Final Project - Suthi de Silva - CSC 285 - 18th Jan 2024

ICC Cricket World Cup CWC23 All innings

Scenario

Cricket, a globally popular sport (second most popular sport on planet, 2.5 billion fans) is known for its rich history, strategic game play, and passionate fan base. Played between two teams, it involves batting, bowling, and fielding. The game unfolds on an oval-shaped field, and the objective is to score runs while dismissing the opposition's players.

In this project, I embark on a journey to leverage the power of RStudio to manipulate, clean up, and visualize data from the Cricket World Cup. Our data set, sourced in CSV format, holds valuable information about matches, teams, players, and various performance metrics. The goal is to refine this data set, ensuring it is devoid of inconsistencies, missing values, and duplicates.

Research Question – How to manipulate, clean up, and better visualize a Cricket World Cup .csv data set with RStudio?

This data set has following fields,

- **team**: 2 or 3 letter code matching that used in player name description.
- **player**: Standard name used in statsguru innings database. Note: this may not match standard name used in statsguru for teamlists.
- **bat_or_bowl**: indicates whether row represents a batting or bowling innings
- **bb_bf**: Ball Bowled or Balls Faced. Provides a consistent and clean statistic. For bowlers, this is a cleaned version of number of overs bowled. So 1.5 overs becomes 11 in the bb_bf column.
- **runs**: Either runs score by batsman or conceded by bowler. If a batsman produces a not out innings, the data is cleaned to only show the score (eg 100, rather than 100*). Reference can be made to the not_out column to determine if the batsman was out or not.
- **wkts**: Number of wickets taken by a bowler in this innings.
- **wicketball_prob**: Number of wickets taken (or lost by batsman) divided by number of balls bowled or faced. Will be zero for a not out batsman. Can be used to represent the probability of taking or losing a wicket in any given delivery. [This stat developed by JDau].
- **runs_per_ball**: Number of runs score or conceded divided by the number of balls bowled or faced. Represents the average runs score or conceded per ball in this innings. [This stat developed by JDau].
- **opposition**: The team this innings was played against
- **ground**: Which ground in India was the game played
- **start_date**: Which date the game was played
- **overs**: Represents the number of overs delivered by each bowler. This is raw data. Compare to the cleaned bb_bf column. A full over is usually 6 deliveries. A partially completed over will be shown as a decimal point where 0.1 represents 1 delivery. So 1.5 overs means the bowler 1 complete over and 5 additional balls for 11 deliveries.

- **mdns**: Number of maidens a bowler bowled. A maiden is an over of 6 balls that does not concede any runs.
- **econ**: The average number of runs conceded by the bowler per over in this innings. Compare to runs_per_ball column.
- **inns**: 1 means this was the first innings of the day. 2 represents the 2nd innings of the day. So a row that includes the value ‘bowl’ in the bat_or_bowl column and 1 in the inns column indicates the innings in this row occurred when that team bowled first and batted second.
- **4s**: How many 4’s did the batsman score
- **6s**: How many 6’s did the batsman score
- **sr**: The batsman strike rate. This has been converted to runs_per_ball by diving the sr by 100.
- **not_out**: Whether the batsman’s innings was a not out or not. This column removes the need for a * beside the batsman’s score.
- **mins**: Duration of a batsman’s inning in minutes.

(citation for the original data set is given at the end)

Loading data

Here are the first 20 rows of uncleaned data for your reference.

```
## # A tibble: 20 x 20
##    team  player bat_o~1 bb_bf  runs  wkts wicke~2 runs_~3 oppos~4 ground start~5
##    <chr> <chr>  <chr>   <dbl> <dbl> <dbl>   <dbl>   <dbl> <chr>   <chr>  <chr>
##  1 PAK   Shahe~ bowl       60    45     3  0.05     0.75  v Sout~ Chenn~ 27-Oct~
##  2 ENG   DJ Wi~ bowl       60    45     3  0.05     0.75  v India Luckn~ 29-Oct~
##  3 NZ    MJ He~ bowl       60    48     3  0.05     0.8   v Engl~ Ahmed~ 5-Oct--~
##  4 NZ    LH Fe~ bowl       60    49     3  0.05     0.817 v Bang~ Chenn~ 13-Oct~
##  5 AFG   Noor ~ bowl       60    49     3  0.05     0.817 v Paki~ Chenn~ 23-Oct~
##  6 AFG   Mujee~ bowl       60    51     3  0.05     0.85  v Engl~ Delhi  15-Oct~
##  7 ENG   AU Ra~ bowl       48    54     3  0.0625   1.12  v Neth~ Pune   8-Nov--~
##  8 NED   LV va~ bowl       53    60     3  0.0566   1.13  v Sout~ Dhara~ 17-Oct~
##  9 BAN   Mehid~ bowl       54    60     3  0.0556   1.11  v Paki~ Eden ~ 31-Oct~
## 10 PAK   Moham~ bowl       60    60     3  0.05     1     v New ~ Benga~ 4-Nov--~
## 11 SA    G Coe~ bowl       60    62     3  0.05     1.03  v Bang~ Wankh~ 24-Oct~
## 12 SA    G Coe~ bowl       54    68     3  0.0556   1.26  v Sri ~ Delhi  7-Oct--~
## 13 SL    D Mad~ bowl       60    69     3  0.05     1.15  v Bang~ Delhi  6-Nov--~
## 14 AUS   A Zam~ bowl       60    74     3  0.05     1.23  v New ~ Dhara~ 28-Oct~
## 15 NED   BFW d~ bowl       60    74     3  0.05     1.23  v Engl~ Pune   8-Nov--~
## 16 BAN   Shori~ bowl       60    75     3  0.05     1.25  v Engl~ Dhara~ 10-Oct~
## 17 NZ    TA Bo~ bowl       60    77     3  0.05     1.28  v Aust~ Dhara~ 28-Oct~
## 18 BAN   Tanzi~ bowl       60    80     3  0.05     1.33  v Sri ~ Delhi  6-Nov--~
## 19 PAK   Haris~ bowl       48    83     3  0.0625   1.73  v Aust~ Benga~ 20-Oct~
## 20 ENG   RJW T~ bowl       53    88     3  0.0566   1.66  v Sout~ Wankh~ 21-Oct~
## # ... with 9 more variables: overs <dbl>, mdns <dbl>, econ <dbl>, inns <dbl>,
## #   `4s` <dbl>, `6s` <dbl>, sr <dbl>, not_out <dbl>, mins <dbl>, and
## #   abbreviated variable names 1: bat_or_bowl, 2: wicketball_prob,
## #   3: runs_per_ball, 4: opposition, 5: start_date
```

Cleaning up data

After looking at the data carefully, I realized that I had multiple columns and rows with “NA” values that might affect the quality of data visualization, so I would remove those columns.

```
## # A tibble: 729 x 5
##     `4s`  `6s`    sr not_out  mins
##    <dbl> <dbl> <dbl>   <dbl> <dbl>
##  1    NA    NA    NA      NA    NA
##  2    NA    NA    NA      NA    NA
##  3    NA    NA    NA      NA    NA
##  4    NA    NA    NA      NA    NA
##  5    NA    NA    NA      NA    NA
##  6    NA    NA    NA      NA    NA
##  7    NA    NA    NA      NA    NA
##  8    NA    NA    NA      NA    NA
##  9    NA    NA    NA      NA    NA
## 10    NA    NA    NA      NA    NA
## # ... with 719 more rows
```

```
## [1] 11
```

Then after running the above code I got rid of those columns. By **checking current number of columns we confirmed that a column reduction has happened** from 20 to 11.

I would want to add an **ID column for each row** for the data set. **Why?** because each row is a **unique performance by a player**, so it would be important for us to uniquely identify each row, when it comes to **calculation, ranking, and plotting purposes.**

```
## # A tibble: 729 x 12
##    PerformID team  player     bat_o~1 bb_bf  runs wicke~2 runs_~3 oppos~4 ground
##        <int> <chr> <chr>      <chr>   <dbl> <dbl>   <dbl>   <dbl> <chr>   <chr>
##  1         1 PAK   Shaheen S~ bowl       60    45  0.05     0.75  v Sout~ Chenn~
##  2         2 ENG   DJ Willey~ bowl       60    45  0.05     0.75  v India Luckn~
##  3         3 NZ    MJ Henry ~ bowl       60    48  0.05     0.8   v Engl~ Ahmed~
##  4         4 NZ    LH Fergus~ bowl       60    49  0.05     0.817 v Bang~ Chenn~
##  5         5 AFG   Noor Ahma~ bowl       60    49  0.05     0.817 v Paki~ Chenn~
##  6         6 AFG   Mujeeb Ur~ bowl       60    51  0.05     0.85  v Engl~ Delhi
##  7         7 ENG   AU Rashid~ bowl       48    54  0.0625   1.12  v Neth~ Pune
##  8         8 NED   LV van Be~ bowl       53    60  0.0566   1.13  v Sout~ Dhara~
##  9         9 BAN   Mehidy Ha~ bowl       54    60  0.0556   1.11  v Paki~ Eden ~
## 10        10 PAK   Mohammad ~ bowl       60    60  0.05     1     v New ~ Benga~
## # ... with 719 more rows, 2 more variables: start_date <chr>, inns <dbl>, and
## #   abbreviated variable names 1: bat_or_bowl, 2: wicketball_prob,
## #   3: runs_per_ball, 4: opposition
```

Above is how it would look like **with the ID**, and “PerformID” is present there.

After checking ESPN Cricinfo database *(citation is given at the end)*, I realized some specific data in the data set are incorrect, which means I would replace them with actual values.

```
## # A tibble: 9 x 12
##   team  player  bat_o~1 bb_bf  runs wicke~2 runs_~3 oppos~4 ground start~5  inns
##   <chr> <chr>   <chr>   <dbl> <dbl>   <dbl>   <dbl> <chr>   <chr>  <chr>   <dbl>
## 1 SL    D Madu~ bowl       60    69  0.05      1.15 v Bang~ Delhi  6-Nov--     2
## 2 SL    D Madu~ bowl       60    80  0.0833    1.33 v India Wankh~ 2-Nov--     1
## 3 SL    D Madu~ bowl       58    49  0.0690    0.845 v Neth~ Luckn~ 21-Oct~     1
## 4 SL    D Madu~ bowl       54    38  0.0556    0.704 v Aust~ Luckn~ 16-Oct~     2
## 5 SL    D Madu~ bowl       54    48  0.0370    0.889 v Afgh~ Pune   30-Oct~     2
## 6 SL    D Madu~ bowl       56    60  0.0357    1.07  v Paki~ Hyder~ 10-Oct~     2
## 7 SL    D Madu~ bowl       60    86  0.0333    1.43  v Sout~ Delhi  7-Oct--     1
## 8 SL    D Madu~ bowl       30    37  0         1.23  v Engl~ Benga~ 26-Oct~     1
## 9 SL    D Madu~ bowl       38    58  0        1.53  v New ~ Benga~ 9-Nov--     2
## # ... with 1 more variable: PerformID <int>, and abbreviated variable names
## #   1: bat_or_bowl, 2: wicketball_prob, 3: runs_per_ball, 4: opposition,
## #   5: start_date
```

```
## # A tibble: 9 x 12
##   team  player  bat_o~1 bb_bf  runs wicke~2 runs_~3 oppos~4 ground start~5  inns
##   <chr> <chr>   <chr>   <dbl> <dbl>   <dbl>   <dbl> <chr>   <chr>  <chr>   <dbl>
## 1 SL    D Madu~ bowl       60    69  0.05     1.15  v Bang~ Delhi  6-Nov--     2
## 2 SL    D Madu~ bowl       60    80  0.0833   1.33  v India Wankh~ 2-Nov--     1
## 3 SL    D Madu~ bowl       58    49  0.0690   0.845 v Neth~ Luckn~ 21-Oct~     1
## 4 SL    D Madu~ bowl       54    38  0.0556   0.704 v Aust~ Luckn~ 16-Oct~     2
## 5 SL    D Madu~ bowl       54    48  0.0370   0.889 v Afgh~ Pune   30-Oct~     2
## 6 SL    D Madu~ bowl       56    60  0.0357   1.07  v Paki~ Hyder~ 10-Oct~     2
## 7 SL    D Madu~ bowl       60    86  0.0333   1.43  v Sout~ Delhi  7-Oct--     1
## 8 SL    D Madu~ bowl       30    37  0        1.23  v Engl~ Benga~ 26-Oct~     1
## 9 SL    D Madu~ bowl       38    58  0        1.53  v New ~ Benga~ 9-Nov--     2
## # ... with 1 more variable: PerformID <int>, and abbreviated variable names
## #   1: bat_or_bowl, 2: wicketball_prob, 3: runs_per_ball, 4: opposition,
## #   5: start_date
```

```
## # A tibble: 8 x 12
##   team  player  bat_o~1 bb_bf  runs wicke~2 runs_~3 oppos~4 ground start~5  inns
##   <chr> <chr>   <chr>   <dbl> <dbl>   <dbl>   <dbl> <chr>   <chr>  <chr>   <dbl>
## 1 BAN   Shorif~ bowl       60    75  0.05     1.25  v Engl~ Dhara~ 10-Oct~     1
## 2 BAN   Shorif~ bowl       38    34  0.0526   0.895 v Afgh~ Dhara~ 7-Oct--     1
## 3 BAN   Shorif~ bowl       60    51  0.0333   0.85  v Neth~ Eden ~ 28-Oct~     1
## 4 BAN   Shorif~ bowl       57    51  0.0351   0.895 v Sri ~ Delhi  6-Nov--     1
## 5 BAN   Shorif~ bowl       54    76  0.0185   1.41  v Sout~ Wankh~ 24-Oct~     1
## 6 BAN   Shorif~ bowl       24    25  0        1.04  v Paki~ Eden ~ 31-Oct~     2
## 7 BAN   Shorif~ bowl       47    43  0        0.915 v New ~ Chenn~ 13-Oct~     2
## 8 BAN   Shorif~ bowl       48    54  0        1.12  v India Pune   19-Oct~     2
## # ... with 1 more variable: PerformID <int>, and abbreviated variable names
## #   1: bat_or_bowl, 2: wicketball_prob, 3: runs_per_ball, 4: opposition,
## #   5: start_date
```

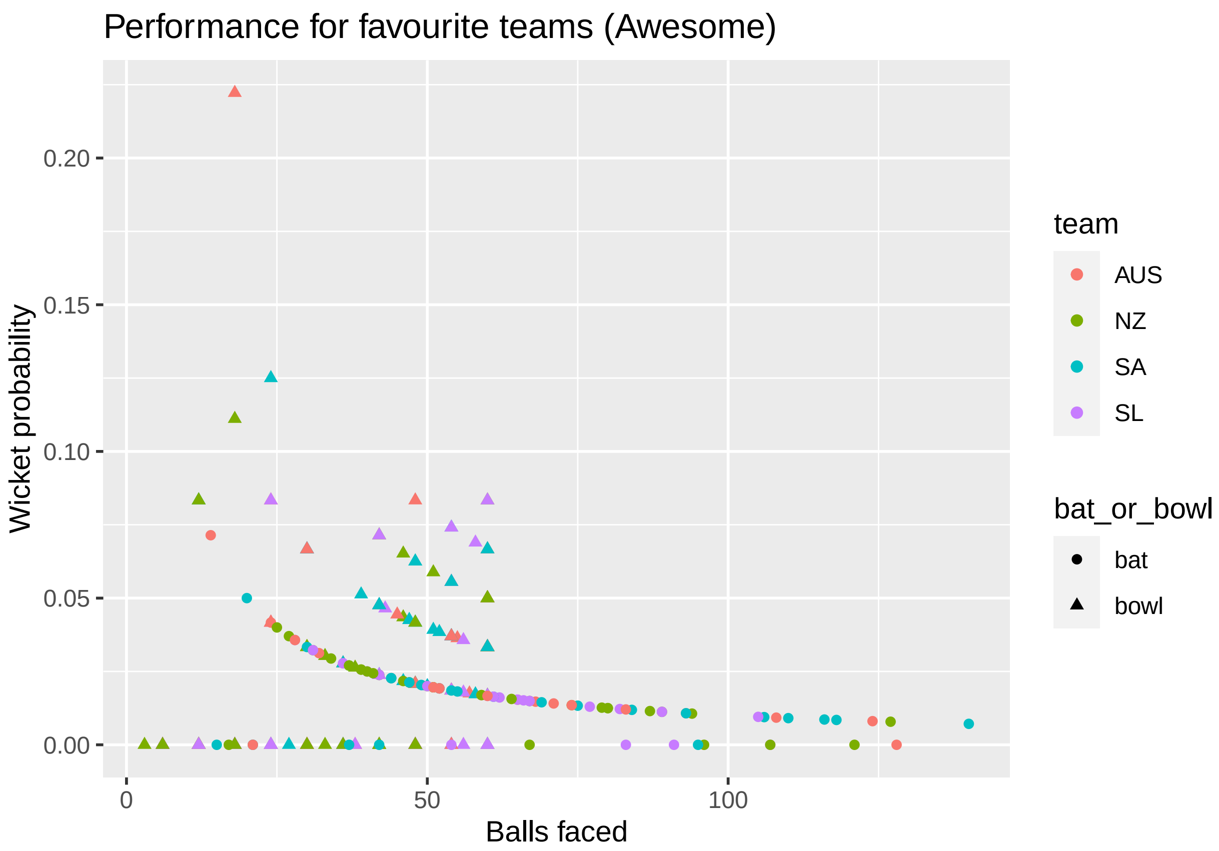
We can observe that the **changes have been made**.

Now, **after carefully observing the data set** further I can safely **confirm** that there are **no outliers, or data type conversions or, negative unexplainable values, or incorrect values exist.** It **makes sense** as in cricket (if you are a great observer) it would be very hard to have outliers or negative values unless it has been entered wrong during the data collection. But I checked the max and min values for each column by toggeling the ordering button, so I can clearly see that the data set is **cleaned enough for further processing**.

Useful information and visualizations derived from graphs

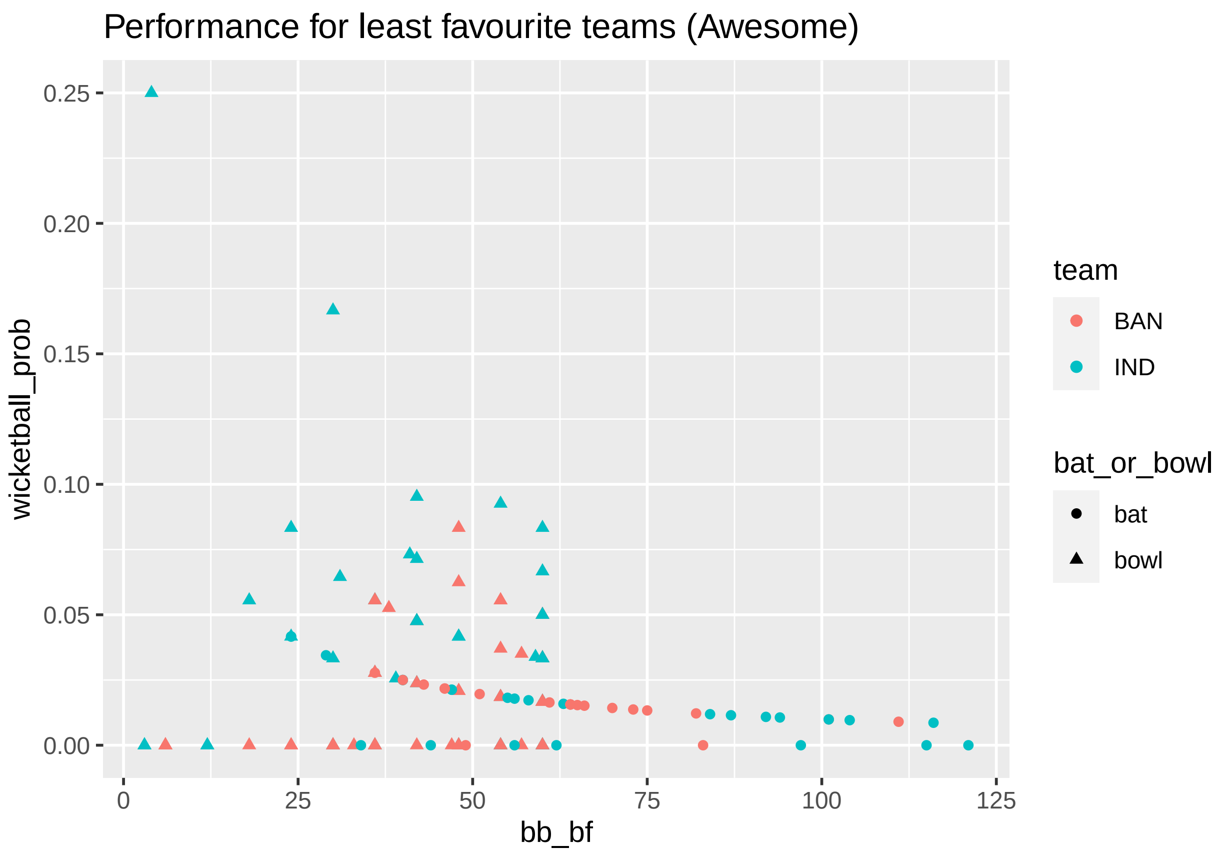
Favourite and least favourite cricket teams' performance (Awesome visualizations)

As for design choice, I think it would be great if we could see a scatter plot of balls bowled or balls faced vs wicket probability for each team as it would help us get some information about the best players of my favorite teams.



The graph makes sense, as we can clearly see there are some decreasing slope correlation with some points, since cricket fans would understand that, wicket probability drastically decrease when bowlers bowl a lot in a single match and then it settles down, specially in longer formats.

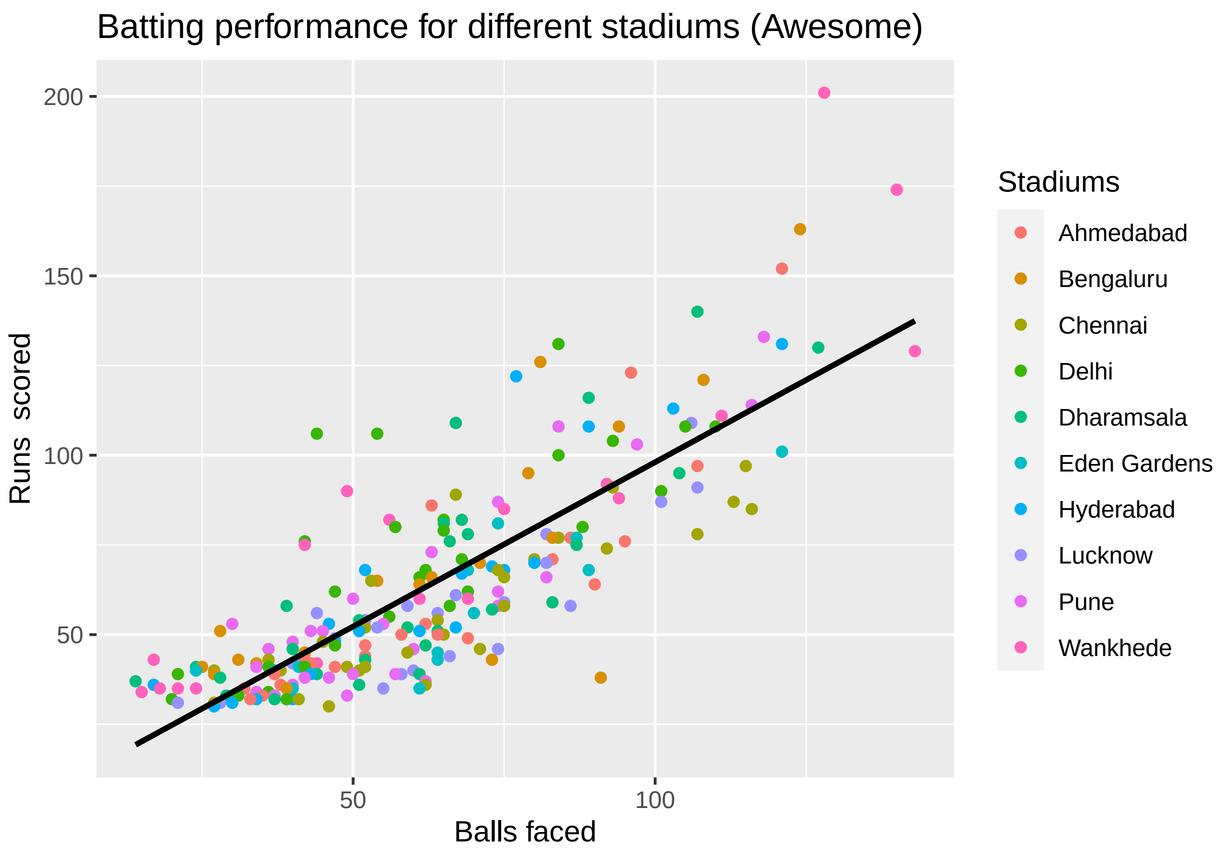
At the same time I would want to see how my least favorite teams (the teams I don't prefer winning) are doing, as it would tell me how is the competition for my favorite teams. So **as for design choice**, scatter plot would be a great option again.



The graph makes sense, as we can clearly see there are some decresing slope correlation with some points just like the situation earlier, and the same way a true cricket fan would understand that, wicket probability drastically decrease when bowlers bowl a lot in a single match and then it settles down, specially in longer formats. At the same time India is better at bowling probabilities so that is why they are doing well in the graph.

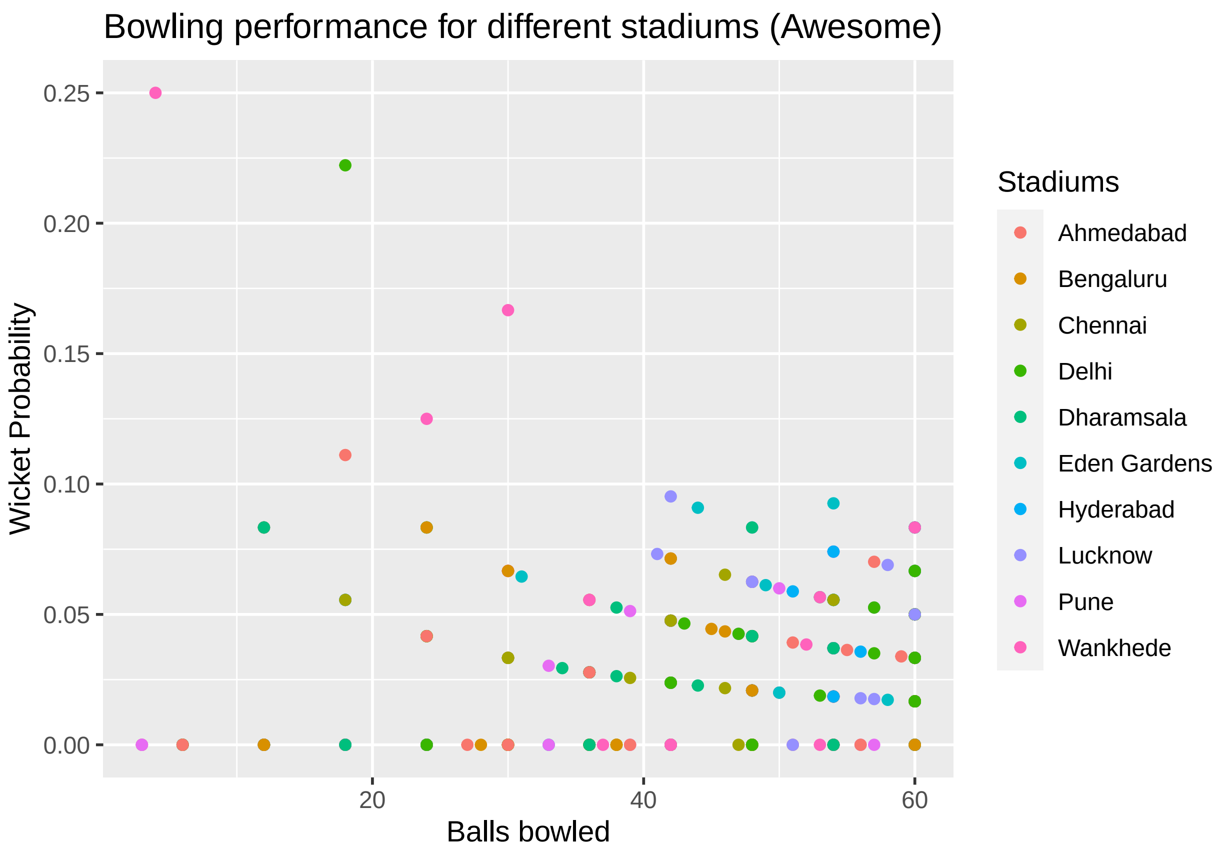
Best stadiums for batting and bowling (Awesome visualizations)

As for design choice, with a scatter plot that shows best stadiums for batting we would be able to determine what are the good stadiums high scoring matches.



It would make sense to have a exponentially increasing relation between balls a batsman has faced and the runs they scored, because obviously they need to face more balls and they gets aggressive with the time when scoring runs. Specifically Wankhede stadium is known for high scoring matches.

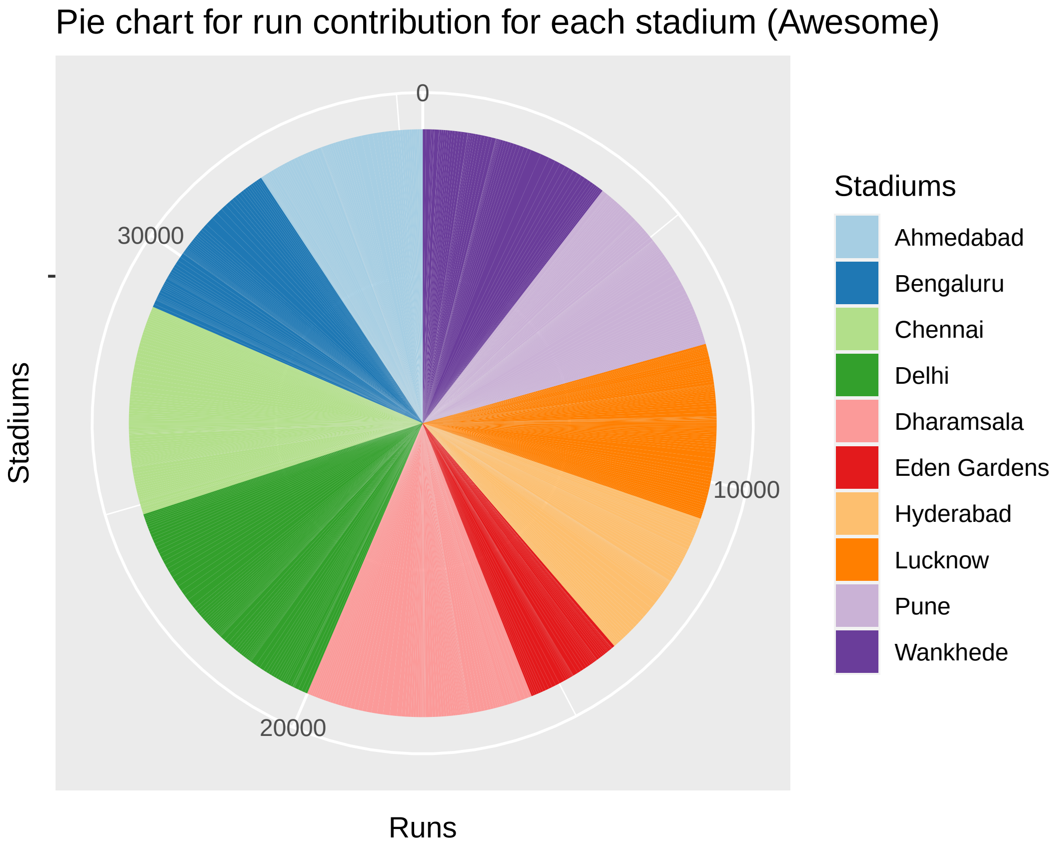
As for design choice, with a scatter plot that shows best stadiums for bowling we would be able to determine what are the good stadiums low scoring matches.



It would make sense to have a decreasing slope relation between balls a bowler has bowled and the wickets, because obviously they need to ball more balls and they get tired with the time when scoring runs. Specifically Wankhede stadium is known for good bowling.

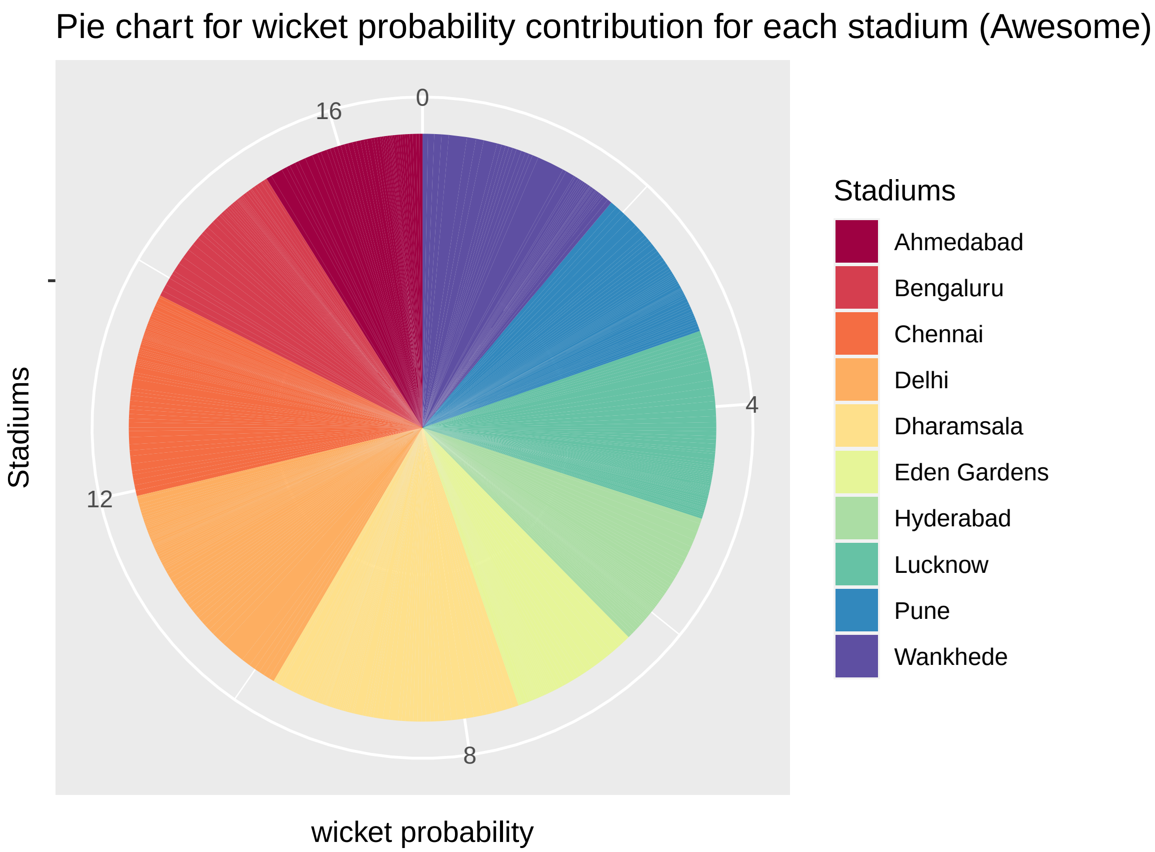
Stadiums with highest run and wicket probability contributions (Awesome visualizations)

As for design choice, pie chart would show the run contribution proportions well visually better than other graphs for each stadium.



This graph makes fully sense as Delhi and Dharmasala have highest scores in all matches on average.

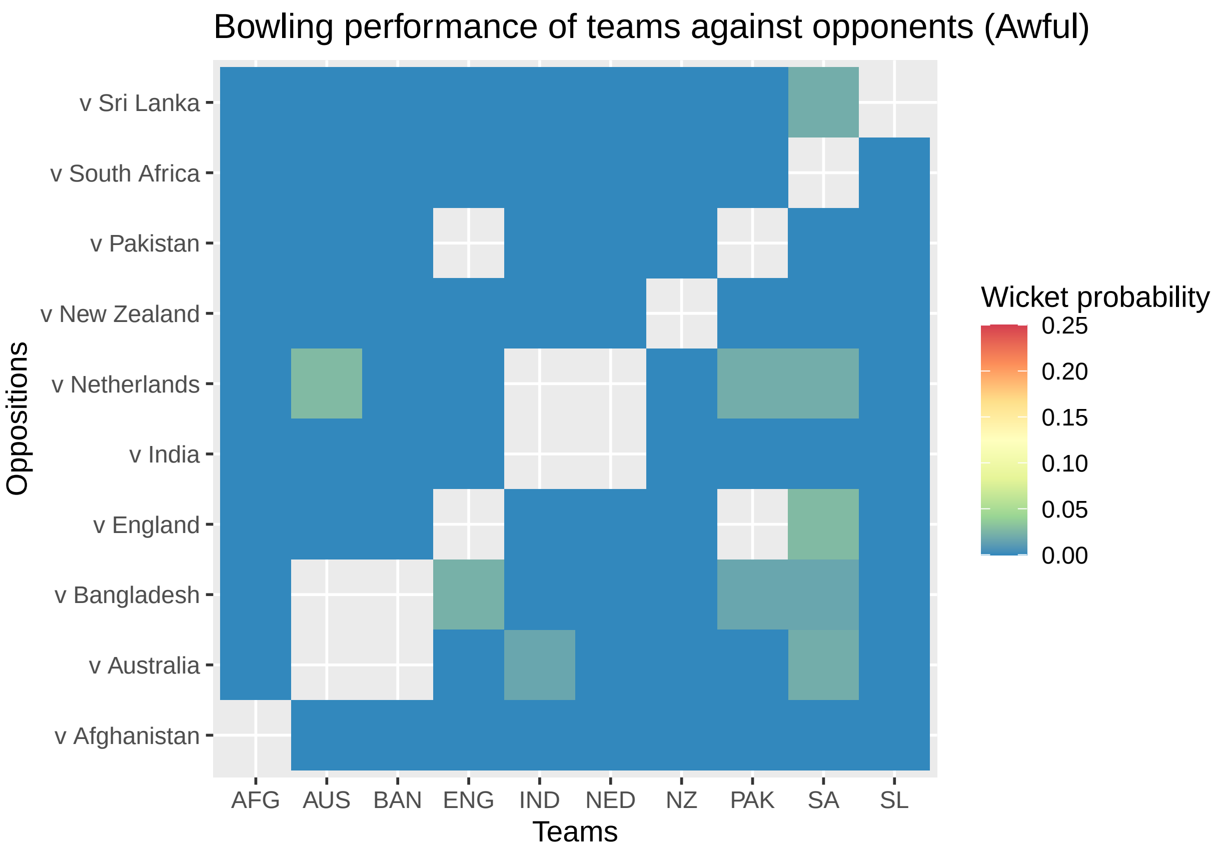
As for design choice, pie chart would show the bowling contribution proportions well visually better than other graphs for each stadium.

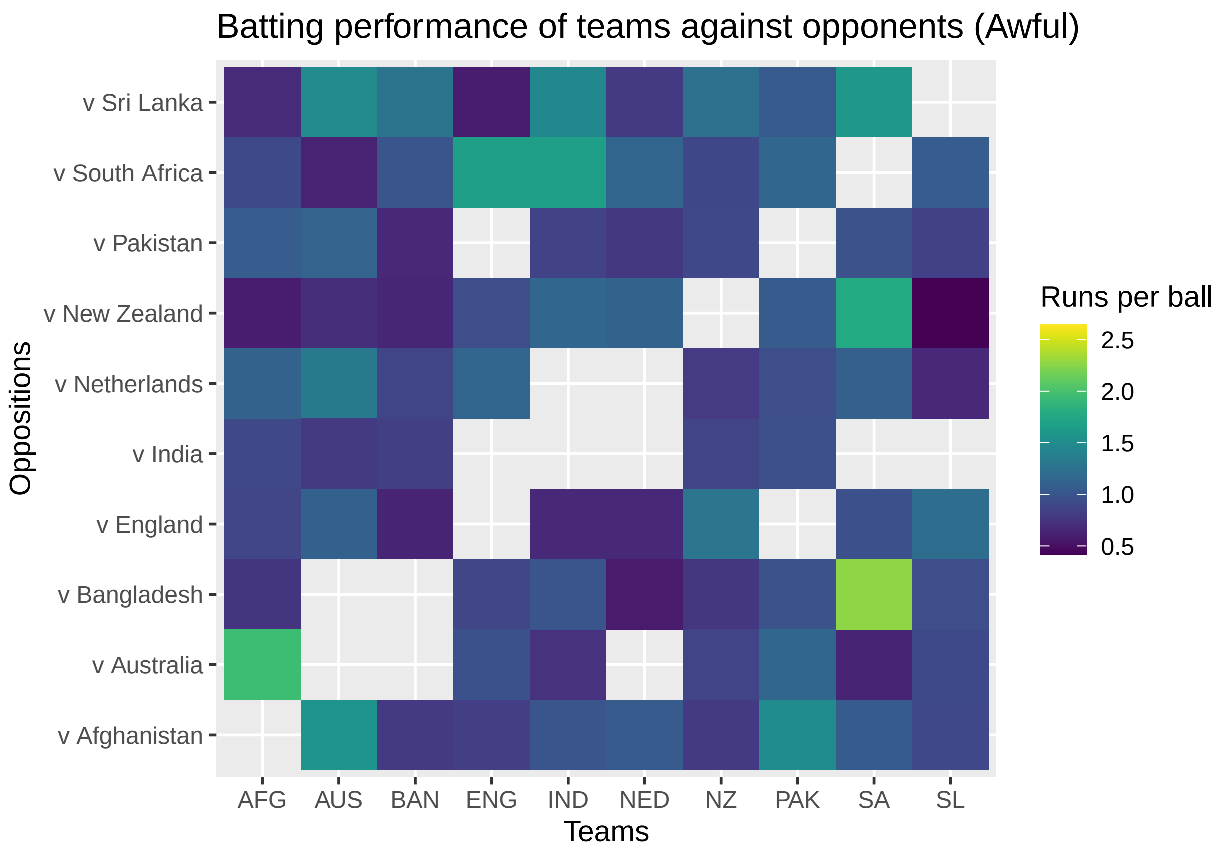


This graph makes fully sense as Delhi, Chennai and Dharmasala have had highest wicket probabilities in all matches on average.

Teams' bowling and batting performances with each other (Awful visualizations)

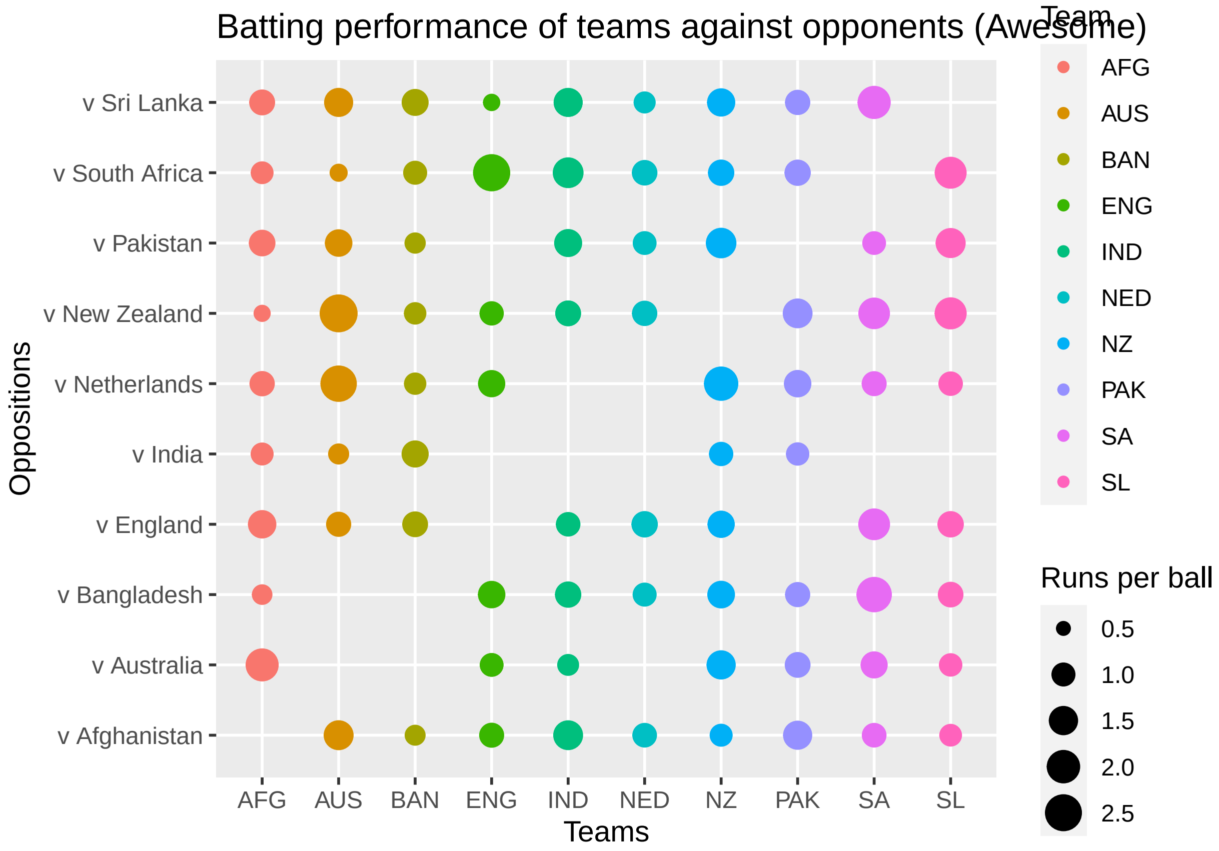
As for design choice, heat maps are worst at when it comes to understanding the right values from the users point of view in my opinion, as it takes some time to read.

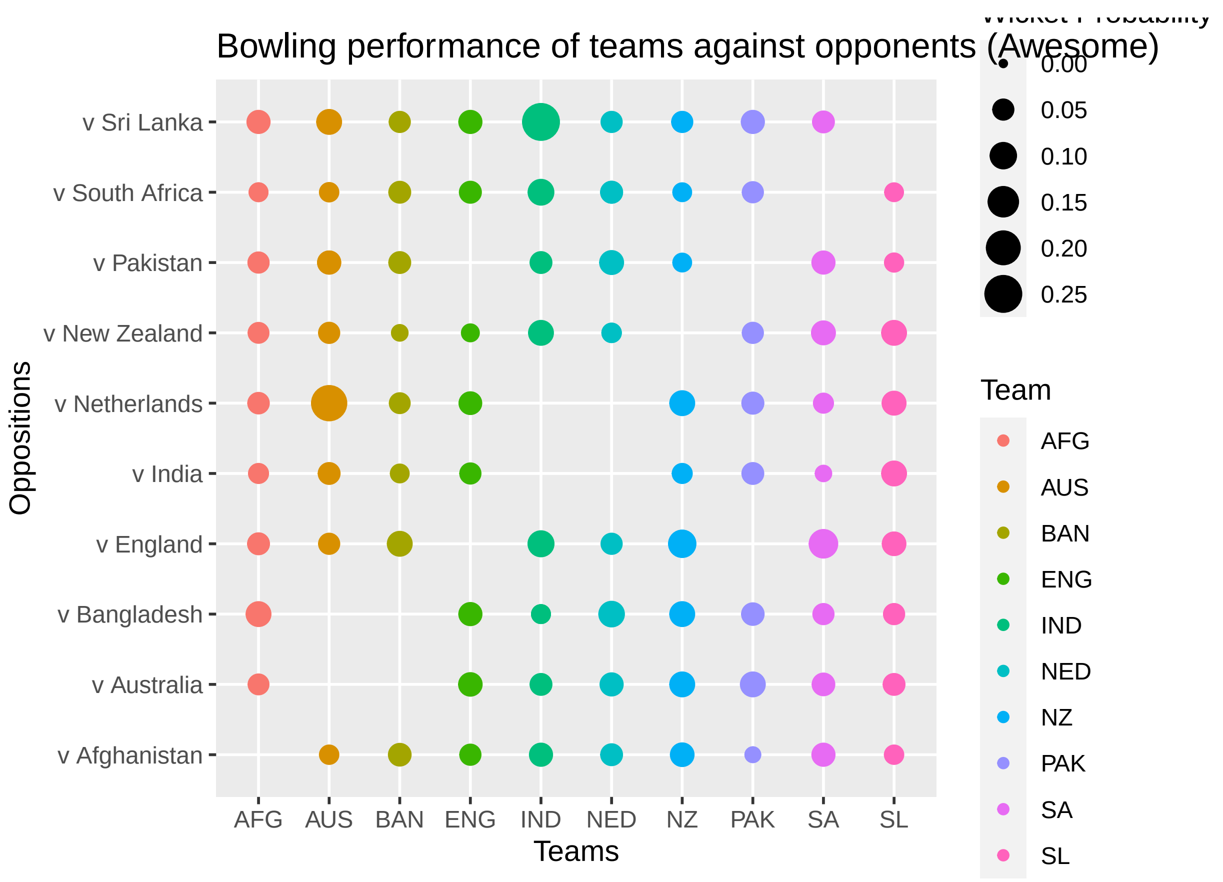

Bowling performance of teams against opponents (Awful)
Oppositions
v Sri Lanka
v South Africa
v Pakistan
v New Zealand
v Netherlands
v India
v England
v Bangladesh
v Australia
v Afghanistan
AFG
AUS
BAN
ENG
IND
NED
NZ
PAK
SA
SL
Teams
Wicket probability
0.25
0.20
0.15
0.10
0.05
0.00



But still, both of these graph makes sense to an extent when it comes to realizing that a team cannot play with themselves, so that is why there are some missing colors diagonally. Also some teams did not get the chance to play with each other because of the format of the fixtures.

So after looking at above graphs I gave my best try to make **a better graph** out of this. Then I ended up with a scatter plot that looked like this.


Batting performance of teams against opponents (Awesome)
Team
AFG
AUS
BAN
ENG
IND
NED
NZ
PAK
SA
SL
Runs per ball
0.5
1.0
1.5
2.0
2.5
Oppositions
v Sri Lanka
v South Africa
v Pakistan
v New Zealand
v Netherlands
v India
v England
v Bangladesh
v Australia
v Afghanistan
AFG
AUS
BAN
ENG
IND
NED
NZ
PAK
SA
SL
Teams



Now they make much more sense. as we can clearly visualize teams and their oppositions with for wicket taking probability or runs per bowl rate.

Conclusion

After cleaning up the data set, and visualizing the data as shown above I have found some specific and useful information related to CWC23.

Australia had the **best batting performance** in the world cup, also South Africa comes second. **Australia** had the best batting performances **against Netherlands and New Zealand**. Meanwhile **India** had the **best bowling performance**, then Australia comes after that. **India** had the best bowling performance **against Sri Lanka**.

Best stadiums **for bowling is Dharamsala** stadium and **for batting is Delhi stadium**. Also **Glenn Maxwell from Australia** was the **best batsman** in terms of runs per ball while **Dilshan Madhushanka from Sri Lanka** was the **best bowler** in terms of wicket taking probability.

Citation for the data set

Kaggle data set - *https://www.kaggle.com/datasets/jdaustralia/icc-cwc23-all-innings-cleaned*

ESPN Cric Info Website - *https://www.espncricinfo.com/records/tournament/icc-cricket-world-cup-2023-24-15338*