Data analysis and visualization of Indian Premier League(IPL) matches

Sandeep Khandelwal
skhande@iu.edu

Abhishek Gupta
abhigupt@iu.edu

ABSTRACT

This exploratory analysis will help us visualize the best players and team in previous IPL matches. There will also be a breakdown on whether home pitch was favorable in terms of wins. It will also depict how the players performed and eventually led to a match winning effort. For example if a team has best players but they never performed together in a single match. If they all play well, it will result in a match win. Hence, these facts can be used to predict the winners in upcoming match. Other analysis can be done are players by number of run, wickets, maximum number of six's, four's etc. Man of the match by each season. Average runs scored by each team over by over. It may also have analysis on poorly performing players for each team. Poor players can be based on their bowling or batting scores.

KEYWORDS

ipl, analysis, python, packages, bowling, batting, fielding, match strategy, T20, lbw, wickets, runs, bowled, stumped

1 INTRODUCTION

We would like to analyze IPL match for last decade for all IPL matches played. This analysis will be done on ball by ball data available from previous IPL matches. This analysis will help visualize the data as well from various dimensions of a match, tournament, location, player etc.

1.1 Convincing Motivation

The official T20 [1] site displays data in a very rudimentary way. It doesn't provide good visualization of IPL data. If you visit the page for historical data it shows cross tab data accross various dimensions. It shows year wise data from 2008 which cuts across various dimensions like runs, players, wickets etc.

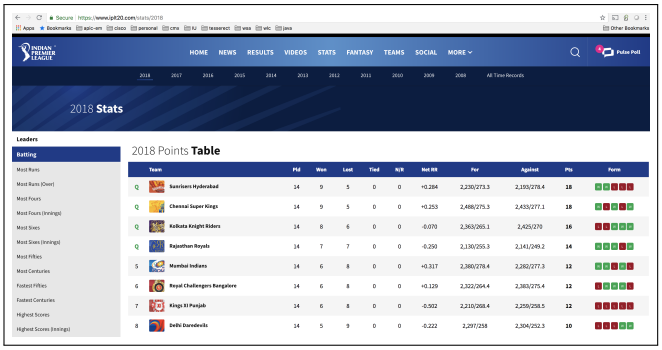

Figure 1: IPL T20 official webpage

The data for all the historical matches is very monotonous and tabular. Tabular representation make it hard to understand and see any type of pattern or anomaly. It also doesn't allow uses to filter the data or aggregate the data for a given range of dates or select a subset of teams. For example one cannot select top batsman from 3 given teams or top bowler from a subset of teams. The data may be sometimes helpful when there is a T20 tournament and only few teams are participating. Another important aspect left out is analysis based on location. As we can see from available leaders it doesn't show any report based on location. This may sometimes help you to answer some questions like how two teams have performed in past on a given location or if this venue has favored home team or if this is a good batting location or a bowling location. These questions really help to predict the outcome of the match. Overall, it lacks

- monotonous
- filter support
- aggregation
- trend / pattern
- location

1.2 Existing Work

For IPL data analysis, I explored few of the existing works to see what kind of insights they provide and what are the drawbacks of this data analysis in terms of visualizing this data.

1.2.1 IPL T20 Official Webpage. As described above, IPL T20[1] official website provides good insight on the data for previous year's IPL tournament but fails to provide good visualization. The visualization provided by this web page is very monotonous and boring. For example, the following figure shows very monotonous tabular data.

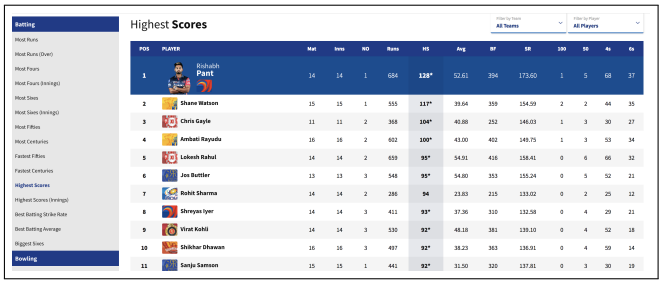

Figure 2: IPL T20 High Run Scorers

1.2.2 CricViz Analysis. CricViz analysis [2] provides visualization using line chart and tabular data. The line charts show match history for all IPL teams and briefly discusses for the teams based on their IPL history. Further provides analysis of delivery and

length of the bowlers. These charts lack depth of analysis of the data as well as cool visualization of the data.

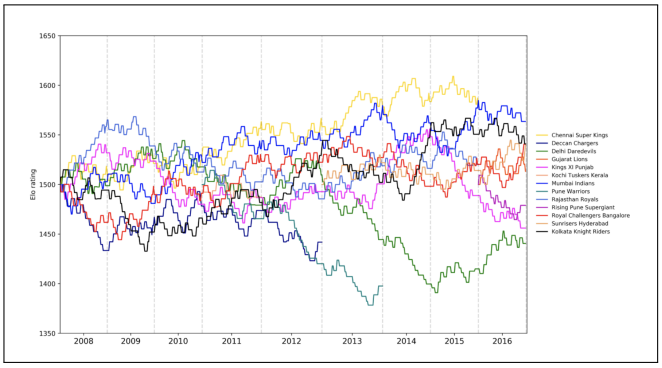

Figure 3: IPL history of various teams

Delivery-Type Analysis

PLAYER	NO MOVEMENT	OFF CUTTER	SLOWER BALL	AWAY SWINGER	IN SWINGER	LEG CUTTER
Bhuveneshwar	60%	8%	1%	20%	10%	1%
McClenaghan	86%	11%	2%	1%	0%	0%
Russell	87%	11%	1%	1%	0%	0%
Mustafizur	53%	46%	0%	0%	1%	0%
Watson	80%	14%	1%	0%	2%	3%

Figure 4: Delivery Type Analysis

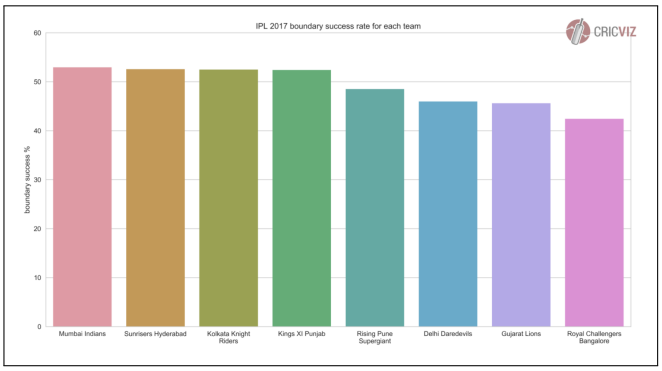

Figure 5: Analysis of IPL T20 matches

1.2.3 Data visualization hacknight: using Pandas for analyzing IPL T20 data. This team tries to visualize [3] the IPL T20 data using some kind of heat maps. Again this lacks most the needed insights.

1.2.4 Reflecting Against Perception: Data Analysis of IPL Batsman . This paper [4] discusses various formulas to come up with a predictive model for IPL matches for example batsman impacting score, run impact score, strike rate impact store etc which is further used to compute the batsman rank etc.

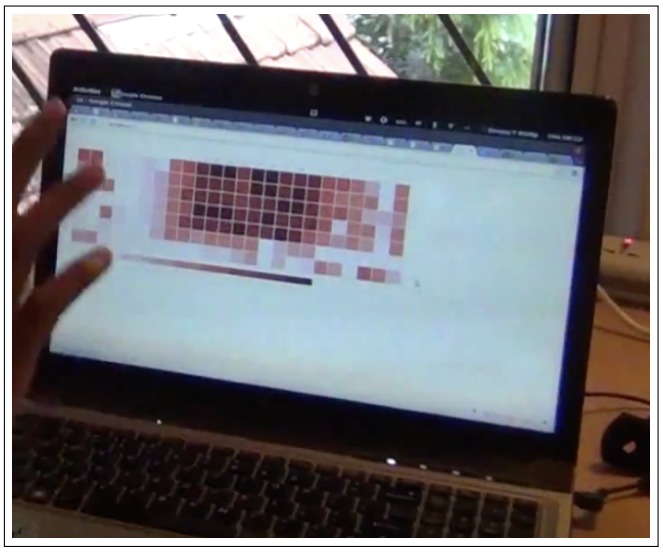

Figure 6: Analysis of IPL T20 data using pandas

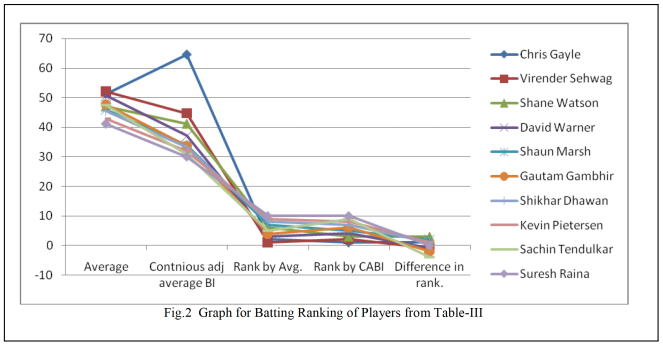

Fig.2 Graph for Batting Ranking of Players from Table-III

Figure 7: Data Analysis of IPL Batsman

This analysis is completely focussed on building predictive model and also predict the batsman performance based on this model. Completely lacks visualization.

1.2.5 Analysis of IPL T20 matches with yorkr templates. This blog [5] tries analyze IPL data and creates modules for various kinds of reports like batsman cumulative strike rate, batsman dismissal etc But this blog fails to provide any details on visualization techniques. The blog mainly focuses on various methods to query the data. It also creates data frame to query the data directly. The methods mainly focus on batsman and bowler statistics functions. These include

- Runs vs deliveries
- Batsman 4s and 6s
- Batsman dismissals
- Runs vs Strike rate
- Batsman Moving Average
- Batsman cumulative average
- Batsman cumulative strike rate
- Batsman runs against oppositions

- Batsman runs vs venue
- Batsman runs predict
- Bowler mean runs conceded
- Bowler Moving Average
- Bowler cumulative average wickets
- Bowler wicket plot
- Bowler wicket against opposition
- Bowler wicket at cricket grounds

These methods are close to the kind of reports and visualization we should be doing for IPL T20 matches but doesn't provide details of any suggested visualization methods.

1.2.6 Contribution. As a part of this analysis of IPL data, our goal here is to provide more meaningful visualization to this data for example if I want to see how tournament progressed in year 2009 or which venues are high score venues. Map kind of visualization will be much better than showing just a tabular data. Or showing Gantt's chart make more sense for showing tournament progress rather than showing tabular data. As a part of this analysis, we have tried to look at the statistics shown during a live match and created similar and more meaningful visualizations for historical data. Another example could be on how to show highest run scorers? In all above methods, the output is demonstrated using tabular data which is good but doesn't visualize the data correctly. Similar is the case of bowlers who took highest wickets. For these two use cases I have used treemap which make more sense which represents a larger tile for the player who scored highest or took maximum wickets along with the team to which he belongs and then reducing the tile size for the players who scored less and so on.

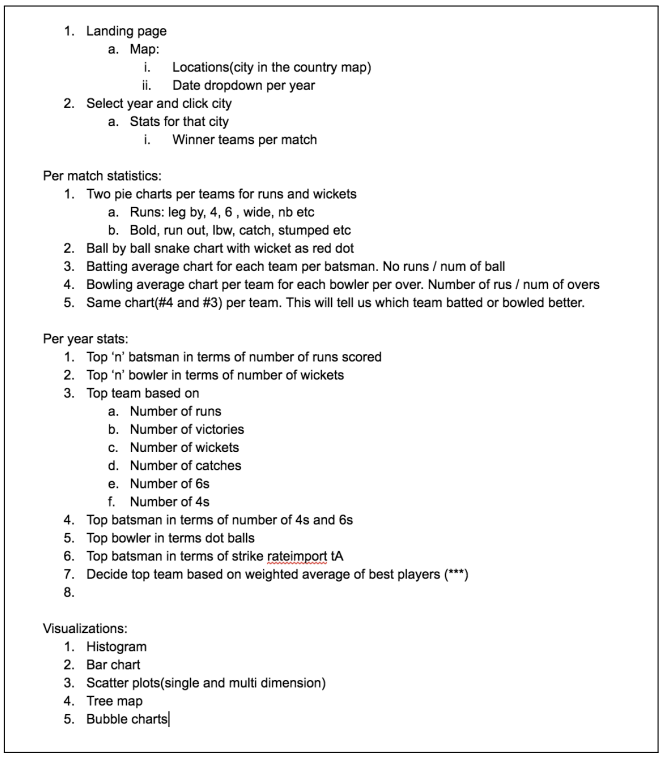

Figure 8: Concept and Design Layout

2 DATA AND METHODS

2.0.1 ideas, sketches, prototypes. The idea here is to show two type of drill downs of the tournament, one is location based and second it year based. For example, if we open a map we should be able to select a location and further visualize the data based on location as a dimension. Also, we should be able to further apply additional filters like team, year etc. Another visualization dimension could be year of the tournament where any visualization could be filtered based on a particular year or a subset of years. These a two primary approaches to start visualizing data. Further, we should be able to do analysis on match level or player level statistics. Match level statistics could be like best teams or how the teams progressed during tournament. As we can see in figure 8 this was the high level design of the data we wanted to visualize for IPL T20 matches.

2.0.2 Visualization Methods selection. For visualizing the data as described in figure 8 we took some ideas based on the charts shown during a live match. Here are some of the visualizations we chose to represent this data.

- location: location is always best represented by a map view, hence we chose to show the location of IPL matches on a map
- tournament progress: this kind of data can be represented in a tabular form but doesn't become so intutive where you have to read into the details of the data. It can also be represented in a tree format but representing date along with team would make it clumsy. Hence, we chose a Gantt chart which is a perfect fit to represent how tournament progressed.
- match progress: a live match shows the progress using a snail chart and we got inspired by the snail chart to show this visualization. Snail chart shows the match progress using two line charts and also how wickets fell while runs being scored by two teams.
- runs and wickets: these are two important aspects of a cricket match and we chose to represent these using a boxplot which will provide us details on median, min, max along with limits and team names for each season.
- top bowler and batsman: we chose to use treemap which gives a good visualization representing larger tile for top batsman and bowler. This will also give details of the team to which this player belongs to.
- dismissal: dismissal has different types like stumped, bowled, run out, caught, hit wicket, lbw etc. A pie chart can represent the numbers better how a player bowls and results in what kind of dismissals. Usually spin bowlers result in maximum caught and fast bowlers result in lbw or bowled.
- team performance: to choose the top team for a particular season or multiple seasons we chose bar chart which gives good comparison or which teams performed better in a tournament.

- batsman performance: batsman performance is represented by strike rate(especially) in a limited overs match like T20. For this form of cricket, we chose bar chart which gives a good comparison of strike rate of various players. Its is important to note that a player can play for different team in different season, hence a bar chart becomes stacked chart for accommodating this scenario.
- runs scored: we chose to use bubble chart, since the size of bubble can clearly represent the high run scorers within a tournament or across multiple tournaments.

3 TECHNOLOGY METHODS

We used Tableau [6] desktop to visualize this data. Tableau makes it easy to import different type of datasources and visualizing the data easily using these imported data sources. There is a slight learning curve involved in understanding the tool and visualization methods. Some visualizations require creating calculated measures. We created these calculated measures like

- count of 6s
- count of 4s
- wickets
- strike rate
- dismissal

4 DATA SOURCES

We will be using datasources from IPL [1] official website and Kaggle challenge [7].

5 RESULTS

5.0.1 Novelty and Insights. Based on design we created following visualizations

- Map View: Map view shows location of IPL matches on a world map. Map view has a associated dashboard which allows us to view team runs for each venue and wins by team per venue.
- snail chart: snail chart shows match progress based on a match ID. It shows total runs on y-axis and overs on x-axis with two line charts for each team. On the line chart it shows the wickets i.e. how players got dismissed. Line chart shows how the teams achieved their target.
- winners chart: its a bar chart which show which team was ranked best for a given season. You can select one or more years for this chart. The hover on the bar shows the rank of the team.
- Toss Winner: this chart shows a comparison of the number of times team won the match vs number of times team won the toss. You can add a filter for a year.
- Top Batsman: This chart is a treemap which shows a bigger tile for the team who has the top run scorer and then shows smaller tile for the other players in its team. This chart can be filtered by year. It gives a good idea of top run scorers for the season and the team.
- Top Bowler: this chart is similar to top batsman where each tile shows the bowler who took maximum number of wickets as well as the team to which he plays. We can filter the data based on year of the tournament.
- Bowling Extras: The chart represents the teams who gave lot of extras to their opponent. This chart is a bubble chart and size of bubble represent the team which gave more number of extra runs. We can select one or more years to visualize this data.
- Top Hitters: This chart represents the players who has big hitters in terms of 4s and 6s. This is important chart since it tells which batsman were the most big hitters for a given season or multiple seasons. This chart is represented by a treemap.
- Runs Scored: This chart represents the number of runs scored in a season. It shows median, low, high watermarks. Also, it shows where each team lies on the boxplot in terms of runs scored.
- Wickets taken: Similar to runs scored chart, it shows number of wickets taken by each team in the tournament. Also, it shows the median, min, max watermarks for the boxplot.
- Match Progress: this chart is a Gantt's chart which shows the how teams played for a given season along with actual match dates. It shows when the teams played and how they reached to the end of the tournament.
- Wins By Venue: Chart shows the number of matches won by a team at a particular venue. The filter to this chart is location. This is a bar chart.
- Runs By Venue: Chart shows runs scored by each team on a particular venue. Its again a bubble chart and size of bubble is proportional to the number of runs scored.
- dismissal type: Its a pie chart and provides a distribution of type wickets taken by a bowler. The wicket(dismissal) can be lbw, stumped, bowled, caught, run out, caught and bowled. It shows dismissal type for each bowler which indicates the bowlers strength in these areas.
- Strike Rate: Indicates how fast a batsman scores runs in limited overs game and how powerful as batsman is. This chart is bar chart and becomes stacked bar chart if a player played for multiple IPL teams.
- Man Of Match: This chart is a scatter plot with each dot on the plot indicates number of times a player received man of match award and which team he belongs to. We can apply a year filter to this chart.
- Extra Runs by Bowler: Its again a bubble chart and indicates number of extra runs for each bowl bowled by the bowler. Each dot indicates the bowler and number of extra runs he conceeded and number of ball he bowled.

5.0.2 craftsmanship and details.

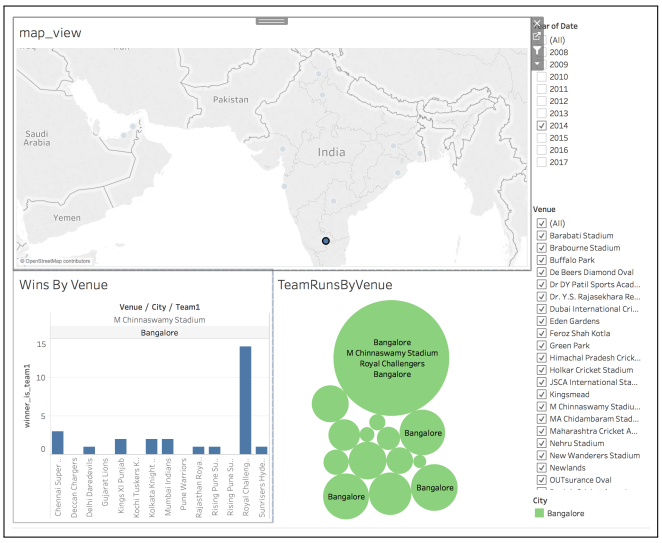

Figure 9: Charts By Venue

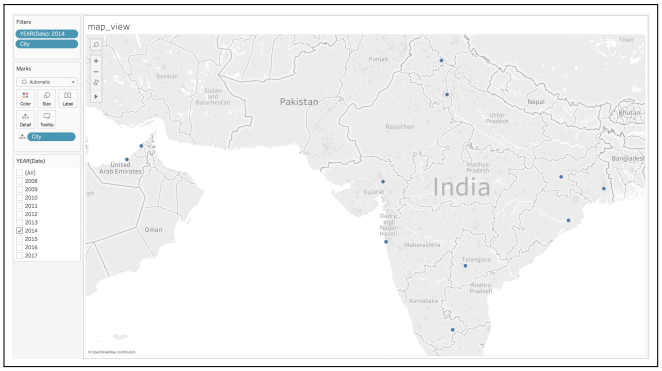

Figure 10: Map View

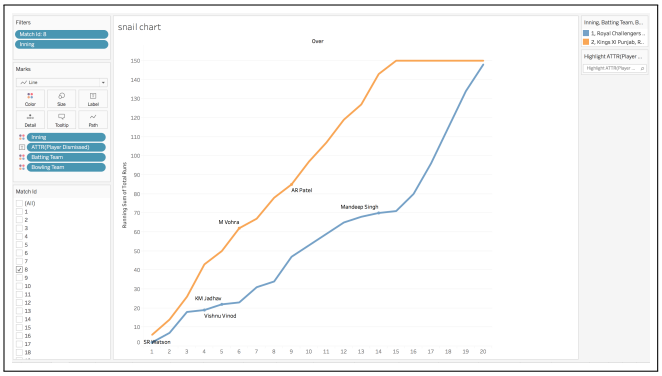

Figure 11: Snail Chart

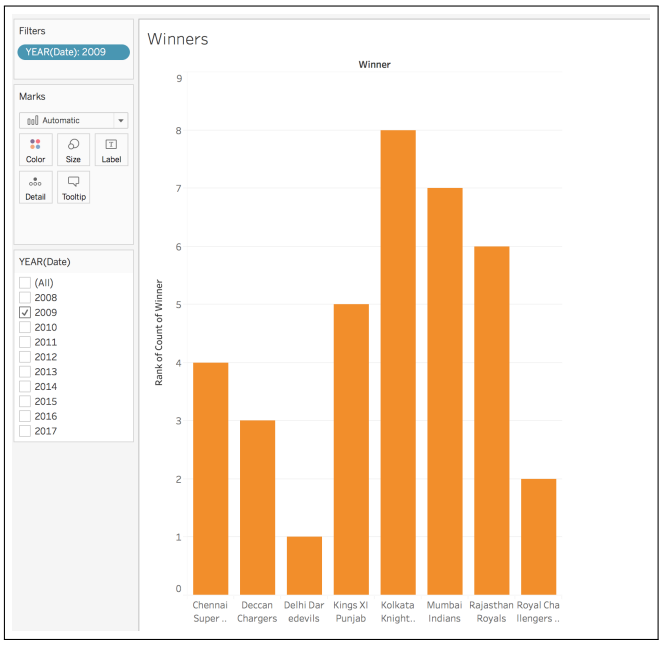

Figure 12: Winners

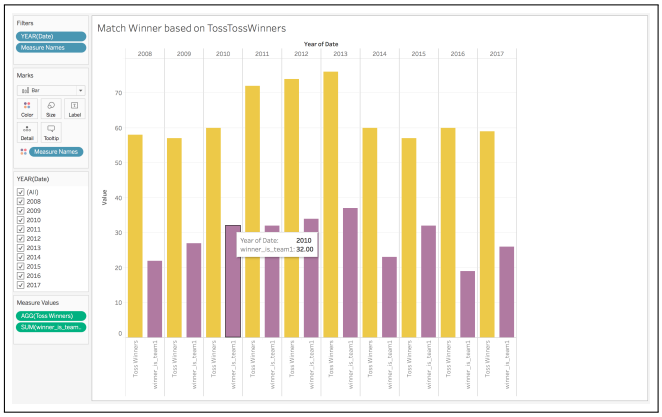

Figure 13: Toss Winners

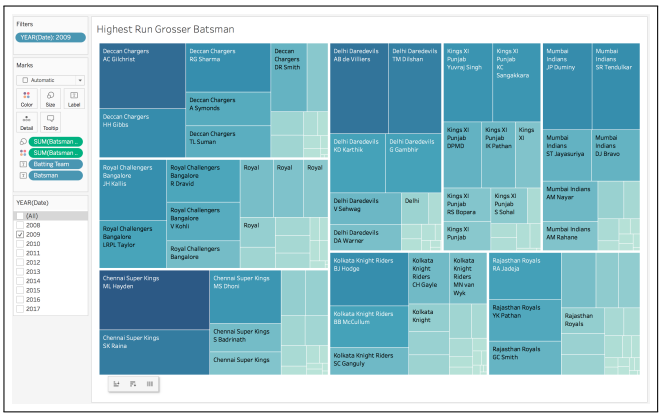

Figure 14: Top Batsman

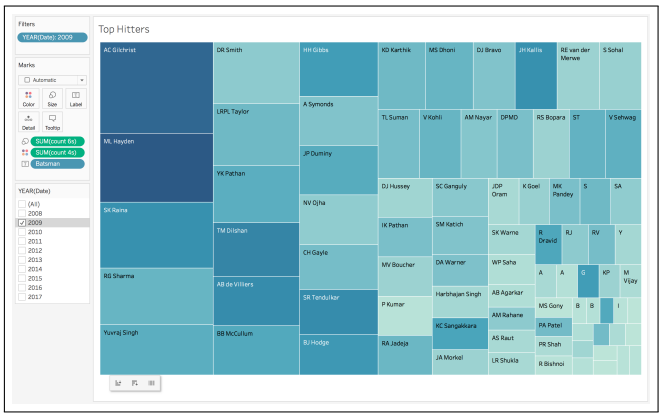

Figure 17: Top Hitter

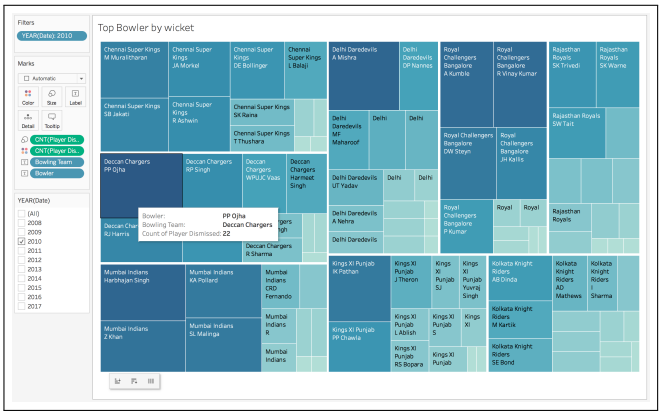

Figure 15: Top Bowler

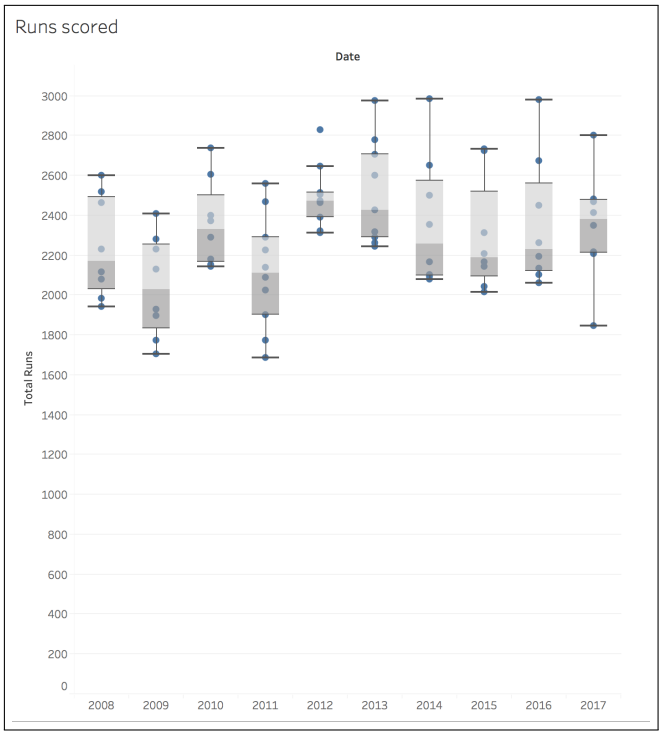

Figure 18: Runs Scored

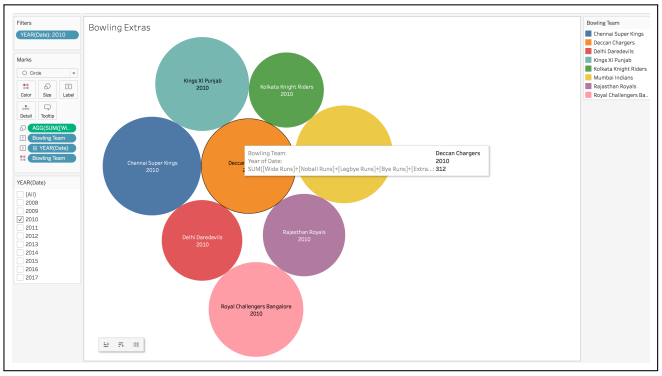

Figure 16: Bowling Extra

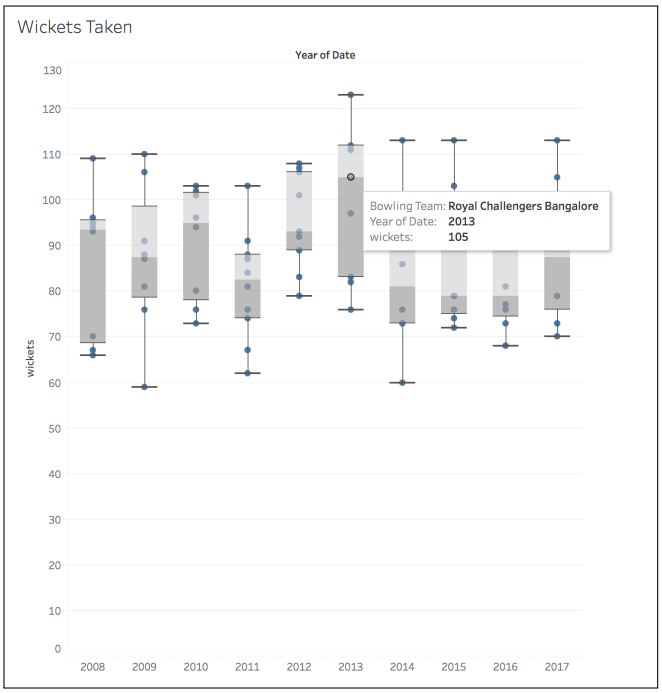

Figure 19: Wickets Taken

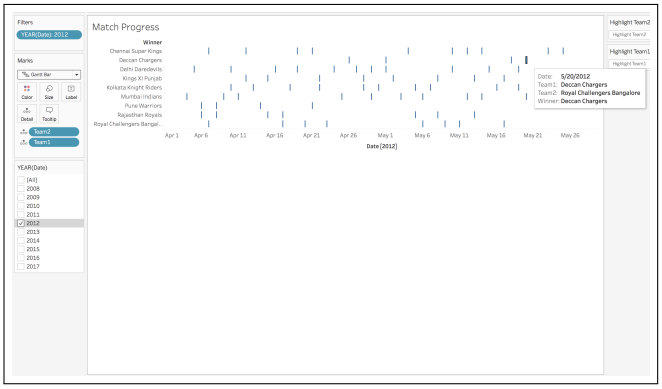

Figure 20: Match Progress

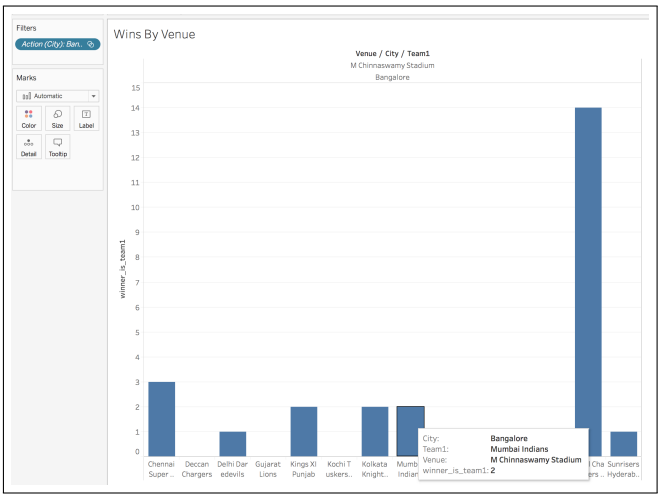

Figure 21: Wins By Venue

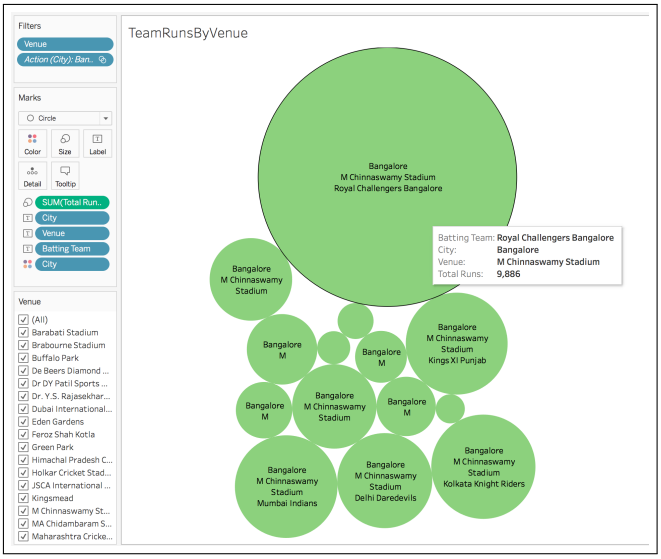

Figure 22: Runs By Venue

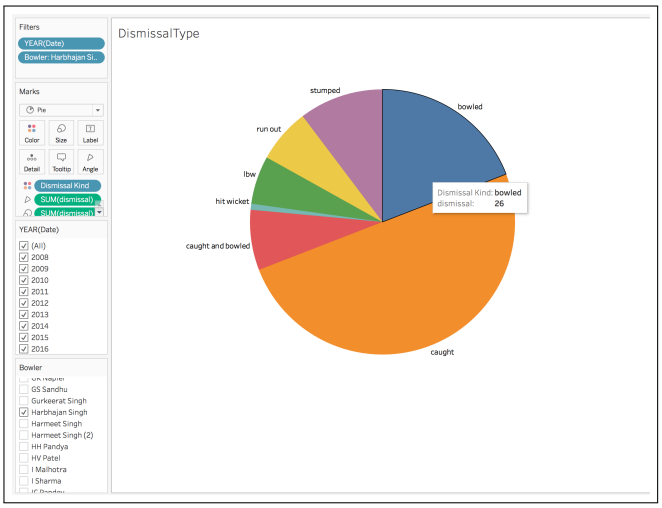

Figure 23: Dismissal Types

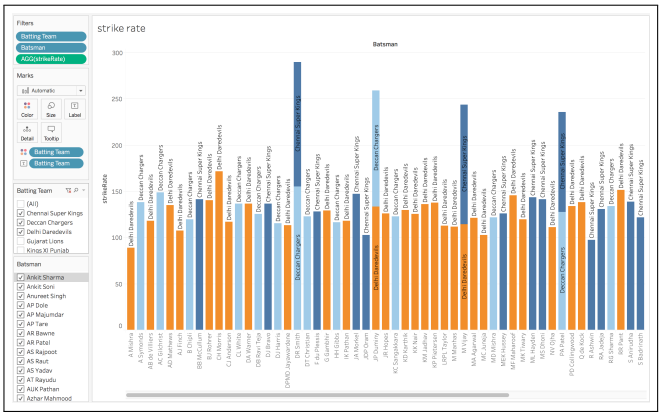

Figure 24: Strike Rate

Figure 25: Man Of the Match

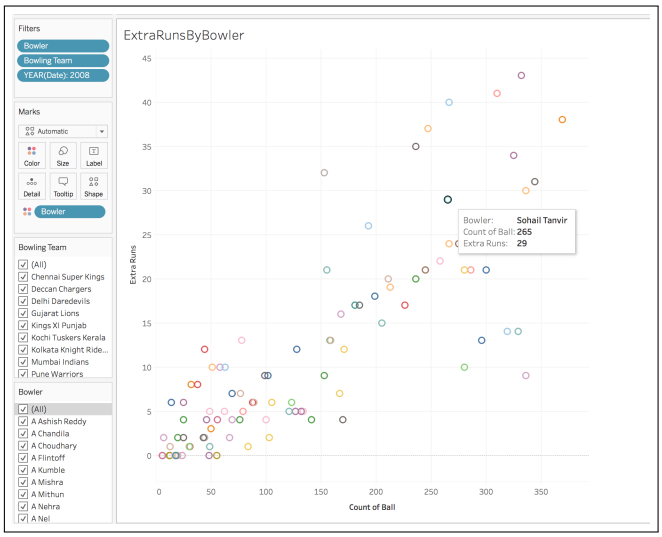

Figure 26: Extra Runs By Bowler

6 DISCUSSION AND CONCLUSION

We faced few challenges while doing this project like

- tooling: we spent sometime in experimenting the right tool/technology. We started off with a javascript based web application and hosting it on a webserver. It was difficult to find a public hosted webserver as well as lot of learning involved in building a full fledge web application. We also tried to put the data in a jupyter notebook using pyplot and pandas but again hosting jupyter notebook was a challenge and getting user input in jupyter notebook is not so intutive. Further we experimented with tableau and really liked the tool which provides a public hosted app as well as facilitates different kind of visualization libraries.
- technology: Tableau [6] was the tool we decided to go with but it involved some learning. We looked at online tutorials and KB articles to get familiar with the tool. Tableau is pretty rich in providing good collection of visualization libraries.
- hosting: With a custom application there was a challenge to host the application. However, tableau has a public web server where different users can host their application. We hosted our application on iplstats application. 27

These visualizations for IPL data provide a different look to the data as well as help the user to look at various trends in data. Another benefit these charts bring is aggregated view of data. With this view one can see charts not just for one season but more than one season by selecting multiple years. These charts provide good analysis of each season for various dimensions which is not covered by any of the sources explored above. It also visualizes the data using different visualization techniques which make it more informative. This can be further extended for providing trend analysis as well as doing some predictive analysis of IPL matches.

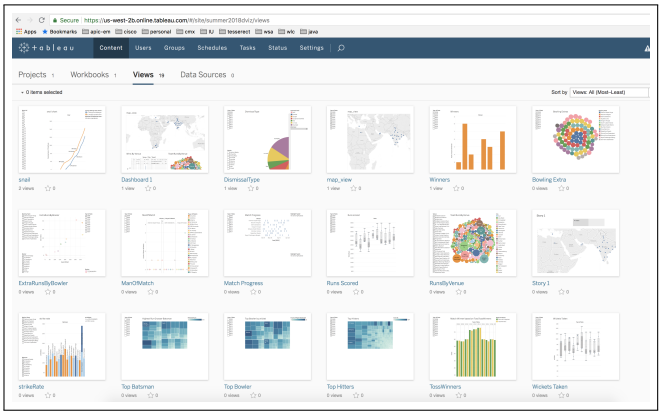

Figure 27: Project Hosted On Tableau Public Webserver

7 ACKNOWLEDGEMENTS

The authors thank Prof. YY Ahn for his technical guidance. The authors would also like to thank TAs of Data Visualization class for their valued support.

8 REPO

All project and report document can be found at github project. Project is hosted on iplstats application

REFERENCES

[1] https://www.iplt20.com. IPL T20, official website. Web Page. Accessed: 2018-05-26.
[2] Rishabh Pant. Cricviz analysis. Web Page. Accessed: 2018-07-20.
[3] Deepu Thomas. Data visualization hacknight: using pandas for analyzing ipl t20 data. Web Page. Accessed: 2018-07-20.
[4] Amit Kumar and Ritu Sindhu. Reflecting against perception: Data analysis of ipl batsman. 2014.
[5] Tinniam V Ganesh. Analysis of ipl t20 matches with yorkr templates. Web Page. Accessed: 2018-07-20.
[6] Tableau Software. Tableau. Web Page. Accessed: 2018-05-26.
[7] Manas Garg - Kaggle. Indian premier league (cricket) ball-by-ball cricket data. Web Page. Accessed: 2018-05-26.