# Process Book: Peek into Indian Premier League

https://github.com/bkommineni/DataVisualization-FinalProject

Page to host the project: https://bkommineni.github.io/DataVisualization-FinalProject/

---

By: Anjani Bajaj (aabajaj2@dons.usfca.edu)

Bhargavi Kommineni (bkommineni@dons.usfca.edu)

## 1. Overview and Motivation

Cricket is an extremely popular sport in India, however it is under represented in the world. We come from India, where we grew up watching cricket. The Indian Premier League (IPL) is a professional Twenty20 cricket league in India contested during April and May of every year by teams representing Indian cities and some states. IPL started in 2008 and includes cricket players from all over the world. It is also the most-attended cricket league in the world and in 2014 ranked sixth by average attendance among all sports leagues.

Hence, we decided to explore and visualize IPL data from kaggle in this project. We plan to visualize the data effectively and hope to find some great insights throughout the process.

**Cricket Basics**

- **Bat** : the wooden implement with which the batsman attempts to strike the ball
- **Ball** : the round object which the batsman attempts to strike with the bat. Also a delivery.
- **Batting** : the act and skill of defending one's wicket and scoring runs
- **Bowling** : the act of delivering the cricket ball to the batsman.
- **Bowling Average** : A bowler's bowling average is defined as the total number of runs conceded by the bowler divided by the number of wickets taken by the bowler.
- **Batting Average** : A batsman's batting average is defined as the total number of runs scored by the batsman divided by the number of times he has been dismissed
- **Catch** : To dismiss a batsman by a fielder catching the ball after the batsman has hit it with his bat but before it hits the ground
- **Caught** :Its a method of dismissing a batsman in the sport of cricket. Being caught out is the most common method of dismissal at higher levels of competition. A batsman is out caught if a fielder catches the ball fully within the field of play without it bouncing once the ball has touched the striker's bat, glove or only the leg of the batsman.
- **Caught and bowled** : When a player is dismissed by a catch taken by the bowler.
- **Caught behind** : A catch by the wicket-keeper.
- **Four** : A shot that reaches the boundary after touching the ground, which scores four runs to the batting side.
- **Six** : a shot which passes over or touches the boundary without having bounced or rolled, so called because it scores six runs to the batting side
- **Hit wicket** : A batsman getting out by dislodging the bails of the wicket behind him either with his bat or body as he tries to play the ball or set off for a run.

- **Inning** : One player's or one team's turn to bat (or bowl). Unlike in baseball, the cricket term "innings" is both singular and plural.
- **LBW** : Leg Before Wicket, It means when there is leg in front of wicket then the batsman is given out, but bowler have to make an appeal to umpire.
- **Leg bye** : Extras taken after a delivery hits any part of the body of the batsman other than the bat or the gloved hand that holds the bat. If the batsman makes no attempt to play the ball with the bat or evade the ball that hits him, leg byes may not be scored
- **Match referee** : an official whose role is to ensure that the spirit of the game is upheld.
- **No ball** : An illegal delivery; the batting side is awarded one extra, the bowler must deliver another ball in the over, and the batsman cannot be dismissed by the bowler on a no-ball. Most usually a front-foot no ball, in which the bowler oversteps the popping crease; other reasons include bowling a full toss above waist height (see beamer), throwing, having more than two fielders (excluding the wicketkeeper) behind square on the leg side, or breaking the return crease in the delivery stride
- **Non-striker** : The batsman standing at the bowling end
- **Obstructing the field** : It is one of the nine methods of dismissing a batsman in the sport of cricket. It dictates that either batsman can be given out if he wilfully attempts to obstruct or distract the fielding side by word or action.
- **Over** : The delivery of six consecutive legal balls by one bowler.
- **Retired out** : For a batsman to voluntarily leave the field during his innings, usually because of injury. A player who retired through injury/illness may return in the same innings at the fall of a wicket, and continue where he left off. A player who is uninjured may return only with the opposing captain's consent
- **Run out** : Dismissal by a member of the fielding side breaking the wicket while the batsman is outside his/her crease in the process of making a run.
- **Stump**
    - one of the three vertical posts making up the wicket ("off stump", "middle stump" and "leg stump");[2]
    - a way of dismissing a batsman in which the wicketkeeper breaks the batsman's wicket with the ball when the batsman is outside his crease but has not attempted a run; or
    - In a match lasting more than one day, "stumps" refers to the end of a day's play when the match is not complete (e.g. a progress score after the first day may be described as the score "at stumps on Day 1"). See also draw stumps

- **Toss** : In the sport of cricket, a coin is tossed to determine which team bats first. This is known as the toss.
- **Umpire** : one of the two (or three) enforcers of the laws[46] and adjudicators of play.
- **Wicket** :
    - a set of stumps and bails;
    - the pitch; or
    - the dismissal of a batsman.
- **Wide** : A delivery that passes illegally wide of the wicket, scoring an extra for the batting side. A wide does not count as one of the six valid deliveries that must be made in each over – an extra ball must be bowled for each wide
- **Duckworth-Lewis(DL)** : A mathematically based rule that derives a target score for the side batting second in a rain-affected one-day match
- **Extra** : A run not attributed to any batsman;
    - there are five types:
        - byes,
        - leg byes,
        - penalties,
        - wides and
        - no-balls.
    - The first three types are called 'fielding' extras (i.e. the fielders are determined to be at fault for their being conceded)
    - The last two are called 'bowling' extras (the bowler being considered to be at fault for their being conceded) which are included in the runs conceded by the bowler.

### 1.1 Match Format

Twenty20 match format is a form of limited overs cricket in that it involves two teams, each with a single innings, the key feature being that each team bats for a maximum of 20 overs. In terms of visual format, the batting team members do not arrive from and depart to traditional dressing rooms, but come and go from a bench (typically a row of chairs) visible in the playing arena, analogous to association football technical area or a baseball dugout. [1]

## 2. Related Work

a. Project on IPL data visualization using python packages:
https://www.kaggle.com/sharddha/data-visualization-of-indian-premier-league

b. Line Trends for Players example:
https://www.hindustantimes.com/interactives/one-day-cricket-batting/

c. Example Cricket Chord Diagram:
https://codepen.io/veereshai/full/pmvwC

d. Example on cricket visualizations:
https://www.behance.net/gallery/12307919/Visualizing-Cricket-Ind-vs-Pak-matches

e. Example cricket visualizations:
https://knoema.com/insights/cricket

f. We were also inspired by geo visualizations and decided to use GeoMap of India for the same. Here is the example which we looked at:
https://bl.ocks.org/JohnCoogan/1531818

## 3. Questions we are trying to answer

The goal of this project is to efficiently visualize the cricket data i.e. data about performance of the players and IPL matches for over 10 years (2008-2017) and identify if there exists some trends.

After the successful completion of this project we hope to answer the following questions:

- Where are the teams playing the IPL geographically located?

Team Level Analysis:

- How can we compare the performance of different teams over the years?
    - What can be the logistics for performance of the teams-
        - Number of matches won
        - Number of Dismissals
        - Number of runs scored etc.

Player Level Analysis:

- How can we analyze player performance across different teams and across different years?
- Can we find out correlations of player performance with team or their experience as a player in IPL?
- How can we compare the batting average, wickets taken and other attributes for different players over the years?

We kept refining the questions over the course of the project. We started with more general questions and narrowed them down.

## 4. Data

Source of the data is Kaggle - https://www.kaggle.com/manasgarg/ipl
The dataset contains 2 files: deliveries.csv and matches.csv.

- matches.csv contains details related to the match such as location, contesting teams, umpires, results, etc.
- deliveries.csv is the ball-by-ball data of all the IPL matches including data of the batting team, batsman, bowler, non-striker, runs scored, etc.

## 5. Exploratory Data Analysis

We first opened the data using Excel and tried to look at the values of different columns. We also used Tableau to initially visualize the small amount of data.

The first challenge we faced while visualizing the data in Tableau was that the data was not grouped according to the teams for each year. Hence, the task was to generate the data of the players for each team for all the years. As we did not have all the details related to players grouped according to a team. We tried to derive it and use them to generate Bar Charts and GeoMap visualizations. We wrote a python script ExtractTeams.py from which teams_by_year.json was generated so that we can group the data teamwise.

To implement player level analysis Line Charts, we needed year for the visualizations. The first idea: addYearToDeliveries.py was written to modify deliveries.csv to make it deliveries_with_year.csv. This idea was dropped as we figured the logic to nest two csv files and hence, we did not use deliveries_with_year.csv for any of the visualizations

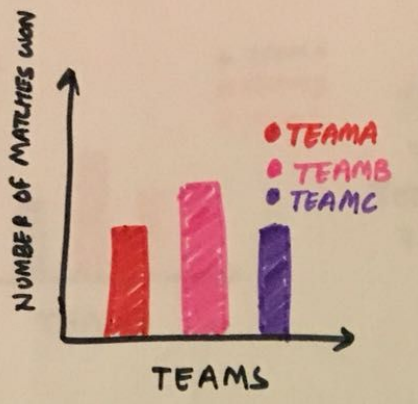
Chord Diagram Data Processing: ExtractTotalMatchesPlayedByTeam.py and Extract2DMatrixForChords.py were written to generate total_stats.json and matrix_wins.json. These files were required to implement chord diagram visualizations.

Scatter Plot Data Processing: We wrote ExtractPlayerDetails.py to generate player_details.json which has all the details (to be plotted) of a player for each year, grouping the necessary data for scatter plot of individual players.

## 6. Design Evolution

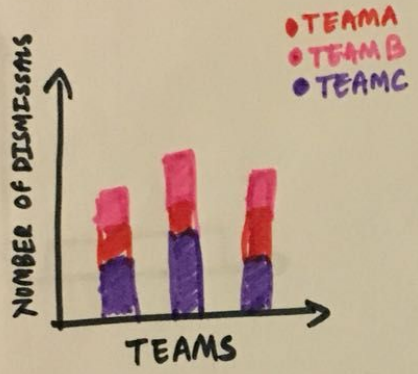
We started out with D3 visualizations in mind. Gradually as the project progressed, we thought of adding everything on one single page making it an easy to interact with dashboard for IPL data. All the visualizations are implemented in d3.js with interactions and embedded on this page https://bkommineni.github.io/DataVisualization-FinalProject/
Step 1. Using pen and paper trying to visualize the kind of visualizations we want:

# SHEET 1 : Brainstorm & Designs



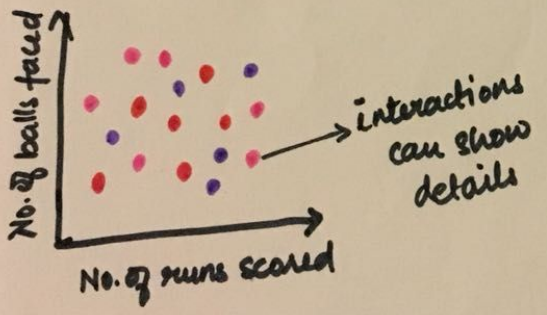

YEAR ← FILTER

BARCHART for
Year v/s teams
Number of matches won

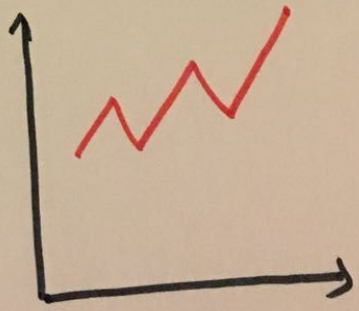



STACKED BAR CHART

Number of dismissals
v/s
teams




Some trends ?

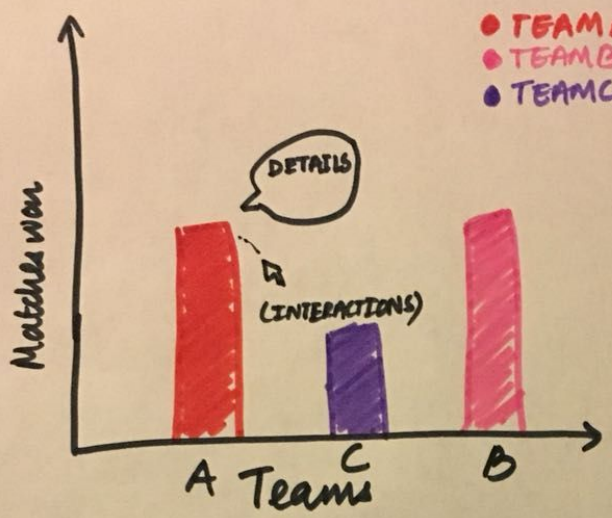


SHEET 2 :

TITLE: Match comparision

filter by year

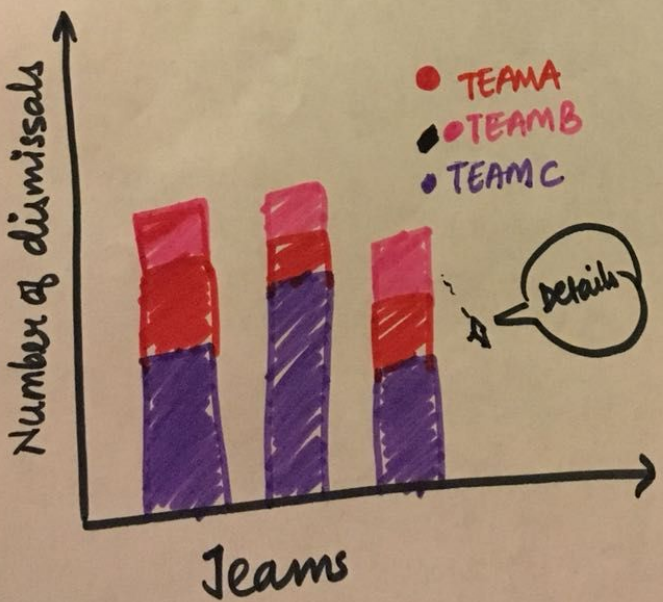
(A different bar chart for each ~~after~~ year!)



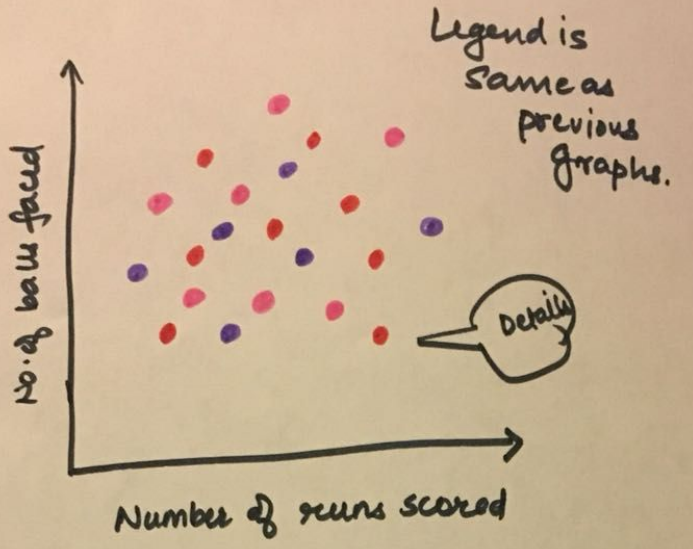

TITLE: Dismissal comparision

Filter by year.

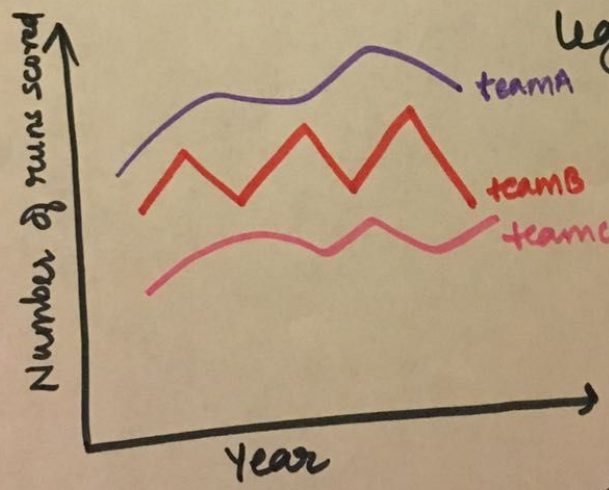
(A different stacked chart for each year)



SHEET 3




TITLE: Scatter Plot for performance

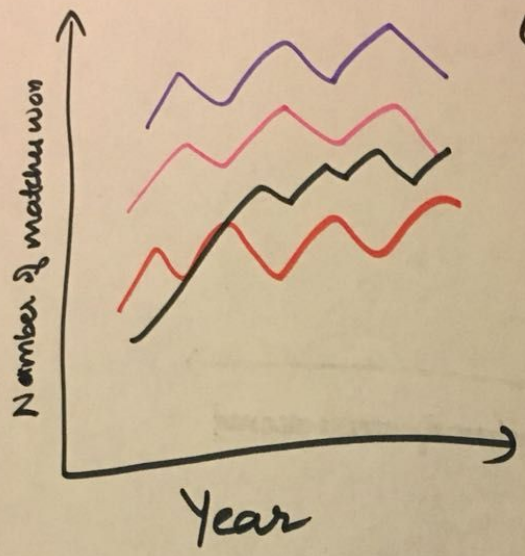



Legend is same

(Highlight selected and Grayout otherwise selection)

TITLE: Player performance over the years

Year wise trends for all teams




Legend is the same

(Filter can be team too)

(A drop down maybe?)

Detail:

Data processing: Python

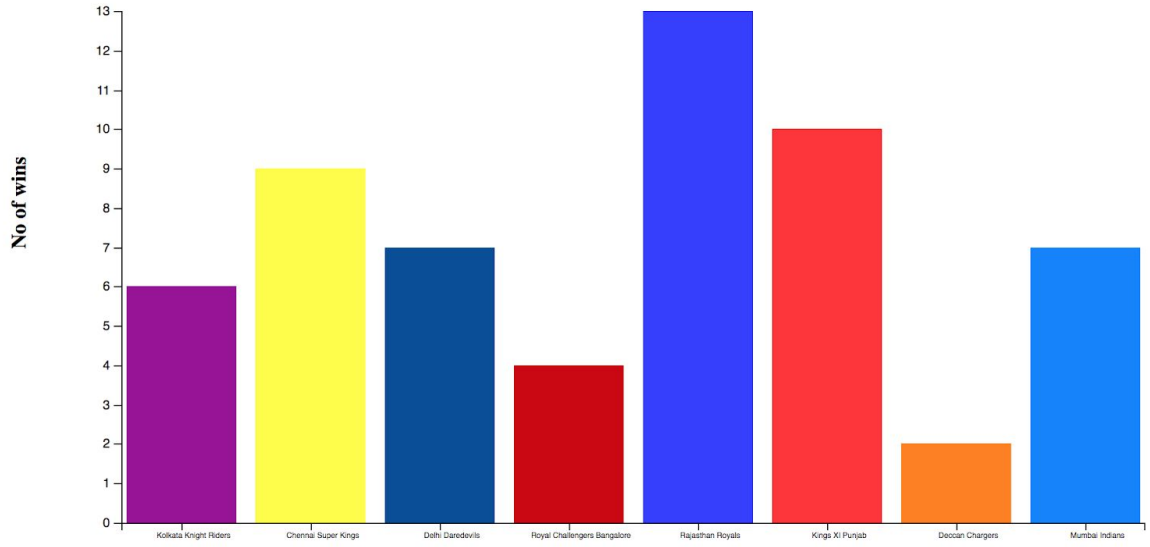
Visualizations: D3.js

Step 2. To get clearer idea of what to plot in the D3 visualizations, we tried to visualize data in Tableau. Below are the pictures explaining the process.

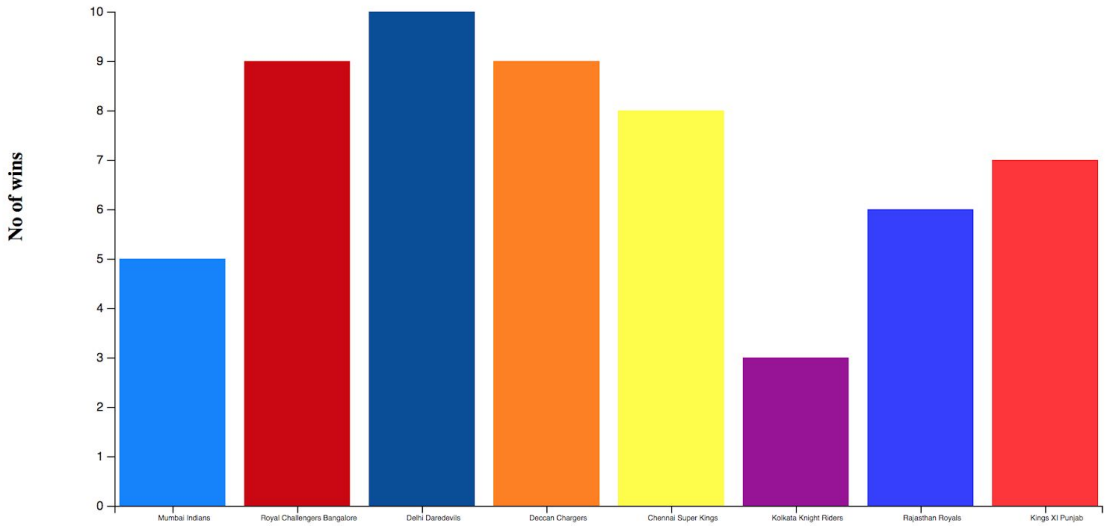
Step 3. Plot static D3 graphs without dropdown and interactions.

- Filter match ids of each year and store year with respective match ids from matches.csv
- For each year based on match ids in deliveries.csv added players under "batsman" and "non_striker" to batting team and players under "bowler" to bowling team
- Verified the output of above process with the actual data available in IPLT20 website

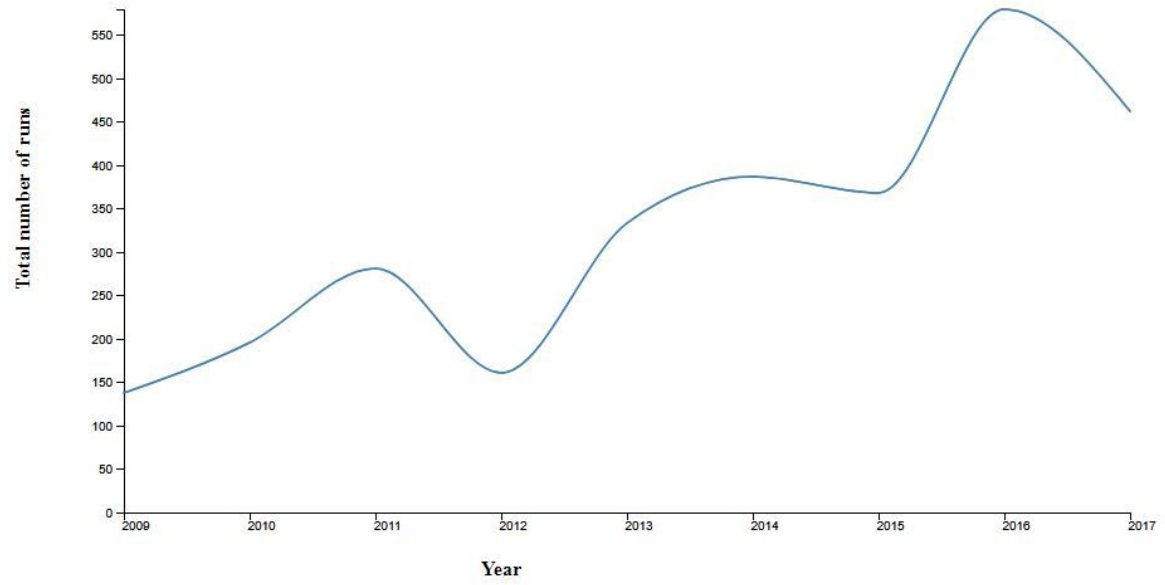
**Bar chart for IPL data during 2008**



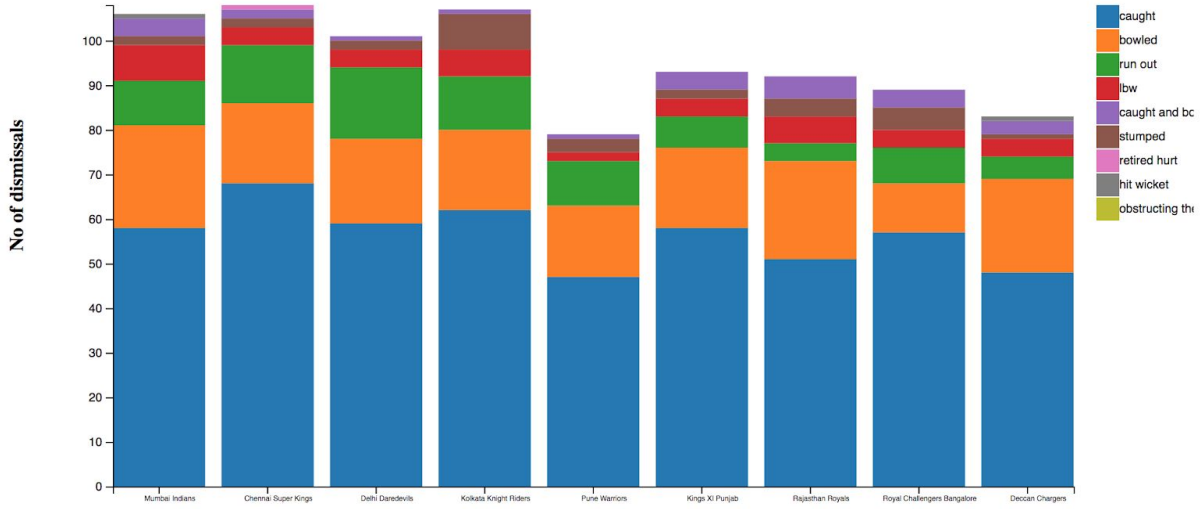

**Bar chart for IPL data during 2009**

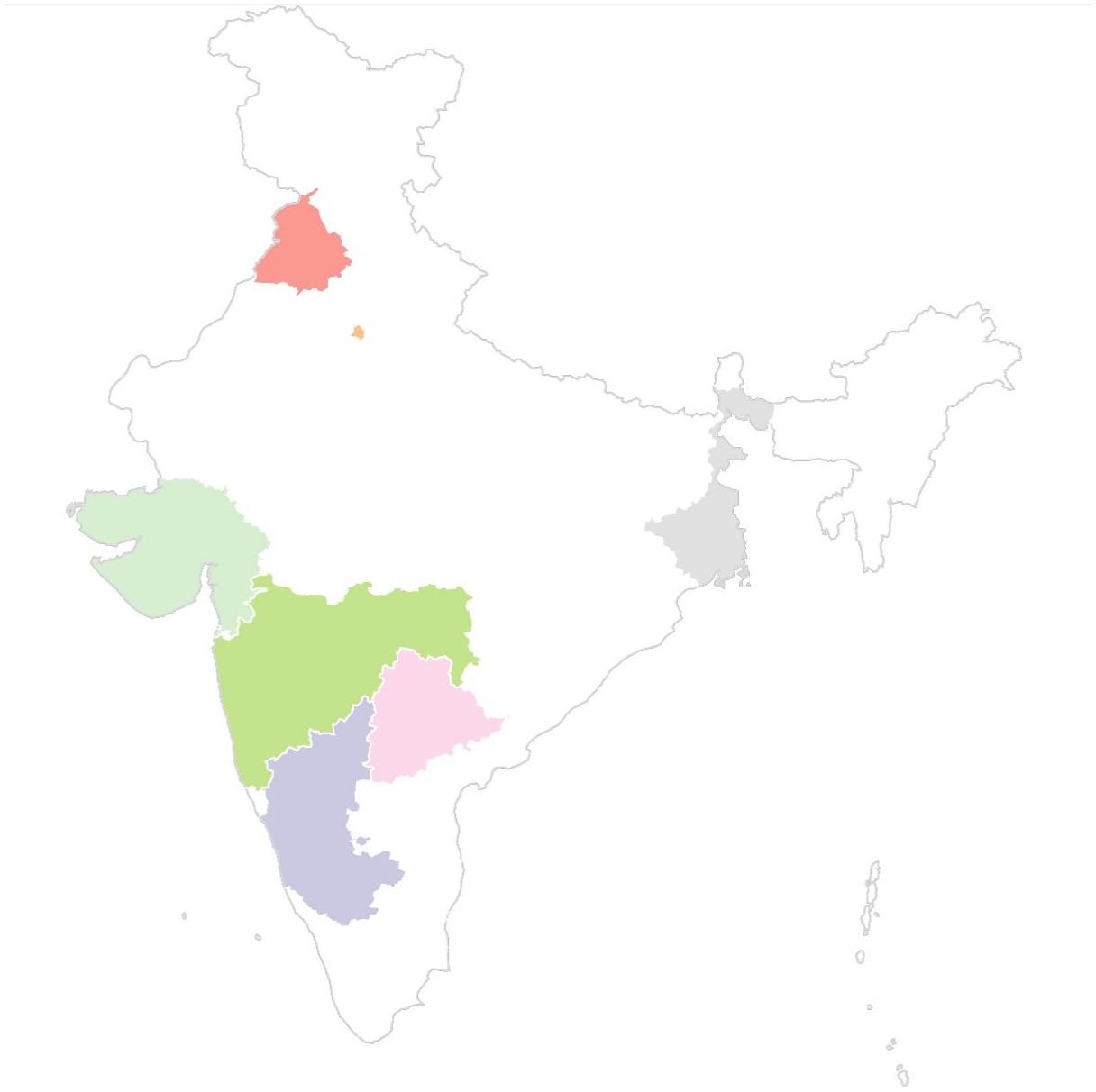


**IPL Data - Line chart for Batsman: DA Warner**




**Stacked Bar chart for IPL data during 2012**



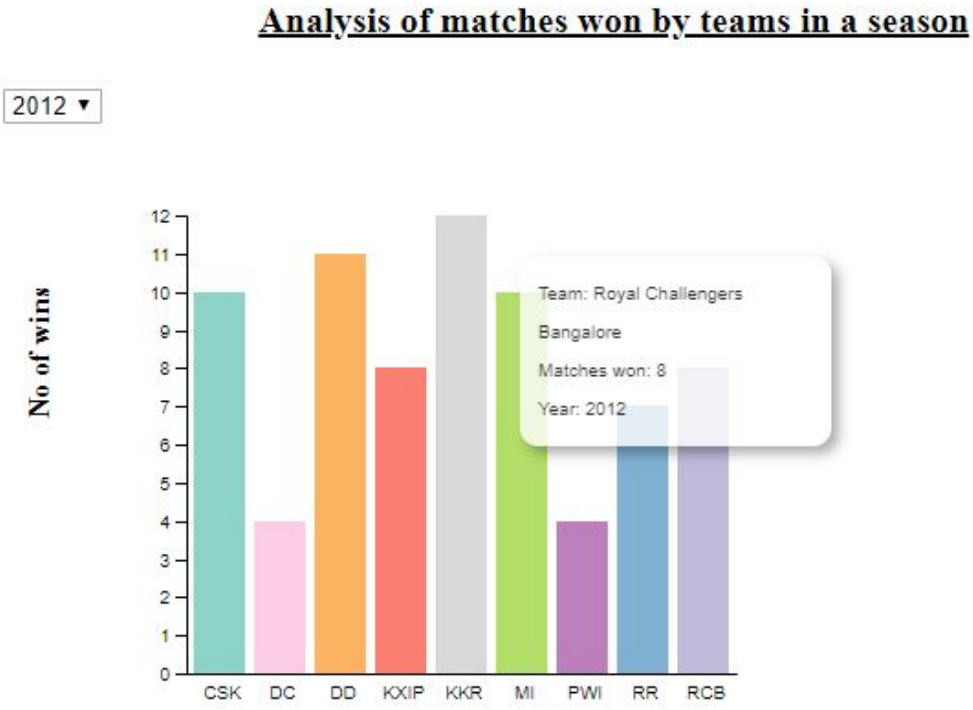


We did not deviate from the project proposal and were able to plot all the promised visualizations successfully. We were also able to complete one of the Optional Features.

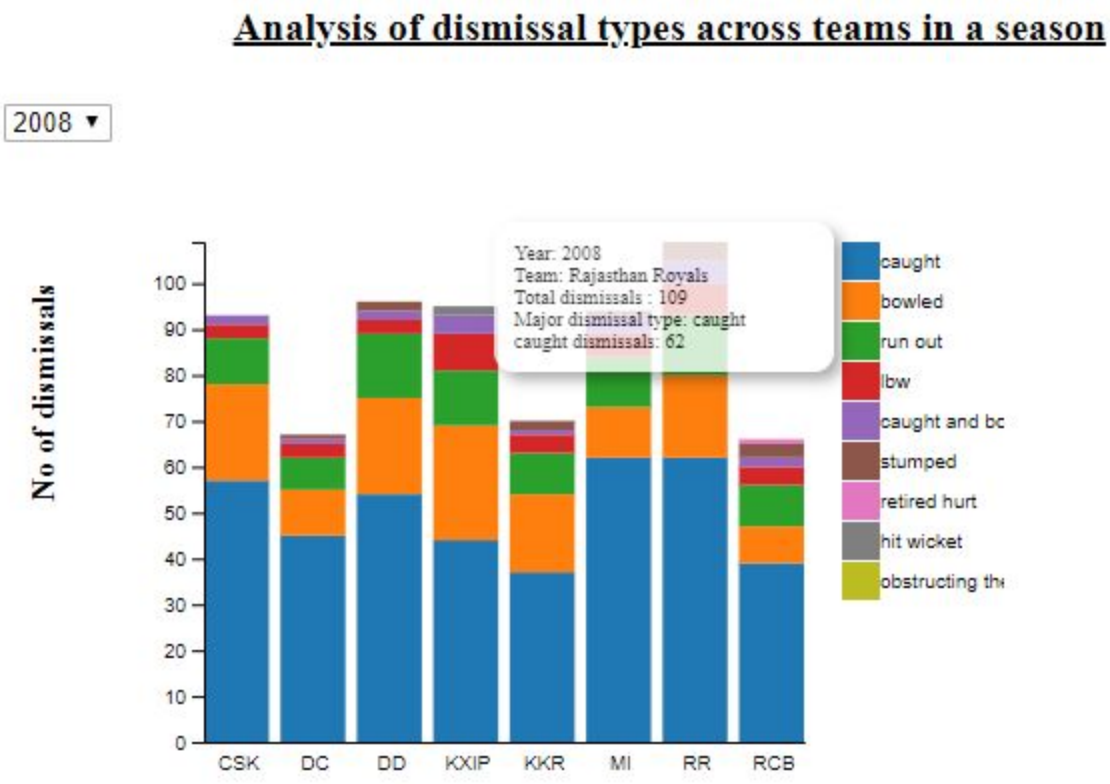
## 7. Implementation

We had brainstormed a bunch of different visualizations and these were finalized-

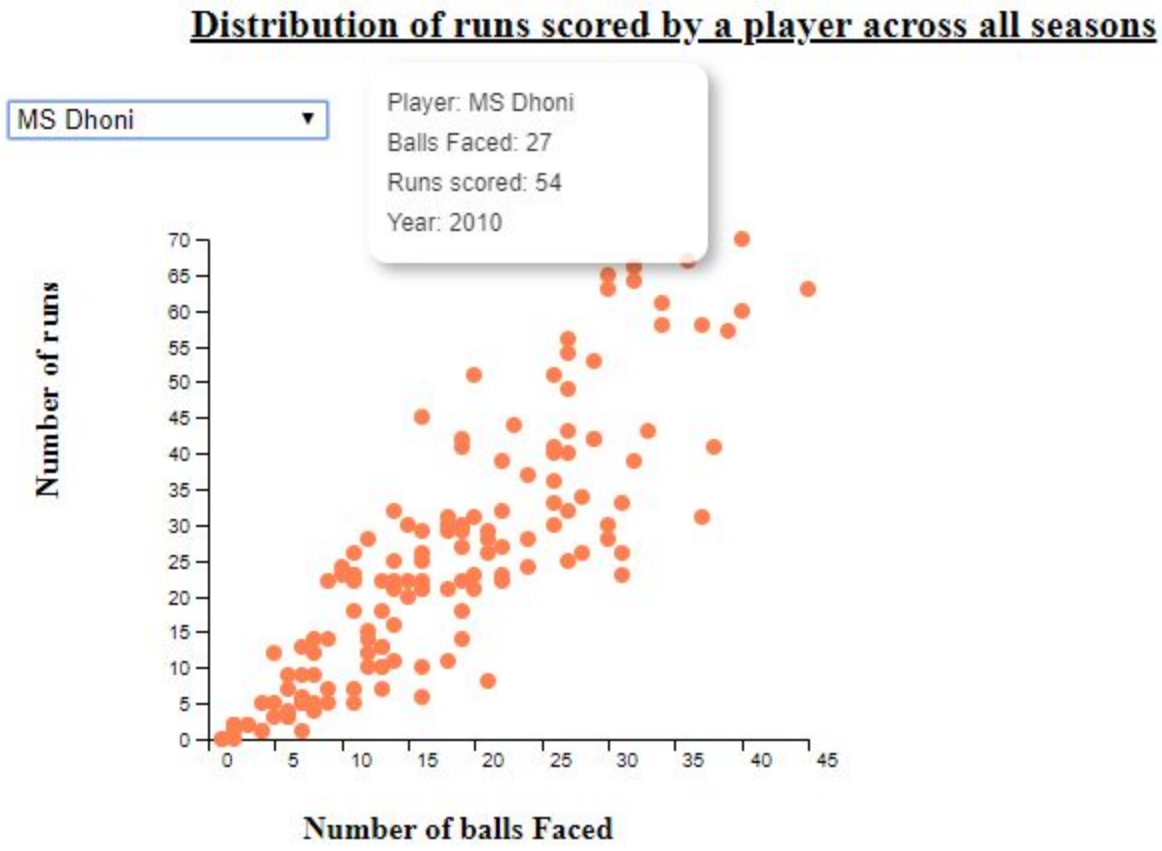
- **BarChart:**
  Filter: Year
  X-axis: all the teams
  Y-axis: Number of matches won
  Drop down: Total number of runs scored and number of matches played



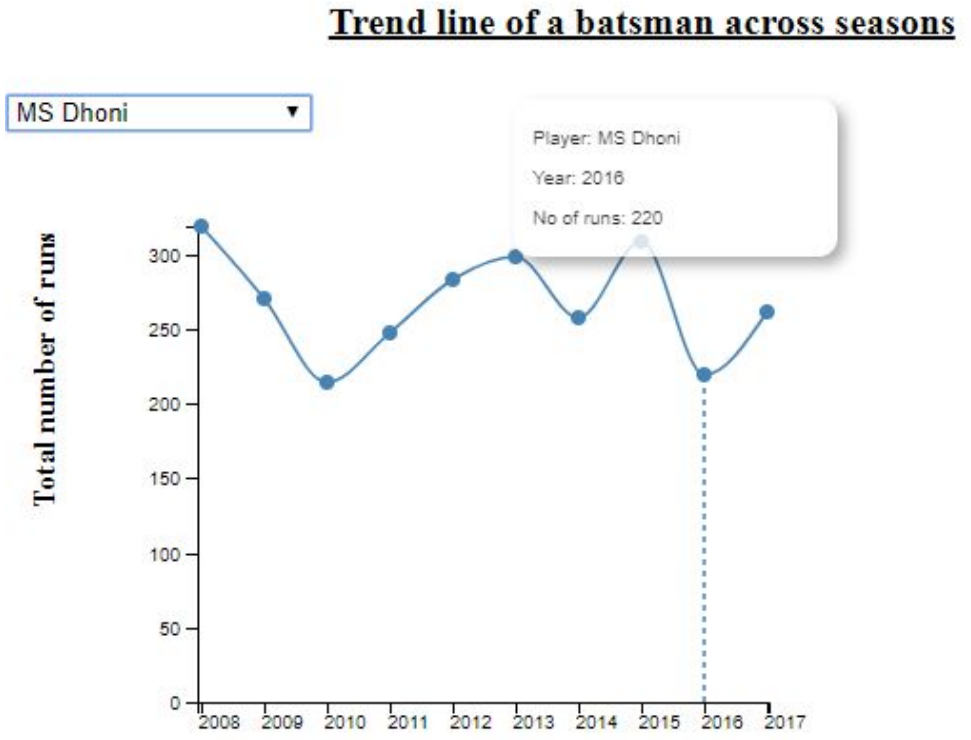
- **Stacked Bar Chart:**
  Filter: Year
  X-axis: All the teams
  Y-axis: Number of dismissals
  Type: stacked bar chart with different dismissal types legend with colors



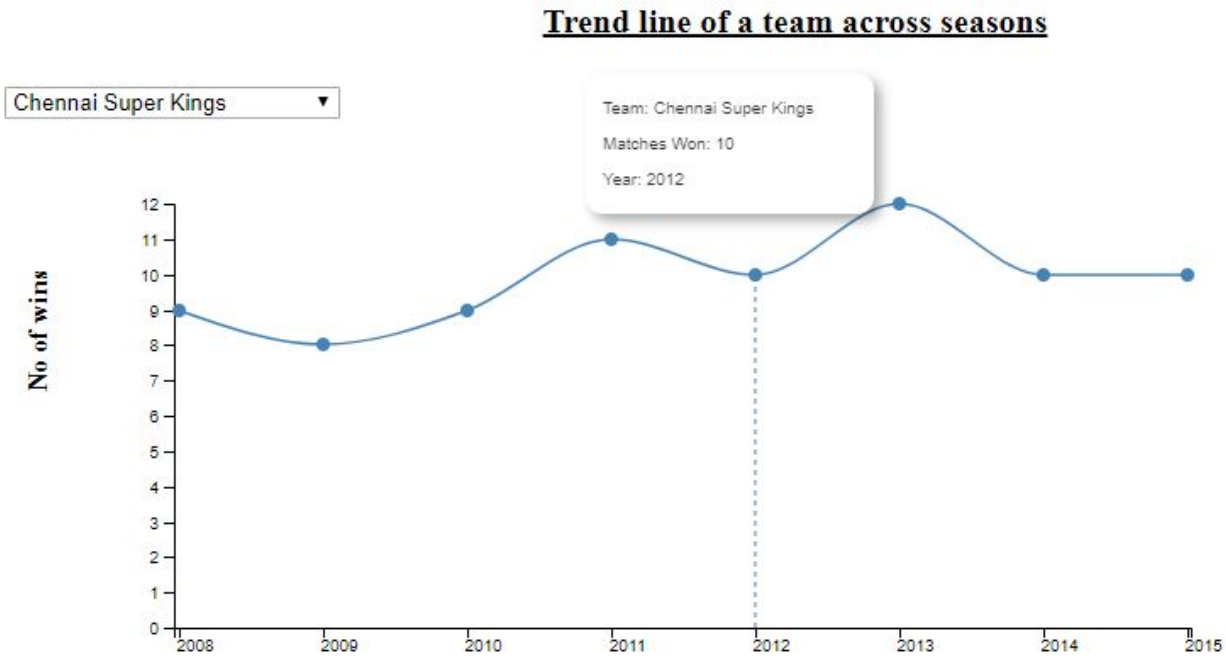
- **ScatterPlot:**
  Player level analysis:
  Filter: Batsman
  Each Datapoint :
  X-axis: Total no of runs scored (in each match)
  Y-axis: Total no of balls faced (in each match)
  More interactions can show which Year and Match details



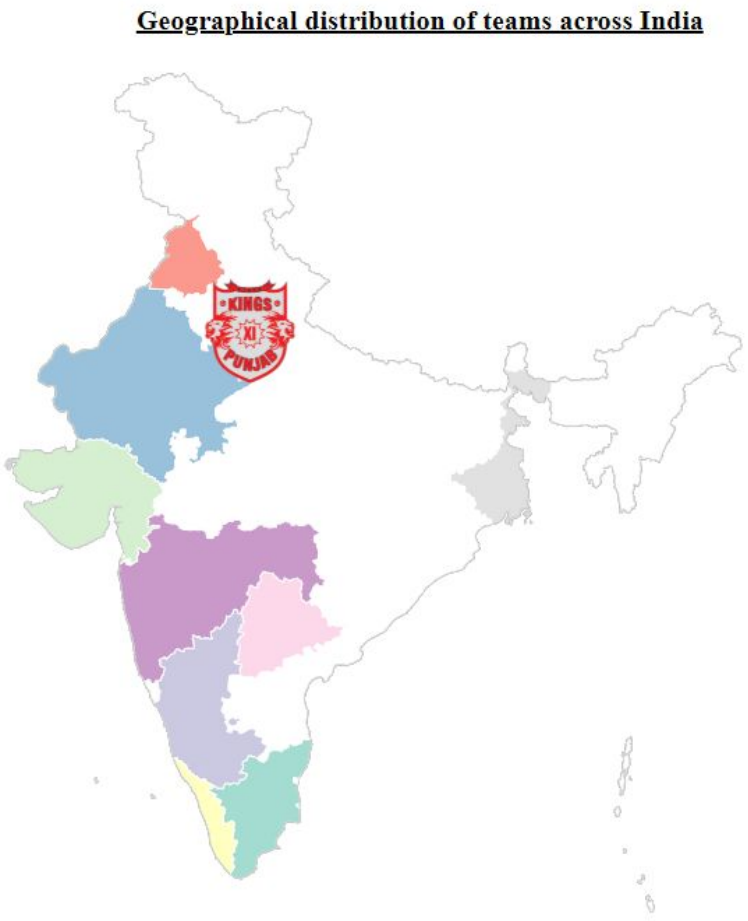
- **Line Trends:**
    1. Player performance over the years
       Filter: Player
       X-axis: Year
       Y-axis: Number of runs scored



2. Year-wise trends for all the teams
   Filter: Team
   X-axis: Year
   Y-axis: Number of matches won



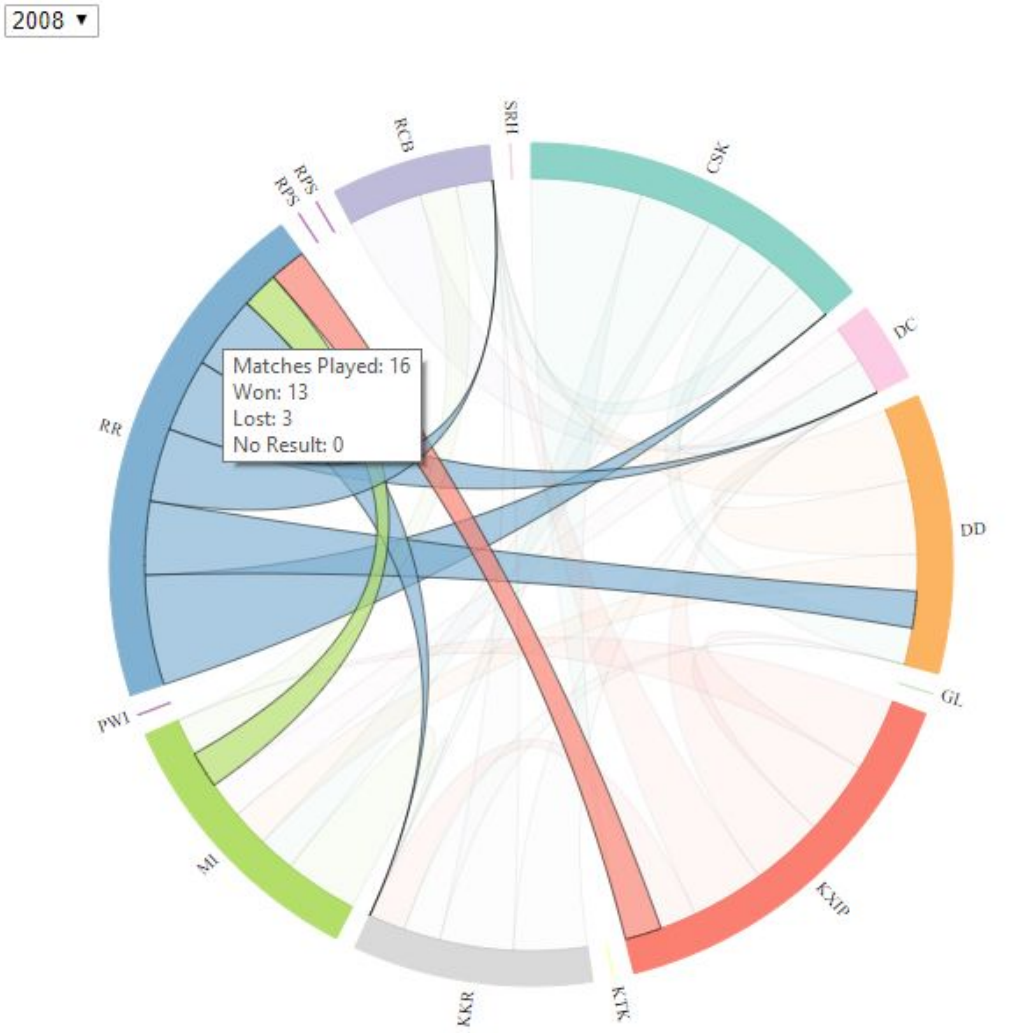
- **GeoMaps:**
    1. Starting with India's map we can plot states corresponding to the team which will be colored by the team color.
       Filter: Year
       On mouse over each region, we can see the team logo.




    2. Plot a world map with the location of all the players
        a. The dataset does not have any attribute related to the "Home Country" of the Player, hence this visualization was not implemented.

- **Chord Diagram:**
    - Mouseover to focus on a team
    - The tool tip can give you the overall matches for a team
    - The thickness of links between countries can encode the relative frequency of matches between two teams: Thicker links represent more matches which result in more wins.
    - Links will be directed: So, mouse over on a link/chord to see wins between the linked countries. Links will be colored by the team with more wins.
      Example chord diagram

The idea of chord diagram came from this Example cricket chord diagram . We were looking for some novel cricket visualization techniques to add to our dashboard.



## 8. Evaluation

After completion of the project, we are efficiently able to visualize the cricket data i.e. data about performance of the players and IPL matches for over 10 years (2008-2017) and we identify some trends. Things we can do:

- Visualize maps to analyze teams and matches
- Visually compare the performance of teams over the years
    - Logistics for performance -
        - Number of matches won
        - Number of dismissals
- Analysis of player performance across different teams and across different years, to find out correlations of their performance with team or their experience as a player in IPL
    - Visually compare the batting average for different players over the years

All the visualizations work function smoothly and are not cluttered. They convey the message in a simple yet effective way.

### 8.1 Challenges & Improvements:

- One of the major challenges we faced was to choose colors for the IPL teams in the bar charts. We started out with choosing the Jersey colors of teams as the color of the bar but that looked too bright and cluttered. Hence, we decided to use a standard color palette which made the graphs look pretty as well as solved the problem of clutter.
- Another challenge was to pick type of colors for Stacked Bar chart(Showing dismissals) and BarChart (Number of runs v/s Teams) as they represent different things. Most of the standard palette had similar colors. We ended up choosing darker colors for dismissals and lighter ones for teams.
- In geomap for India in which we wanted to represent geographical locations of the teams, we used the same color palette as the bar chart. The problem here was that there were two different teams ***"Mumbai Indians"*** and ***"Pune Warriors"*** from the same state (as we colored the state and these two cities are in the same state). We thought of using a different color altogether if both the teams are present in that particular season or a symbol map for the same. The topojson file which we are using for the coordinates of the geomap does not have coordinates for some cities of India, hence to use symbol maps we would have to extract the coordinates online.
- Interaction in stacked bar chart can be improved by adding a tooltip for each bar in the stack.
- We can also implement a new feature of having multiple trend lines in the same line chart to compare the performance of different teams/players simultaneously.