E0 259 : Data Analytics

Project Report : ODI LIVE WIN PREDICTOR

TEAM NAME : Fortune Tellers **TEAM NUMBER: 2**

TEAM MEMBERS

1. **Rahul Ranjan**
2. **M.V.S. Ajay**
3. **Dhaval Chavda**
4. **Smit Radadiya**

Introduction :

The problem is to predict the live win probability of cricket teams during One Day International (ODI) matches. The objective is to develop a model that calculates the win probability for each team dynamically, ball by ball, throughout the course of the game. This involves analyzing a comprehensive dataset containing historical ODI match data, including team performance, match conditions and outcomes. The challenge lies in designing a predictive model that considers the evolving nature of a cricket match, accounting for changing game dynamics. The project aims to provide valuable insights for cricket enthusiasts, analysts, and teams by enhancing the understanding of live win probability trends in ODIs.

Description about the dataset :

We took men's one day data from cricsheet.org. The raw data was a .json file, for each match. The json file contained metadata about the match like stadium, city, players, umpires etc and ball by ball data[batter, non-striker, bowler, runs-by-bat, wides, no-balls, penalty, kind-of-wicket, dismissed-player etc] for both innings.

We processed this json into csv formatted similar to the duckworth-lewis dataset of the first assignment. We made one csv by combining all the matches. We also removed matches which were incomplete or had no result. The final processed data has columns :

['match_id', 'innings', 'over_and_ball', 'batting_team_name', 'batsman', 'non_striker', 'bowler', 'runs_off_bat', 'extras', 'wides', 'no_balls', 'byes', 'leg_byes', 'penalty', 'kind_of_wicket', 'dismissed_player', 'valid_ball', 'wicket_ball', 'runs_on_this_ball', 'overs', 'date', 'toss_winner', 'match_winner', 'toss_winner_chose_bat_first', 'match_country', 'runs_scored', 'wickets_fallen', 'balls_bowled', 'req_run_rate', 'target_score', 'innings_total_runs', 'fielding_team', 'runs_remaining', 'wickets_in_hand', 'run_rate']

Extracted features.

1.) Head to head match results.
2.) Last 5 matches results of batting team.
3.) Last 5 matches results of bowling team.

List of methods tried:

1. Duckworth lewis(DLS) par score
2. Basic logistic regression
3. Dynamic logistic regression model
4. XG-Boost

Design Choices

In a match, there is a significant difference between the first innings and second innings such as the addition of features such as ‘required_run_rate’, ‘target_score’. This motivated us to model first and second innings separately. We also removed appropriate features while feeding into the model for first and second innings models such as ‘innings_total_runs’ etc, basically, any features which are not available in a live match.

Feature Engineering

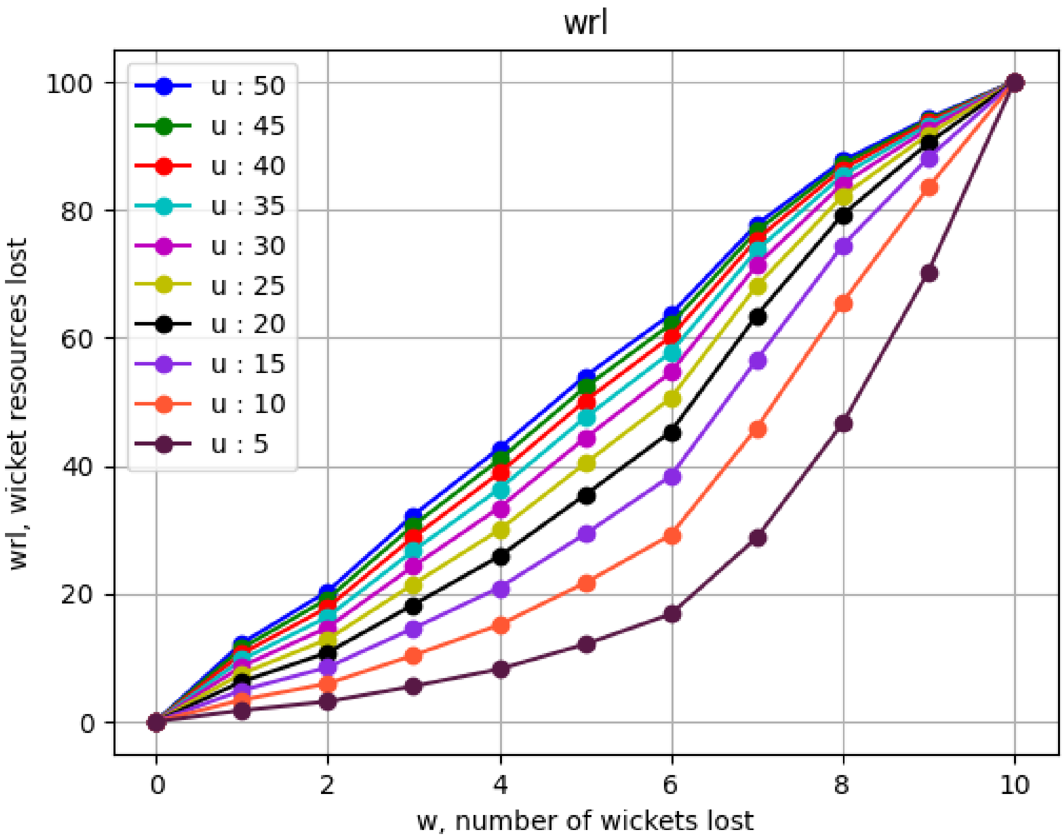
We also calculated new features to squeeze out performance from the models.

1. ***Wicket Resource lost (wrl)*** **:**

   In an innings, wickets fallen at the **early overs** are **more valuable** than the ones at the **death overs**, i.e. wickets have different value for different value of the balls or over remaining. Also the wickets of top order are more valuable than the bottom order. To represent this, we made a new feature, ‘wrl’ which tries to model this fact.

   $$wrl = \frac{Z(u, 0) - Z(u, w)}{Z(u, 0)}$$

   where Z(u,w) denotes the expected runs with ‘u’ overs remaining and ‘w’ wickets lost. The graph below is drawn using the coefficients obtained from the first assignment.



As one can observe, wrl is linear when number of overs to go is large but flattens(i.e not much resources lost) at the death overs for top order, but steeply increases for the bottom order at death overs.

2. *Form difference for the teams:*

We calculate a team's current form as a weighted mean of match outcomes over their last five games. Specifically, let y_t = 1 if a team won the match played t matches ago, and 0 otherwise. We then define the team current form as

$$form = \frac{\sum_{t=1}^{5} w(t,\theta)y_t}{\sum_{t=1}^{5} w(t,\theta)}, \text{ where } w(t,\theta) = (1 - \theta)^{t-1} \text{ and } 0 < \theta < 1.$$

A team will have a form of 1 if they have won their most recent five matches and form of 0 if none of the matches were won. The function $w(t, \theta)$ is a discounting factor, so that the **most recent match** receives the **highest weight.** This implies that for a given θ, two teams with the same number of wins would have different values of the form, **depending** on the **order of their wins**.(We found θ to be 0.2041)

We take the form difference between the two teams as a feature.

3. ***Predicted remaining score:***

   Predicted remaining score is nothing but the estimated score in the remaining overs.Using knn neigbours, we are calculating this feature. Adding this feature to input feature to predict the probability.

Methods

DLS par score :

We are calculating the DLS par score for both innings and based on that we predict the win probability for the batting team for a particular inning.

1st inning par score:

- Par score = Z (u, w) = Z0 (w)[1 − exp{−Lu/Z0 (w)}]

For 2nd inning par score is calculated based on resource used and target set by the 1st inning.

- Par score = Resources used × target
- Resources used = $\frac{Z(u,w)}{Z(N,10)}$

  Where;
  u = overused
  w = wickets fallen
  N = 50

Using Par score and run scored finding probability of winning of batting team

- If both same then 50%-50% for both teams
- If Run scored > par scored
  - Probability = (Run scored - Par score)/Run scored
- Else
  - Probability = 1 - (Par scored - Run score)/Par scored

Basic Logistic Regression

We applied logistic regression on the appropriate features with the target variable as the Indicator random variable of batting team winning, for both innings.

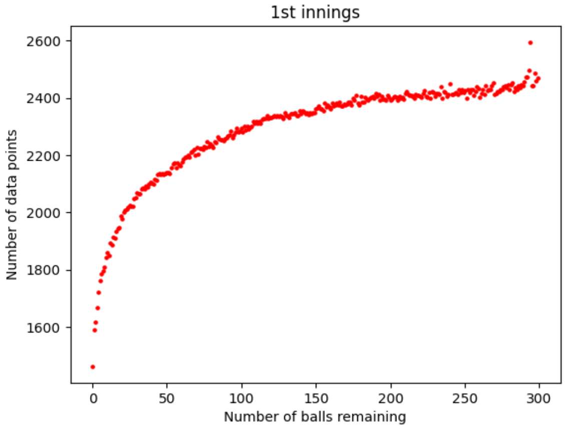
$f(x) = \frac{1}{1+e^{-x}}$ where $x = w^T x_d$ where x_d is the data point and w is the vector of coefficients.

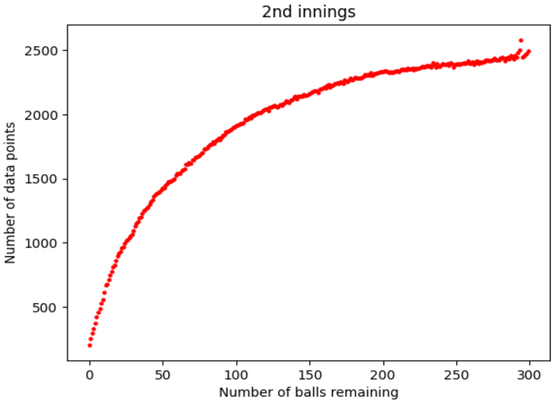
$f(x)$ gives us a value between 0 and 1, which can be interpreted as probability.

Dynamic Logistic Regression

At the start of an inning, we have resources, balls remaining, u = 300 and wickets left, w = 10. So, ‘u’ can range from 300 to 0, and ‘w’ from 10 to 0, in the span of the whole innings. There are 301 x 11 distinct combinations of ‘u’ and ‘w’. For each of these configurations, we filtered the data points and then fit a logistic regression model. By doing the above, we have 301 x 11 models for predicting the win probability at each possible state of the match.

For each u and w, we will have some number of data points, from the dataset above. We plotted the number of data points against the number of balls left.





For predicting at a particular point of time, we use ‘u’ and ‘w’ to get the corresponding model and then use that model to get the probability of the batting team winning.

XG - Boost :

Belonging to the ensemble learning family, XGBoost utilizes gradient boosting,It is extensively used in various real world cases.In this method we will train several weak models to reduce the bias.

As we discussed above, We trained two different model for both innings. We plot the feature importance graph for both innings.

Results :

For each approach we are using accuracy as metric.

Accuracy:

Accuracy is calculated by taking predicted as 1 if probability is greater than 50%, else 0, then the fraction of correct prediction by total predictions.

- DLS par score

First Innings	38.34%
Second Innings	43.23%

- Basic logistic regression

First Innings	74.57%
Second Innings	92.07%

- Dynamic logistic regression

First Innings	73.78%
Second Innings	85.6%

- Xg boost

First Innings	77.78%
Second Innings	92.43%

- Xg boost with engineered feature (fd , wrl, predicted_remaining_score)

First Innings	94.5%
Second Innings	97.7%

<u>CASE STUDY:</u>

Match : **India vs Australia (ICC ODI World cup 2023 league match)**

Match Summary:

1st Innings : AUS : 199 (49.3)

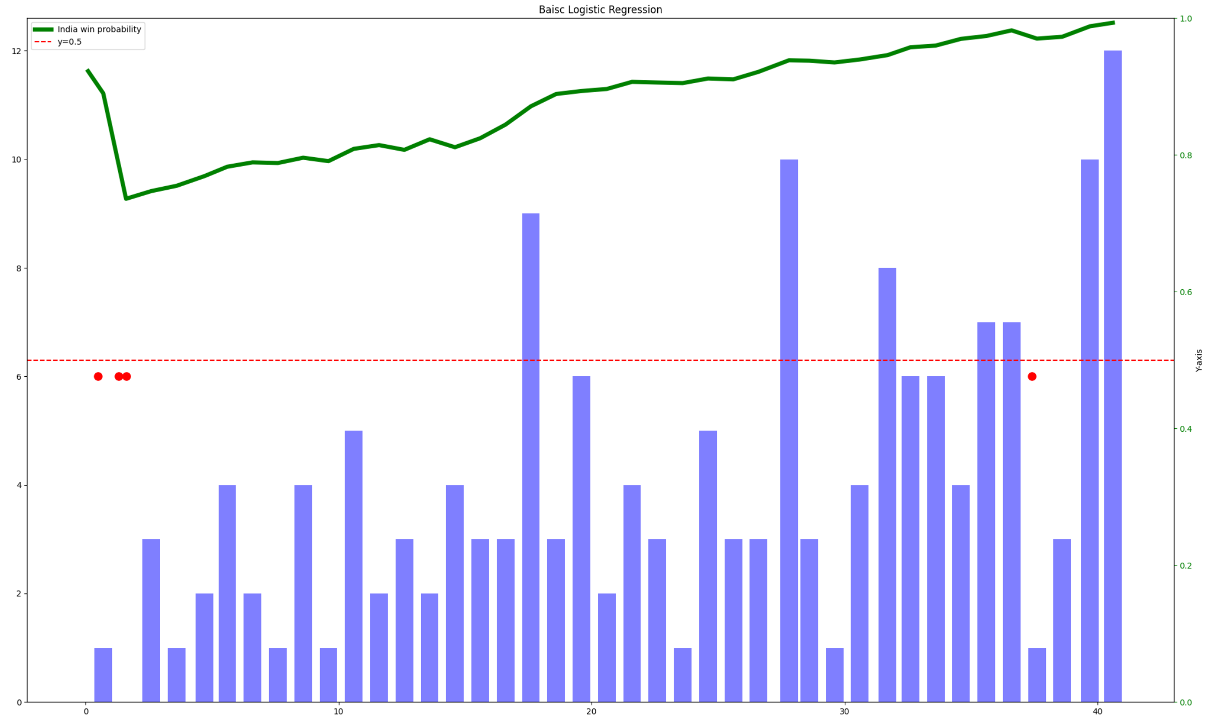
2nd Innings : IND : 201/4 (41.2)

Toss Winner	First Innings	Second Innings	Match Winner
AUSTRALIA	AUSTRALIA	INDIA	INDIA

For each of the methods, live win probability over the span of the 2nd innings [India Batting].

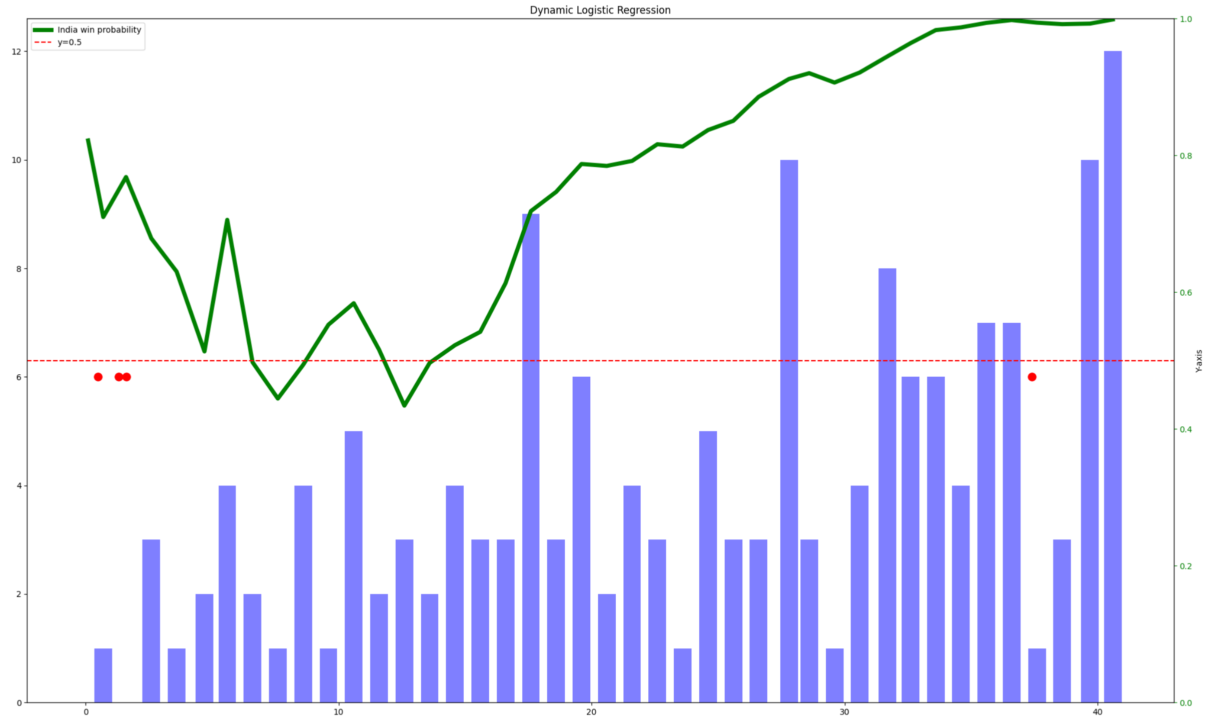
For 2nd Innings of India

- **Basic logistic regression**



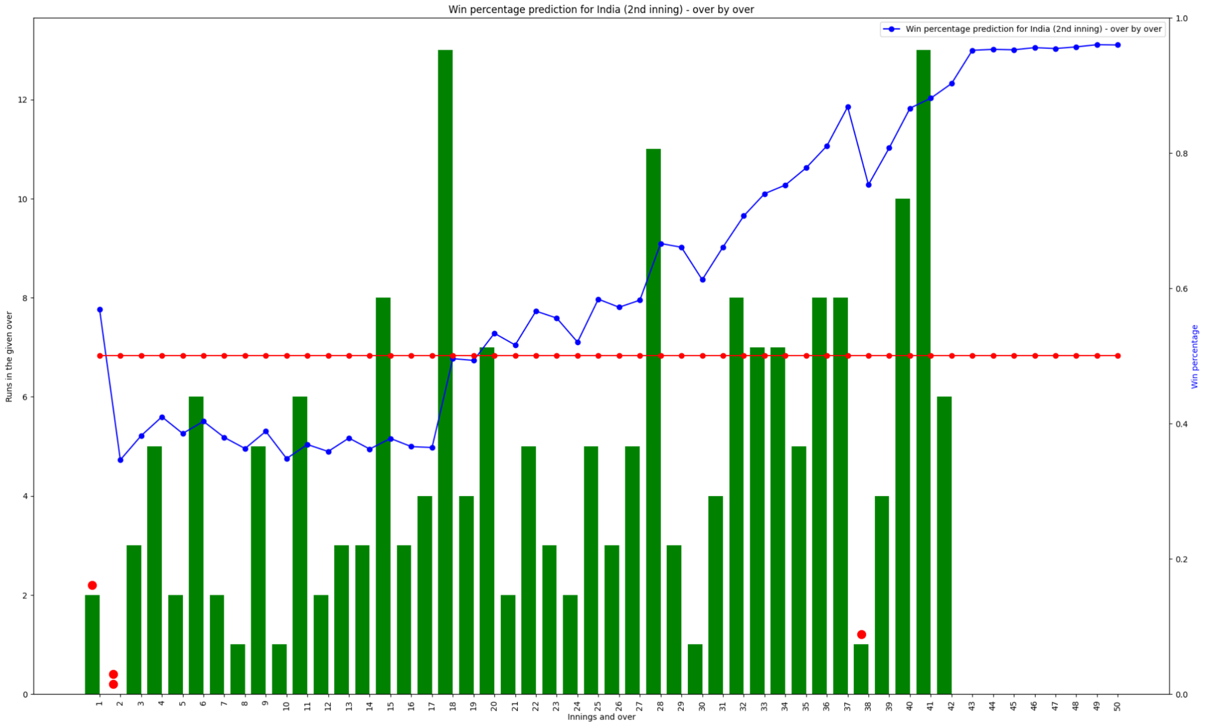
- Here, India lost one wicket in first over and 2 wickets with zero runs in 2nd over. So, probability fell sharply.
- It started with high probability because of less target.
- In 38th over, India lost a wicket, but down fall of probability is not that sharp, because there are 6 wickets left , 12 overs left and 33 more runs to chase.

- **Dynamic logistic regression**



  - Same comments as above method, But this method is sensitive to run rate.

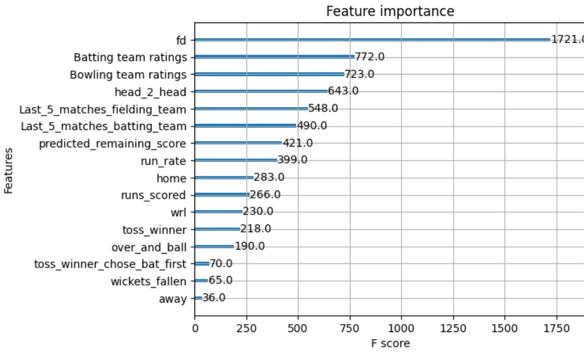
- **XG Boost :**



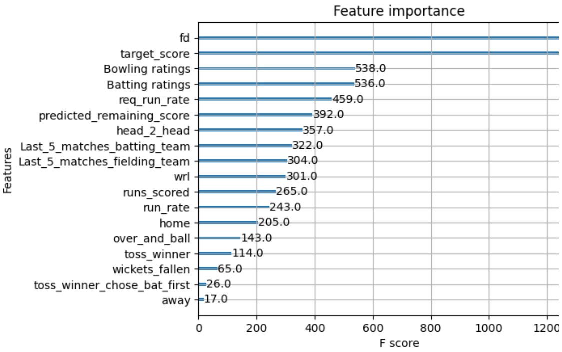
- Same comments as above methods, but this method is more sensitive to wickets , irrespective of runs to chase

Importance score

- **1st Innings**



innings 1



Innings 2

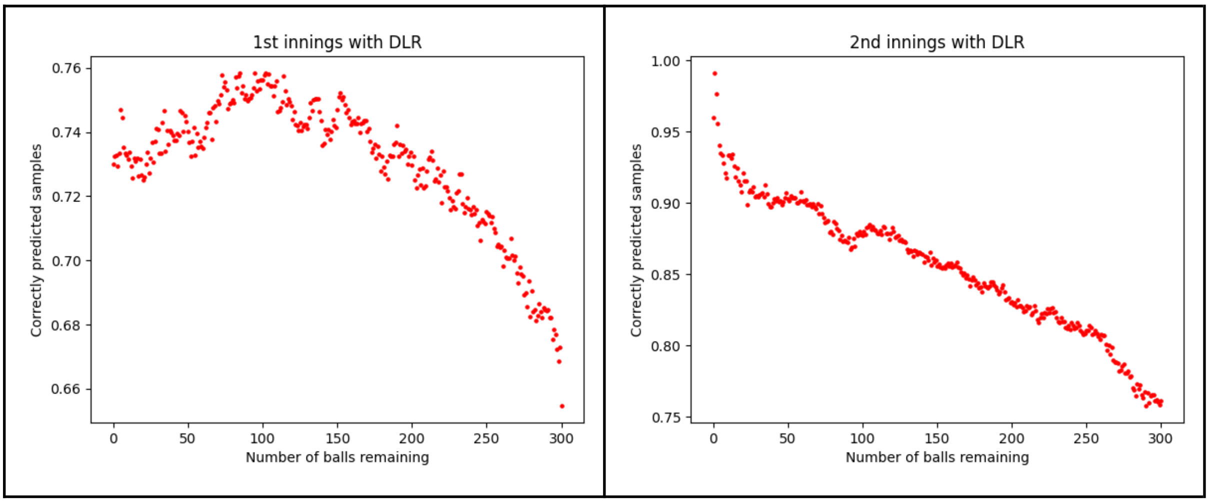
→ As we can observe the engineered form difference(fd) is statistically significant feature to give win probability.

→ Also it significantly increased the accuracy of the 1st innings model (by 15%).

→ We can see in the 1st innings , the difference between the 1st important feature and the 2nd important feature is significantly larger. That’s why this impacting 1st innings more.

Results for the dynamic logistic regression :

- Graph of balls remaining vs percentage of correctly predicted samples.



As we know cricket is a game of uncertainties. This is also evidenced by the above graph as if the balls remaining are large, the model has the lowest fraction of correctly predicted samples and the fraction increases with decreasing balls remaining for the innings.

Conclusion

The best model came out to be XGboost with feature engineering and the worst was the DLS par score method. The three models namely basic logistic regression, dynamic logistic regression and XG boost depend on quality of the features, hence, depend upon good feature engineering. For the model XG boost we used some features like last 5 matches of batting and fielding team, head to head matches, is batting team which are giving some less then one probability for winning team at last 50th over where team was already won the match.

From the importance score, we can observe that features of wicket resource lost(wrl), form difference(fd) and predicted score significantly improved the results of XG boost model. Form difference played a key role in 1st innings model.

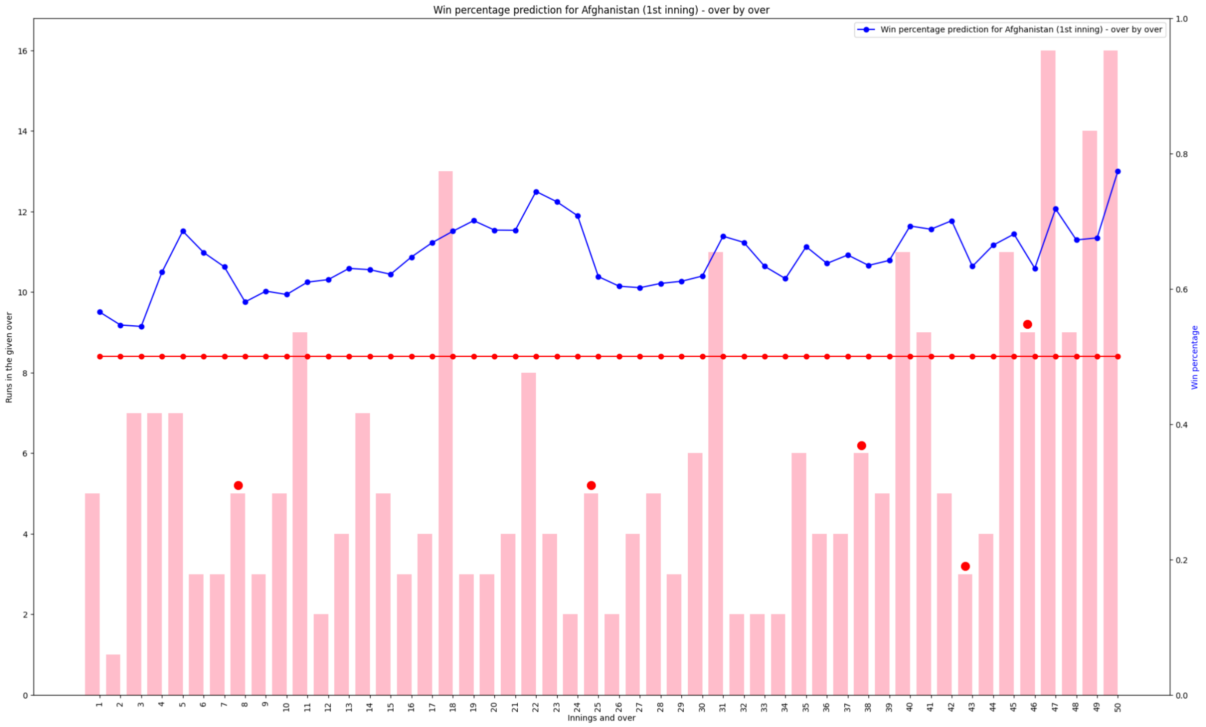
Looking ahead, there is recognition of the potential for further improvement. Additional features, such as partnership dynamics, pitch conditions, current ICC rankings of players, and historical data on interactions between bowlers and batsmen, were not incorporated in this iteration but are acknowledged as potential contributors to enhanced predictive accuracy. Future iterations of the model could explore the inclusion of these features for more comprehensive and precise win probability predictions.

Appendix

Match example: Australia vs Afghanistan (ICC ODI World cup 2023 league match)

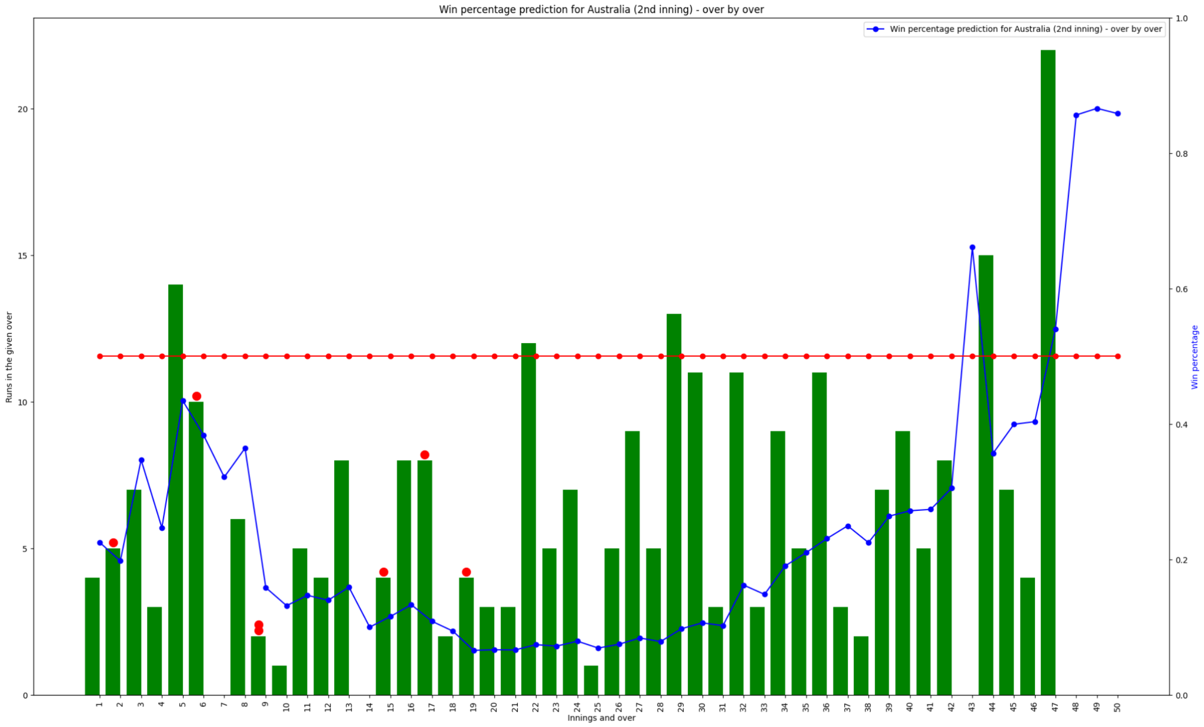
- Match Data :
  - Toss winner : Afghanistan
  - First inning : Afghanistan
  - Second Inning : Australia
  - Match winner : Australia

Win probability of Afghanistan :



- Here, We can see that whenever less runs came in over probability of afghanistan decreased little bit and when runs were more, then the probability increased.

Win Probability of Australia :

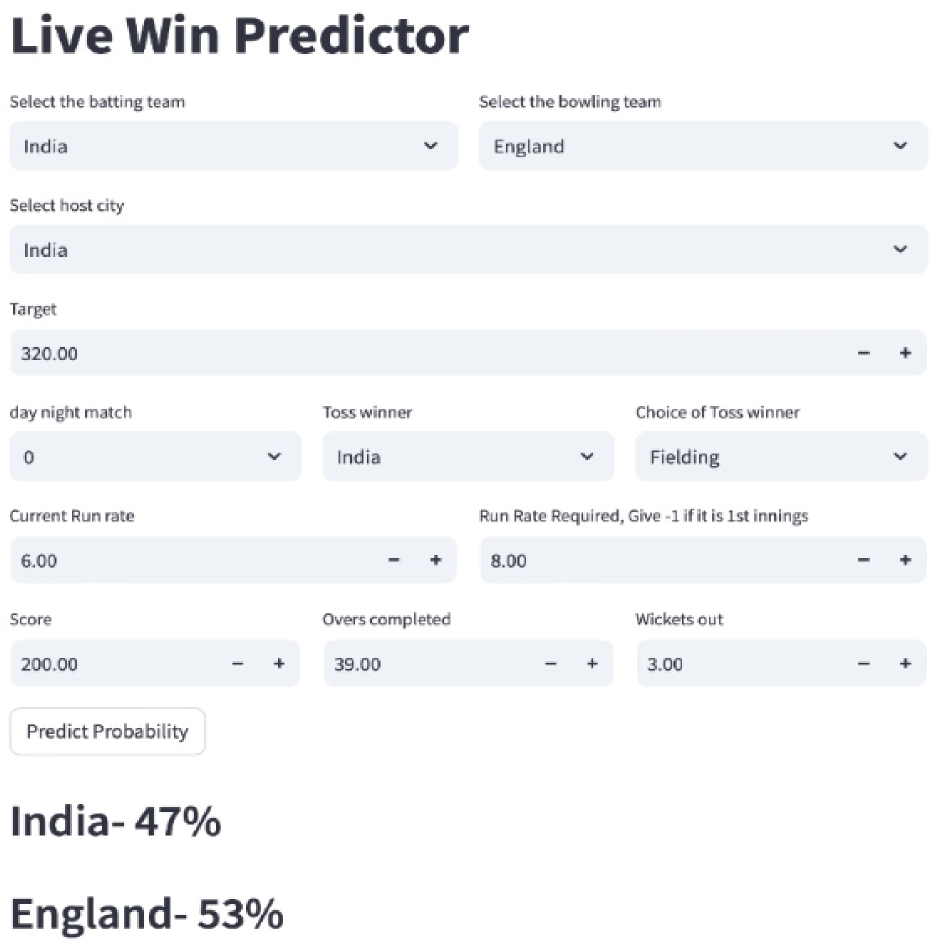


- Here, Australia’s win probability came to around 18% in 19th over because they lost 7 wickets and more score to be chased.

- But, after 19th over probability started increasing slowly as run rate became stable.

- The point to be noted here is until 42nd over , the probability of Australia winning is less than 0.5.

Glimpse of front end tool:



Live Win Predictor

Select the batting team: India

Select the bowling team: England

Select host city: India

Target: 320.00

day night match: 0

Toss winner: India

Choice of Toss winner: Fielding

Current Run rate: 6.00

Run Rate Required, Give -1 if it is 1st innings: 8.00

Score: 200.00

Overs completed: 39.00

Wickets out: 3.00

India- 47%

England- 53%