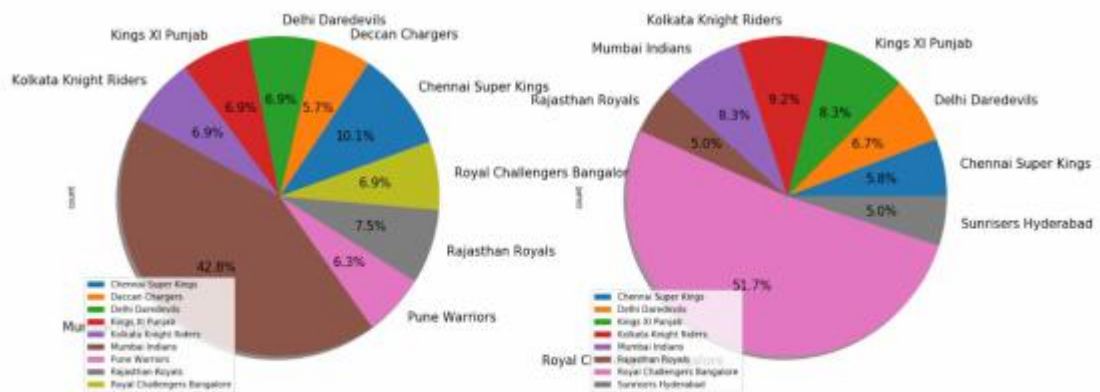


Fig 4.1.5 Cities Mumbai and Bangalore hosting matches of different Teams.

Analysis: We followed the same for 6 cities viz, Mumbai, Bangalore, Chennai, Delhi, Kolkata and Chandigarh. Wherein we calculated the total matches played by the Teams in the respective Cities. Fig 4.1.5 depicts the Pie Chart for Cities Mumbai and Bangalore.

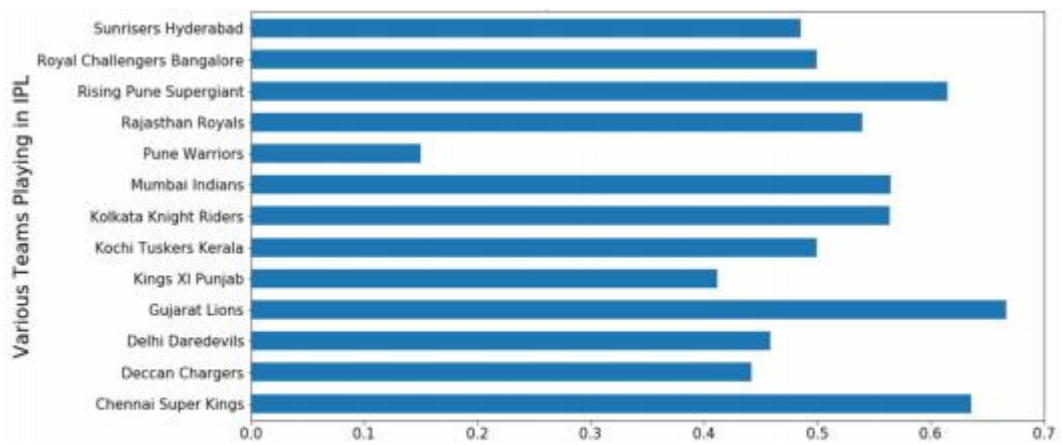


Fig 4.1.6 Horizontal Bar chart for Percentage win of the team when it won the Toss

Analysis: In Fig 4.1.6, we found out the total wins of each team against their toss wins and plotted the same above.

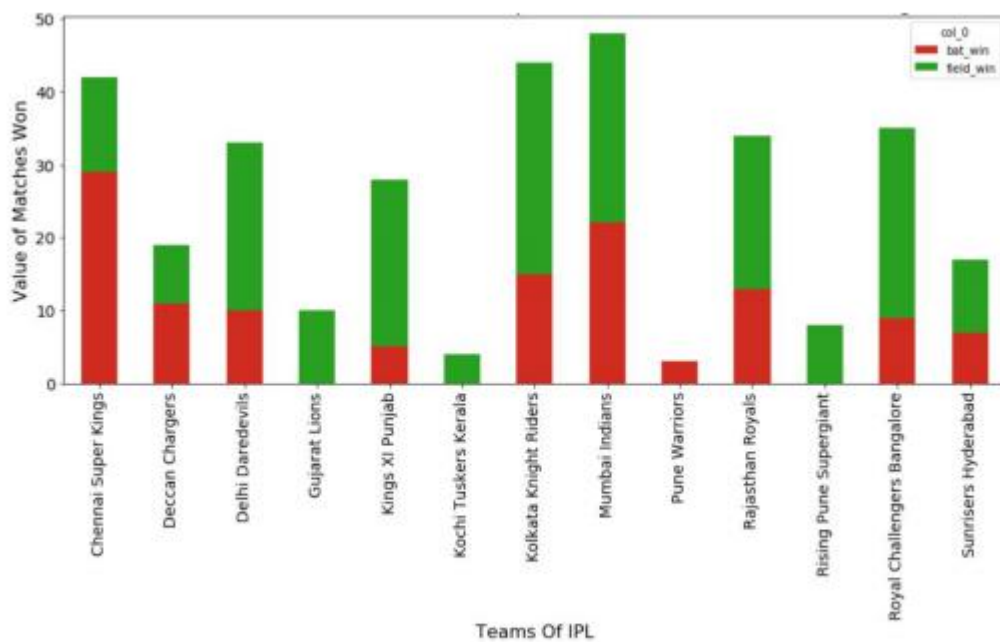


Fig 4.1.7 Stacked Bar Chart for the matches won wrt preference to field or bat after winning the Toss

Analysis: Here, we analyzed the Toss Winning team's decision to field or bat against the Outcome of the Match.

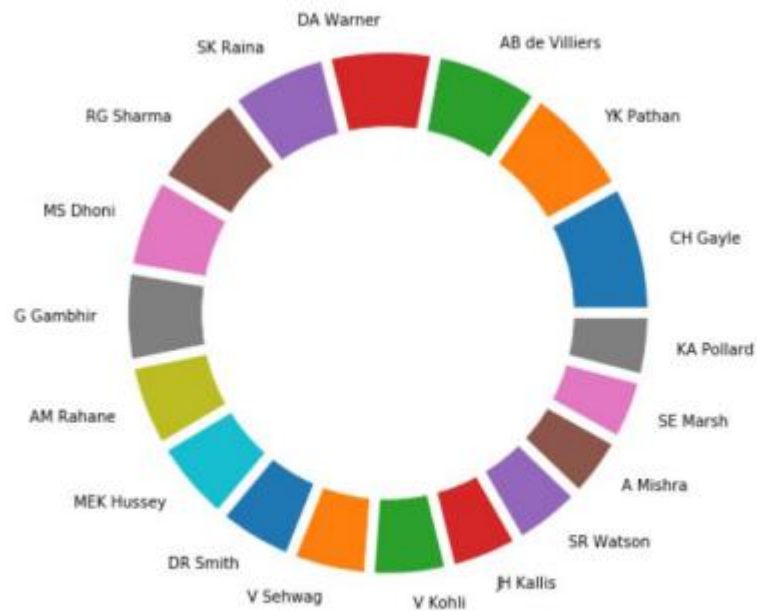


Fig 4.1.8 Doughnut Chart for Top 15 Cricketers who have been Man of the Matches

Analysis: In Fig 4.1.8, We found out the Top players from every Team based on the Title viz “Man of the Match” acquired by them.

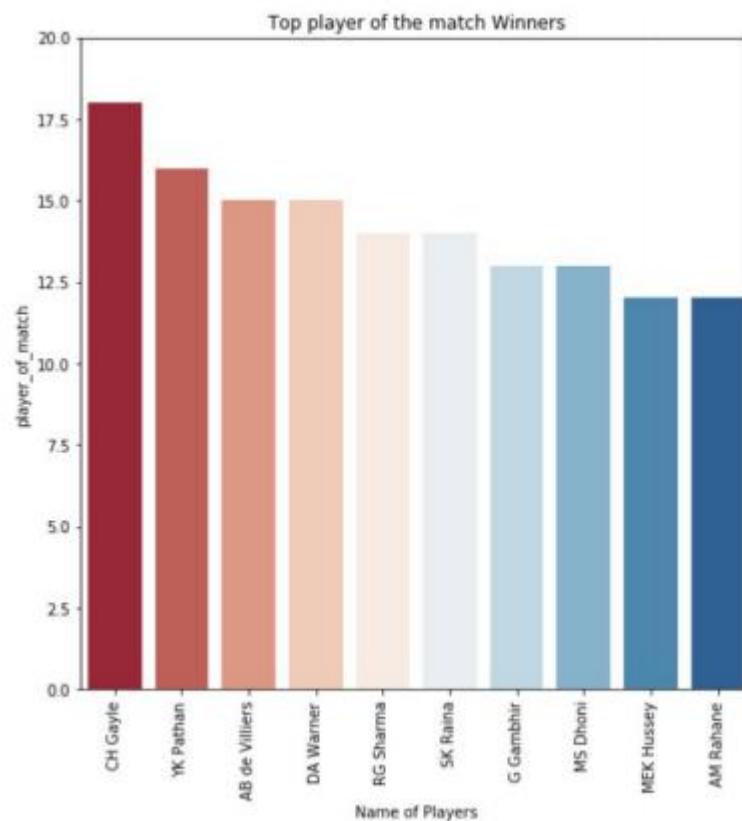


Fig 4.1.9 Top Players of the Match Winners

Analysis: Here, we have compared the Top players with the help of a Bar chart as shown in Fig 4.1.9. We can see that CH Gayle tops the list followed by YK Pathan.

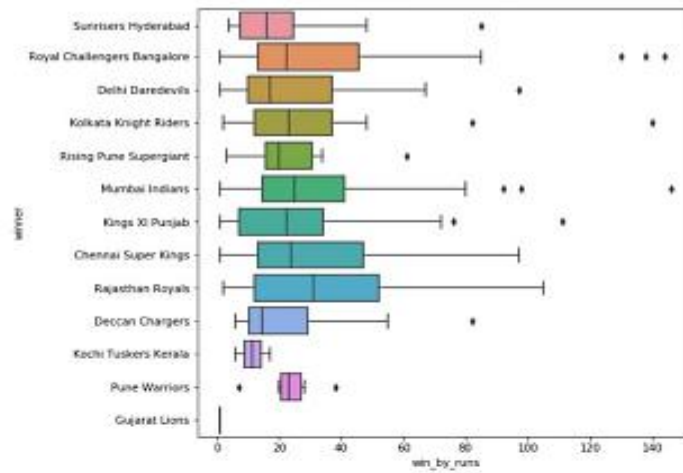


Fig 4.1.10 Box Plot showing Team Performance - Winning by Runs

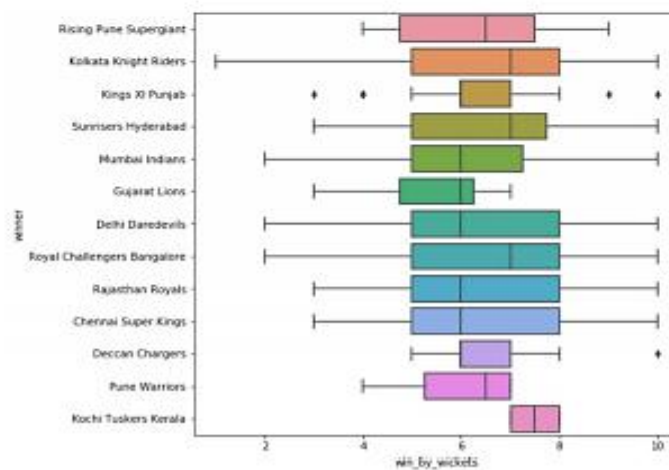


Fig 4.1.11 Box Plot showing Team Performance - Winning by Wickets

Analysis: In Fig 4.1.10 and 4.1.11, we have attempted to plot a performance chart with the help of their win rates as calculated earlier and the method through which they won.

9. Conclusion

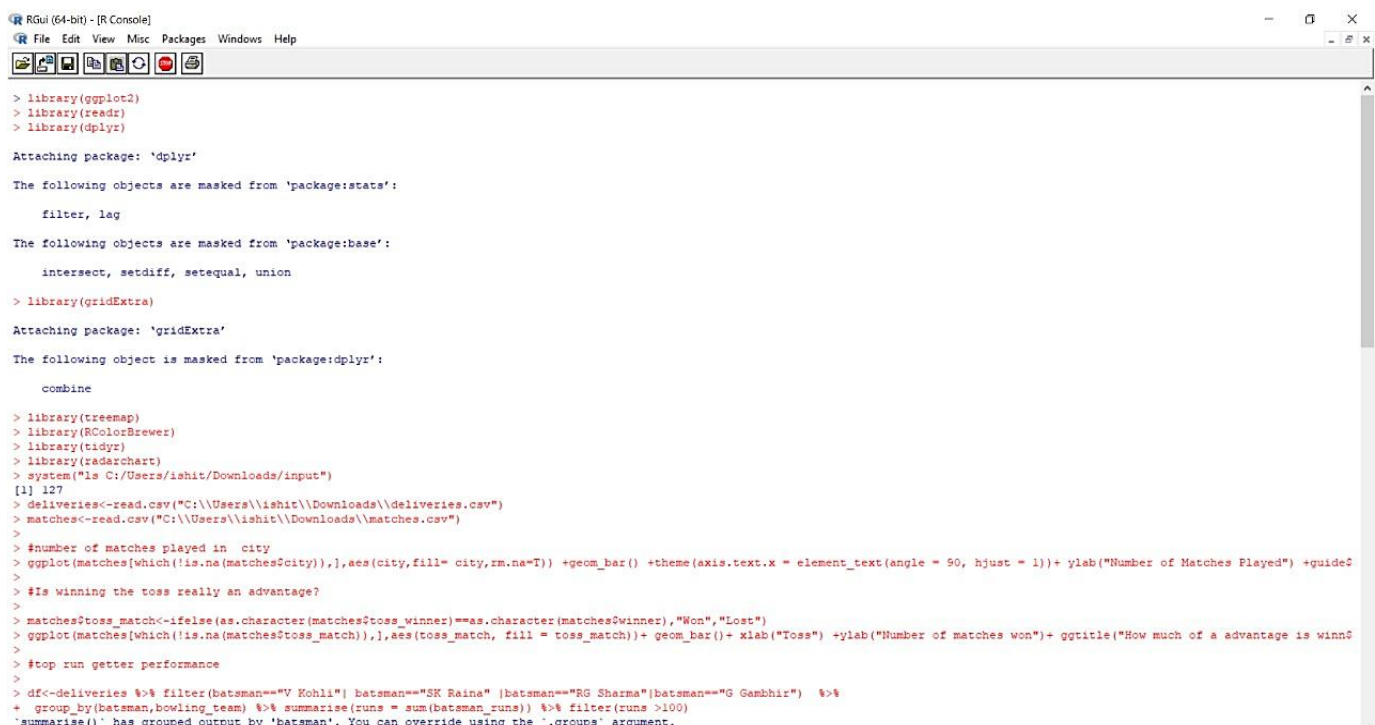
The dataset contained all the match data since the beginning of Indian Premier League till 2019. We used a trained model to predict the outcome of any IPL match, immediately after the toss. We have successfully been able to predict the winner of an IPL match using R Script ,ggplot2 ,readr ,&dplyr &gridExtra &treemap &RColorBrewer, tidyr as library. Even though our prediction might not always be accurate, it gives a basic idea about the strategies and methodologies applied into designing a solution as such.

9.1 Future Scope

- Since the dawn of the IPL in 2008, it has attracted viewers all around the globe. A high level of uncertainty and last moment nail biters has urged fans to watch the matches. Within a short period, IPL has become the highest revenue-generating league of cricket. Data Analytics has been a part of sports entertainment for a long time.
- In a cricket match, we might have seen the scoreline showing the probability of the team winning based on the current match situation. This is and will continue being Data Analytics in action.
- The predictions for a match have taken a step further in today's world. Thus, the need for experts to predict the outcome of games and performances of individual players has come to the fore.
- 'Fantasy cricket' is a new feature for the fans that have developed in recent years and has been engaging a lot of people since its inception. Websites and mobile applications like Dream11, iplfantasy.com and many more have created a user interface for such predictions and people have been earning money through it.

10. Appendix

Screenshots:



```
RGui (64-bit) - [R Console]
File Edit View Misc Packages Windows Help

> library(ggplot2)
> library(readr)
> library(dplyr)

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':
  filter, lag

The following objects are masked from 'package:base':
  intersect, setdiff, setequal, union

> library(gridExtra)

Attaching package: 'gridExtra'

The following object is masked from 'package:dplyr':
  combine

> library(treemap)
> library(RColorBrewer)
> library(tidyr)
> library(radarchart)
> system("ls C:/Users/ishit/Downloads/input")
[1] 127
> deliveries<-read.csv("C:\\Users\\ishit\\Downloads\\deliveries.csv")
> matches<-read.csv("C:\\Users\\ishit\\Downloads\\matches.csv")
>
> #number of matches played in city
> ggplot(matches[which(!is.na(matches$city)),],aes(city,fill= city,lm.na=T)) +geom_bar() +theme(axis.text.x = element_text(angle = 90, hjust = 1))+ ylab("Number of Matches Played") +guide$
>
> #Is winning the toss really an advantage?
>
> matches$toss_match<-ifelse(as.character(matches$toss_winner)==as.character(matches$winner),"Won","Lost")
> ggplot(matches[which(!is.na(matches$toss_match)),],aes(toss_match, fill = toss_match))+ geom_bar() + xlab("Toss") -ylab("Number of matches won")+ ggtitle("How much of a advantage is winn$
>
> #top run getter performance
>
> df<-deliveries %>% filter(batsman=="V Kohli"| batsman=="SK Raina" |batsman=="RG Sharma"|batsman=="G Gambhir") %>%
+ group_by(batsman,bowling_team) %>% summarise(runs = sum(batsman_runs)) %>% filter(runs >100)
'summarise()' has grouped output by 'batsman'. You can override using the '.groups' argument.
```




```

+ top_n(n= 10, wt = total)
+ p<- ggplot(aes(x = reorder(dismissal_kind, -total), y= total), data = x)+
+ geom_bar(aes(fill= dismissal_kind), stat = "identity")+
+ labs(list(title = type, x = "Dismissal Kind", y = "Total Wickets"))+
+ theme(axis.text.x=element_text(angle=75, hjust=1), plot.title = element_text(size = 8, face = "bold"),text = element_text(size=8))
+ return(p)
+ }
>
> is home advantage really a thing
Error: unexpected symbol in "is home"
>
> Data<-matches[matches$season!="2009",]
> Data$date<- as.Date(Data$date)
> Data<-Data[Data$date < as.Date("2014-04-16") | Data$date > as.Date("2014-04-30"),]
> Data$home_team[Data$city=="Bangalore"]<- "Royal Challengers Bangalore"
> Data$home_team[Data$city=="Chennai"]<- "Chennai Super Kings"
> Data$home_team[Data$city=="Delhi"]<- "Delhi Daredevils"
> Data$home_team[Data$city=="Chandigarh"]<- "Kings XI Punjab"
> Data$home_team[Data$city=="Jaipur"]<- "Rajasthan Royals"
> Data$home_team[Data$city=="Mumbai"]<- "Mumbai Indians"
> Data$home_team[Data$city=="Kolkata"]<- "Kolkata Knight Riders"
> Data$home_team[Data$city=="Kochi"]<- "Kochi Tuskers Kerala"
> Data$home_team[Data$city=="Hyderabad" & Data$season <=2012]<- "Deccan Chargers"
> Data$home_team[Data$city=="Hyderabad" & Data$season >2012]<- "Sunrisers Hyderabad"
> Data$home_team[Data$city=="Ahmedabad"]<- "Rajasthan Royals"
> Data$home_team[Data$city=="Dharamsala"]<- "Kings XI Punjab"
> Data$home_team[Data$city=="Visakhapatnam" & Data$season== 2015]<- "Sunrisers Hyderabad"
> Data$home_team[Data$city=="Ranchi" & Data$season== 2013]<- "Kolkata Knight Riders"
> Data$home_team[Data$city=="Ranchi" & Data$season > 2013]<- "Chennai Super Kings"
> Data$home_team[Data$city=="Rajkot" ]<- "Gujarat Lions"
> Data$home_team[Data$city=="Kanpur" ]<- "Gujarat Lions"
> Data$home_team[Data$city=="Raipur" ]<- "Delhi Daredevils"
> Data$home_team[Data$city=="Nagpur" ]<- "Deccan Chargers"
> Data$home_team[Data$city=="Indore" ]<- "Kochi Tuskers Kerala"
> Data$home_team[Data$city=="Pune" & Data$season!= 2016]<- "Pune Warriors"
> Data$home_team[Data$city=="Pune" & Data$season== 2016]<- "Rising Pune Supergiants"
> Data<-Data[ which(!is.na(Data$home_team)),]
> Data$win_host <- ifelse(as.character(Data$winner)==as.character(Data$home_team),"Home","Away")
>
> ggplot(Data[which(!is.na(Data$win_host)),],aes(win_host,fill= win_host))+geom_bar()+
+ ggtitle("Is home advantage a real thing in IPL?")+
+ xlab("Team")+
+ ylab("Number of Matches won")+labs(aesthetic="Winner")
>

```



```

+ group_by(batsman,bowling_team) %>% summarise(runs = sum(batsman_runs)) %>% filter(runs >100)
'summarise()' has grouped output by 'batsman'. You can override using the '.groups' argument.
> treemap(df, #Your data frame object
+         index=c("batsman","bowling_team"), #A list of your categorical variables
+         vSize = "runs",
+         vColor = "bowling_team",
+         type="categorical", #Type sets the organization and color scheme of your treemap
+         palette = brewer.pal(12,"Set3"), #Select your color palette from the RColorBrewer presets or make your own.
+         fontsize.title = 15,
+         fontfamily.title = "serif",
+         fontfamily.labels = "symbol",
+         title = "Runs against diff teams",
+         aspRatio = 1,
+         border.col="#FFFFFF",bg.labels = "#FFFFFF" ,fontcolor.labels= "black",fontsize.legend = 0
+ )
There were 50 or more warnings (use warnings() to see the first 50)
>
>
> bowl<-function(type){
+ x<-df1 %>%
+ filter(bowler==type & dismissal_kind %in% c("caught","bowled","lbw","stumped","caught and bowled", "hit wicket")) %>%
+ group_by(dismissal_kind) %>%
+ summarise(total = n()) %>%
+ arrange(desc(total)) %>%
+ top_n(n= 10, wt = total)
+ p<- ggplot(aes(x = reorder(dismissal_kind, -total), y= total), data = x)+
+ geom_bar(aes(fill= dismissal_kind), stat = "identity")+
+ labs(list(title = type, x = "Dismissal Kind", y = "Total Wickets"))+
+ theme(axis.text.x=element_text(angle=75, hjust=1), plot.title = element_text(size = 8, face = "bold"),text = element_text(size=8))
+ return(p)
+ }
>
> is home advantage really a thing
Error: unexpected symbol in "is home"
>
> Data<-matches[matches$season!="2009",]
> Data$date<- as.Date(Data$date)
> Data<-Data[Data$date < as.Date("2014-04-16") | Data$date > as.Date("2014-04-30"),]
> Data$home_team[Data$city=="Bangalore"]<- "Royal Challengers Bangalore"
> Data$home_team[Data$city=="Chennai"]<- "Chennai Super Kings"
> Data$home_team[Data$city=="Delhi"]<- "Delhi Daredevils"
> Data$home_team[Data$city=="Chandigarh"]<- "Kings XI Punjab"
> Data$home_team[Data$city=="Jaipur"]<- "Rajasthan Royals"
> Data$home_team[Data$city=="Mumbai"]<- "Mumbai Indians"

```