Group ID: 24

Student Information:

Name	Enrollment Number	Semester	Batch
Panchal Nemil Ramanbhai	220160116048	5th	IT/A3
Pandey Shubham Tilakram	220160116050	5th	IT/A3

Indian Cricket team(BCCI)

- Analysis of Indian player performance.
- Analysis of Indian Cricket team.
- Match outcome prediction.
- World Cup description prediction.
- Conclusion.

Introduction

- Briefly describe the relevance of analyzing the performance of Indian players and predicting match outcomes, especially in the context of the Cricket World Cup.
- In a country where cricket is akin to religion, accurate predictions on player performance and match outcomes can drive strategic decisions, enhance fan engagement, and offer new perspectives on the team's potential success in the World Cup.

Objective

- **Objective 1:** Analyze the historical performance of top Indian cricketers over various international tournaments and conditions (home, away, neutral venues).

- **Objective 2:** Develop a predictive model to forecast match outcomes, focusing on key performance indicators such as batting averages, bowling economy, and fielding efficiency.

- **Objective 3:** Generate World Cup outcome predictions using predictive algorithms, evaluating potential future scenarios for the Indian cricket team.

Key Questions

- **Player Performance:** Which players consistently outperform in matches?
- **Match Outcome Predictors:** What are the most significant predictors of match outcomes for the Indian cricket team? Is it dependent on individual brilliance or team synergy?
- **World Cup Trends:** Based on historical data, how do different factors like pitch type, weather conditions, and opposition impact India's World Cup performance?

Data Collection and Preparation

- **Data Sources:**
  - ESPN Cricinfo and other cricket statistics databases for player and match data.
  - Historical match outcomes, weather data, and team compositions.
- **Prediction:**
  - **Step 1:** Clean data to remove anomalies (e.g., incomplete scorecards, missing player stats).
  - **Step 2:** Create new features, such as player form metrics (moving averages) and match difficulty scores.
  - **Step 3:** Normalization and handling of categorical data, like match venues and opposition teams.

Data Analysis

- **<u>Analysis Techniques:</u>**
  - **Descriptive Statistics:** Use historical averages, strike rates, and other key indicators to establish a baseline of player performance.
  - **Correlation Analysis:** Identify which factors (e.g., toss outcomes, batting order) most strongly correlate with match outcomes.
  - **Player Consistency Analysis (Moving Average):**The moving average helps in tracking a player's consistency over time by smoothing out short-term fluctuations in performance.
  - **Time Series Forecasting:**Time series forecasting is used to predict future values in a dataset based on historical data.
  - **Visualization:** Create visualizations such as bar-chart,histograms.

Modeling Approach

- **Model Selection:**
  - **Random Forest:** For predicting match outcomes based on multiple features.
  - **Time Series Analysis:** To forecast player performance trends
- **Model Training:**
  - Split data into training and test sets (e.g., 80% train, 20% test).
  - Train the models using appropriate algorithms and evaluate performance using metrics like accuracy, precision, and recall.

MODEL : ***RANDOM FOREST***

How does it work : Random Forest is an ensemble learning method that builds multiple decision trees using random subsets of the data and features. It combines the predictions of these trees through majority voting for classification tasks, enhancing accuracy and reducing overfitting compared to individual trees.

```
# Data Collection
# Sample dataset creation
data = {
    'Match_ID': [1, 2, 3, 4, 5, 6, 7, 8, 9],
    'Venue': ['India', 'Australia', 'England', 'New Zealand', 'South Africa', 'Sri Lanka', 'UAE', 'Bangladesh', 'West Indies'],
    'India_Score': [300, 250, 280, 320, 290, 270, 310, 240, 260],
    'Opposition_Score': [290, 260, 300, 310, 280, 230, 300, 250, 270],
    'Toss_Winner': ['India', 'Australia', 'India', 'New Zealand', 'South Africa', 'Sri Lanka', 'UAE', 'Bangladesh', 'West Indies'],
    'Outcome': ['India', 'Australia', 'India', 'New Zealand', 'India', 'Sri Lanka', 'India', 'Bangladesh', 'West Indies']  # Match outcome
}

# Create DataFrame
df = pd.DataFrame(data)
df['Outcome'] = df['Outcome'].map({'India': 1, 'Australia': 0, 'New Zealand': 0, 'Sri Lanka': 0, 'Bangladesh': 0, 'West Indies': 0})  # Map outcomes to 1 and 0

# Data Preprocessing
# Convert categorical variables to numerical format using one-hot encoding
df = pd.get_dummies(df, columns=['Venue', 'Toss_Winner'], drop_first=True)
```

```python
# Features and target variable
X = df.drop(columns=['Match_ID', 'Outcome'])
y = df['Outcome']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Initialize and Train the Random Forest Model
rf_model = RandomForestClassifier(n_estimators=100, random_state=42)
rf_model.fit(X_train, y_train)

# Make Predictions
y_pred = rf_model.predict(X_test)

# Evaluate the Model
accuracy = accuracy_score(y_test, y_pred)
print('Accuracy:', accuracy)
class_report = classification_report(y_test, y_pred)
print(class_report)

# Feature Importance
importances = rf_model.feature_importances_
feature_importance_df = pd.DataFrame({'Feature': X.columns, 'Importance': importances})
feature_importance_df = feature_importance_df.sort_values(by='Importance', ascending=False)

# Plot feature importances
plt.figure(figsize=(10, 5))
plt.barh(feature_importance_df['Feature'], feature_importance_df['Importance'])
plt.title('Feature Importance')
plt.xlabel('Importance')
plt.ylabel('Features')
plt.show()
```

OUTPUT

```
Accuracy: 1.0
              precision    recall  f1-score   support

           0       1.00      1.00      1.00         2

    accuracy                           1.00         2
   macro avg       1.00      1.00      1.00         2
weighted avg       1.00      1.00      1.00         2
```

Step To Predict Outcomes For New Matches

```python
# Step to predict outcomes for new matches
new_matches = {
    'Venue': ['India', 'Australia', 'England'],  # List of venues
    'India_Score': [310, 290, 280],              # India scores
    'Opposition_Score': [295, 280, 300],         # Opposition scores
    'Toss_Winner': ['India', 'Australia', 'India'],  # Toss winners
}

new_matches_df = pd.DataFrame(new_matches)

# Preprocess the new match data
new_matches_df = pd.get_dummies(new_matches_df, columns=['Venue', 'Toss_Winner'], drop_first=True)

# Ensure the same feature columns as the training data
# Reindex using the feature columns from the training data (X)
new_matches_df = new_matches_df.reindex(columns=X.columns, fill_value=0)

# Make predictions
predictions = rf_model.predict(new_matches_df)

# Add predictions to the DataFrame
new_matches_df['Predicted_Outcome'] = predictions

# Map the outcome to readable format
new_matches_df['Predicted_Outcome'] = new_matches_df['Predicted_Outcome'].map({1: 'India Wins', 0: 'Opponent Wins'})

# Print the results without the 'Venue' column
print(new_matches_df[['India_Score', 'Opposition_Score', 'Predicted_Outcome']])

# Include the actual venue based on one-hot encoding
for index, row in new_matches_df.iterrows():
    venue = 'India' if row.get('Venue_India', 0) == 1 else (
            'Australia' if row.get('Venue_Australia', 0) == 1 else (
            'England' if row.get('Venue_England', 0) == 1 else 'Unknown'))
    print(f"Match Venue: {venue}, India Score: {row['India_Score']}, Opposition Score: {row['Opposition_Score']}, Predicted Outcome: {row['Predicted_Outcome']}")
```

Output:

```
   India_Score  Opposition_Score Predicted_Outcome
0          310               295        India Wins
1          290               280        India Wins
2          280               300        India Wins
Match Venue: India, India Score: 310, Opposition Score: 295, Predicted Outcome: India Wins
Match Venue: Unknown, India Score: 290, Opposition Score: 280, Predicted Outcome: India Wins
Match Venue: England, India Score: 280, Opposition Score: 300, Predicted Outcome: India Wins
```

- **Key points to note:**
  - Outcomes are mapped to numerical values: India=1, Opponent=0.
  - One-Hot Encoding: Converts categorical variables (Venue, Toss_Winner) into numerical format for model compatibility.
  - Features and Target Variable: Features (X): All columns except Match_ID and Outcome. Target (y): Outcome of the match.
  - Train and Test: The dataset is divided into training and testing sets (80/20 split).
  - Random Forest Classifier: Initializes the model with 100 decision trees.Fits the model using the training data.
  - Predictions: Predictions are made on the test set.
  - Performance Metrics: Accuracy of the model is calculated. Classification report includes precision, recall, and F1-score.

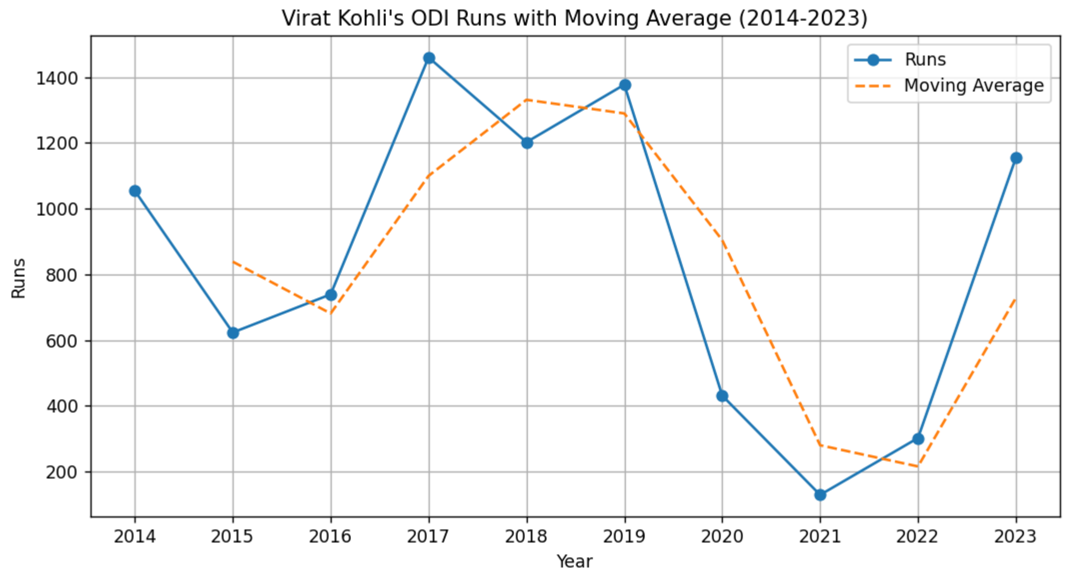
TIME SERIES ANALYSIS

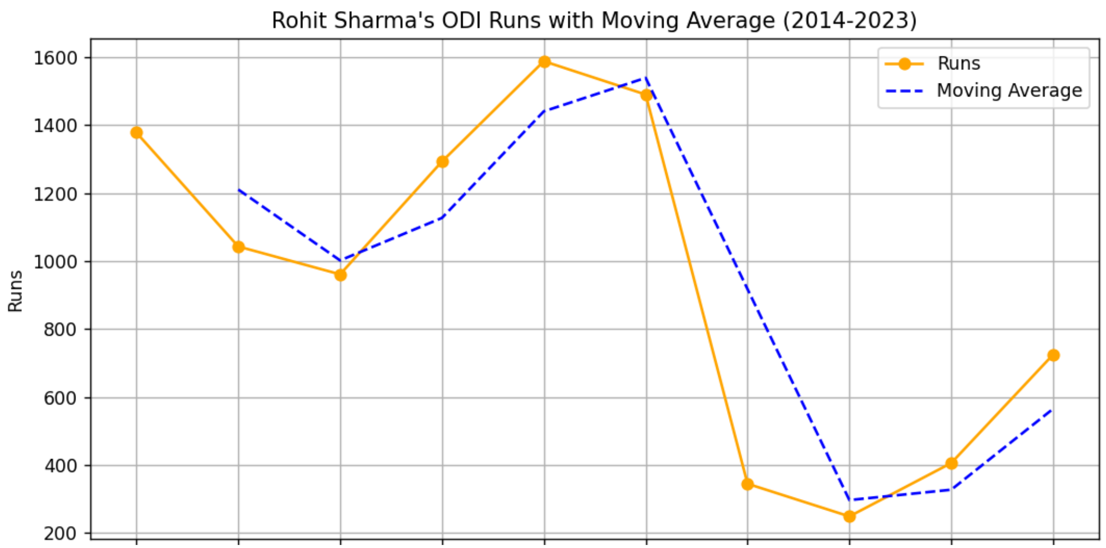
Why Forcasting? :

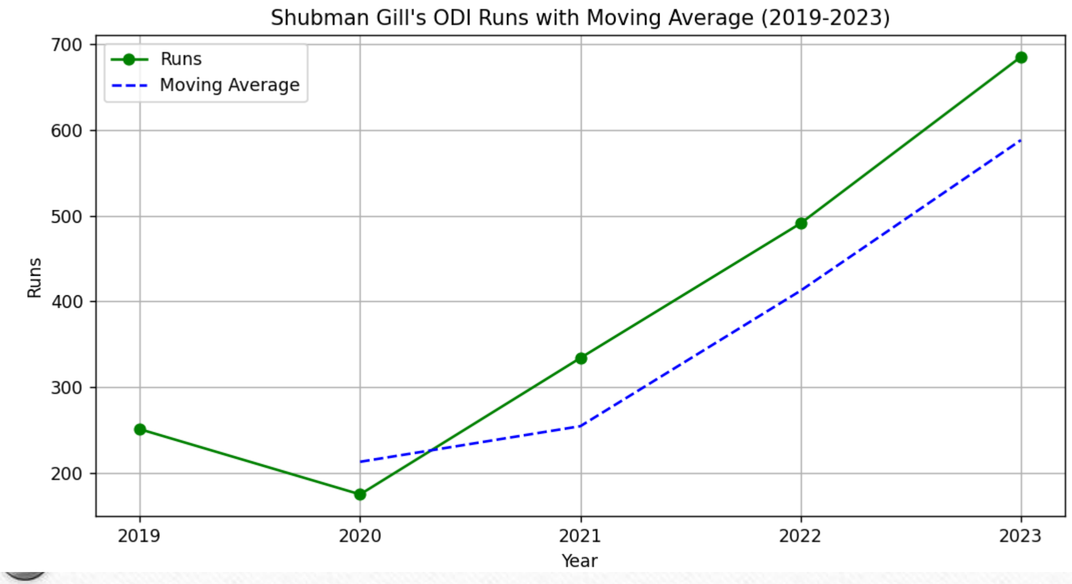
1. Forecasting in cricket helps teams and analysts predict player performance, match outcomes, and tactical decisions based on historical data and trends. It enables better strategic planning, resource allocation, and risk management to enhance competitive advantage in all formats of the game.

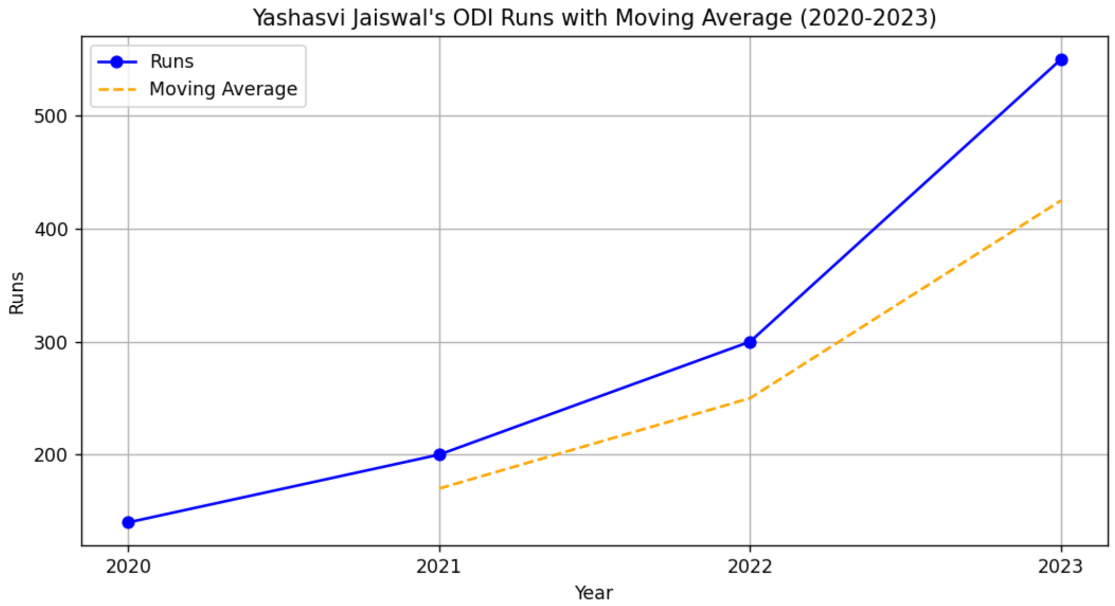
2. Measurements are made on at the regular time intervals.

3. Helps to pickup the in-form player for particular matches.


Virat Kohli's ODI Runs with Moving Average (2014-2023)
Runs
Moving Average
1400
1200
1000
800
600
400
200
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
Year
Runs


Rohit Sharma's ODI Runs with Moving Average (2014-2023)
Runs
Moving Average
1600
1400
1200
1000
800
600
400
200
Runs

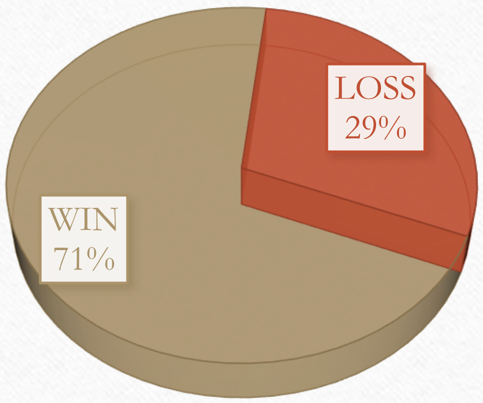

Shubman Gill's ODI Runs with Moving Average (2019-2023)
Runs
Moving Average
700
600
500
400
300
200
2019
2020
2021
2022
2023
Year
Runs


Yashasvi Jaiswal's ODI Runs with Moving Average (2020-2023)
Runs
Moving Average
500
400
300
200
2020
2021
2022
2023
Year
Runs

Results and Visualization

- <u>**Key Findings:**</u>
  - **Performance Insights:** Present findings on key players and their performances in critical matches.
  - **Match Outcome Predictions:** Showcase prediction results with accuracy metrics.
- <u>**Visualizations:**</u>
  - **Player Performance Heatmaps:** Visualize performance metrics across different conditions.
  - **Win Probability Graphs:** Show win probabilities based on various conditions (e.g., player form, opponent strength).
  - **Bar Charts:** Compare the performance of top players against competitors.

INDIAS WIN RATE IN ODI WORLD CUPS FROM 1992-2023 AGAINST OTHER COUNTRY

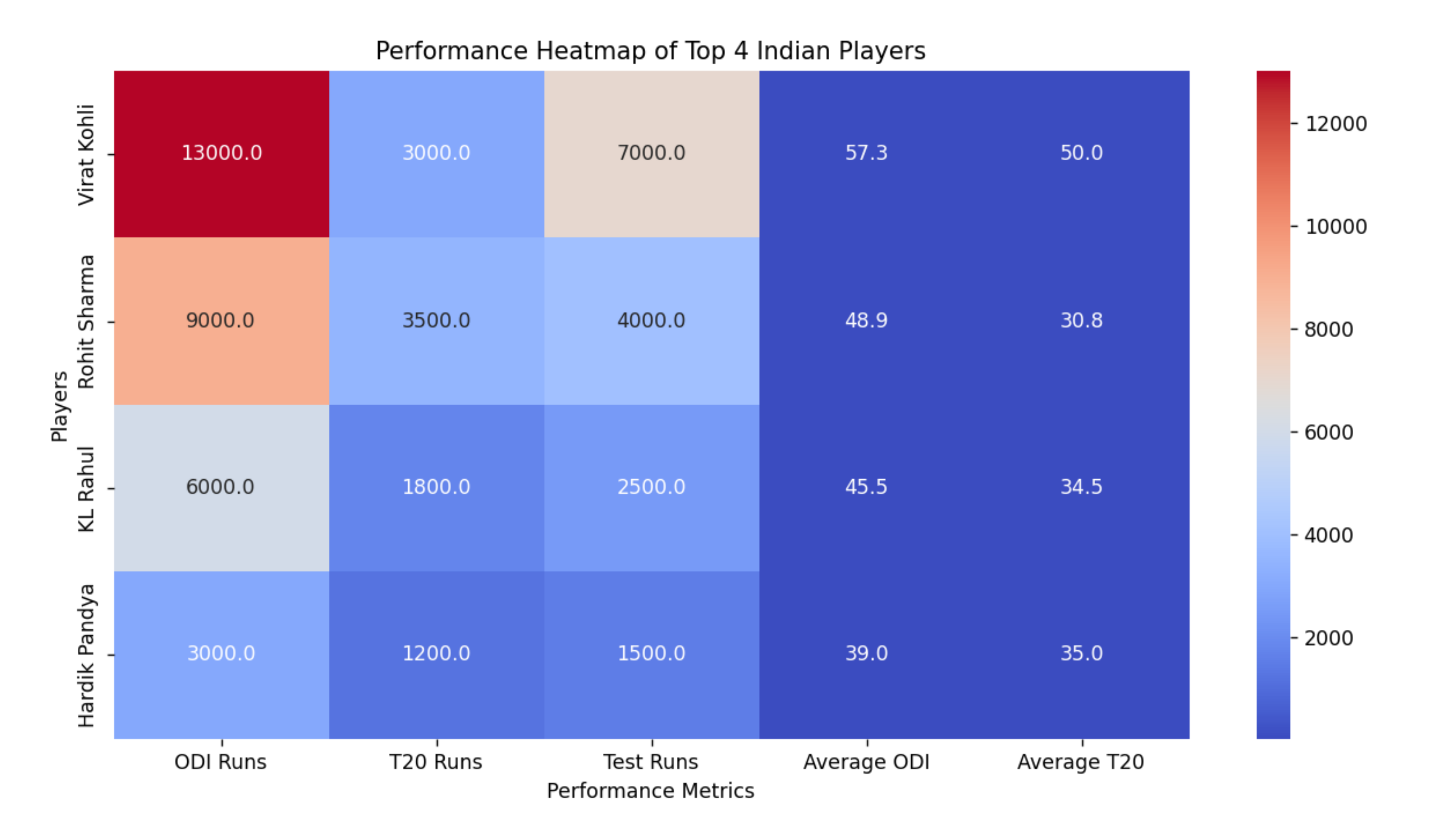


How does the toss affect the Indian cricket team in ODIs world cup(1992-2023) if we win toss:

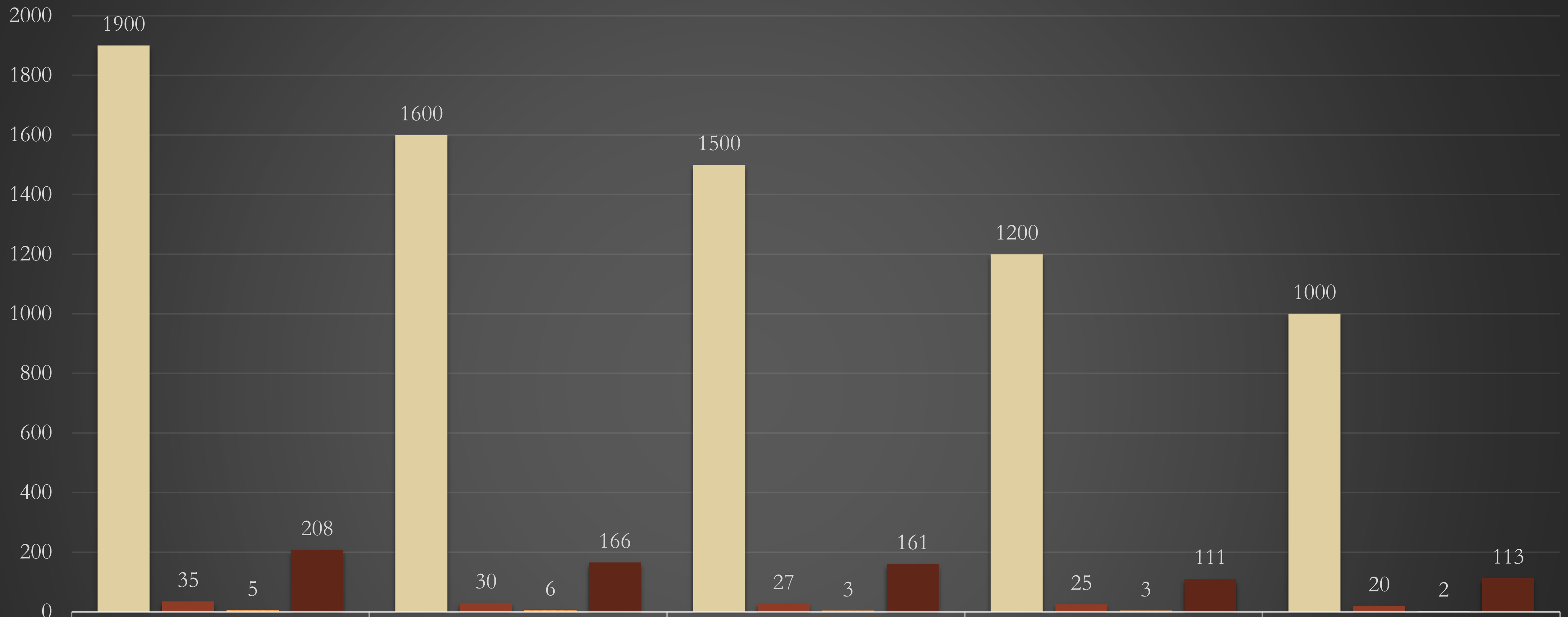
- Total Matches after Winning Toss: 41
- Total Wins after Winning Toss: 29
- Total Losses after Winning Toss: 12

- ***How does batting order affect Indian cricket team in ODIs world cup?***
  - **Top Order** is crucial for India’s success in World Cups. If the top 3 fire, India’s chancesof posting competitive totals or chasing targets increase significantly.
  - **Middle Order** holds the key to stabilizing innings, especially in run chases and knockoutmatches. Players like Yuvraj Singh and MS Dhoni have been instrumental in securingwins from challenging situations.
  - **Lower Order** rarely scores heavily but has delivered in crucial games, especially when middle-order collapses occur, providing crucial support in tight finishes.

performance of the top 4 current Indian cricket players using a heatmap



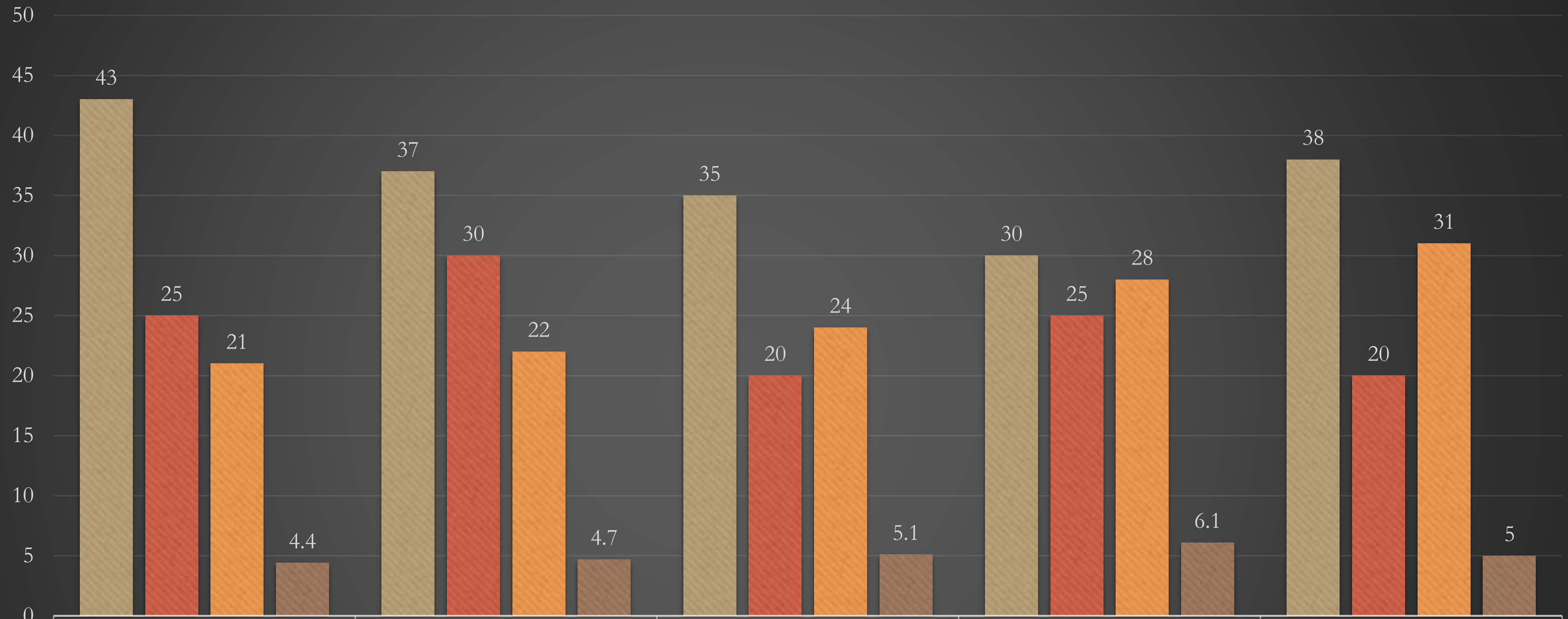
TOP 5 CRITICAL BATSMAN OF INDIA IN ODI FORMATES(Last 3 Years)



	Shubman Gill	Virat Kohli	Rohit Sharma	KL Rahul	Shreyas Iyer
Total Runs	1900	1600	1500	1200	1000
Total Matches	35	30	27	25	20
Centuries	5	6	3	3	2
Best Score	208	166	161	111	113

Total Runs   Total Matches   Centuries   Best Score

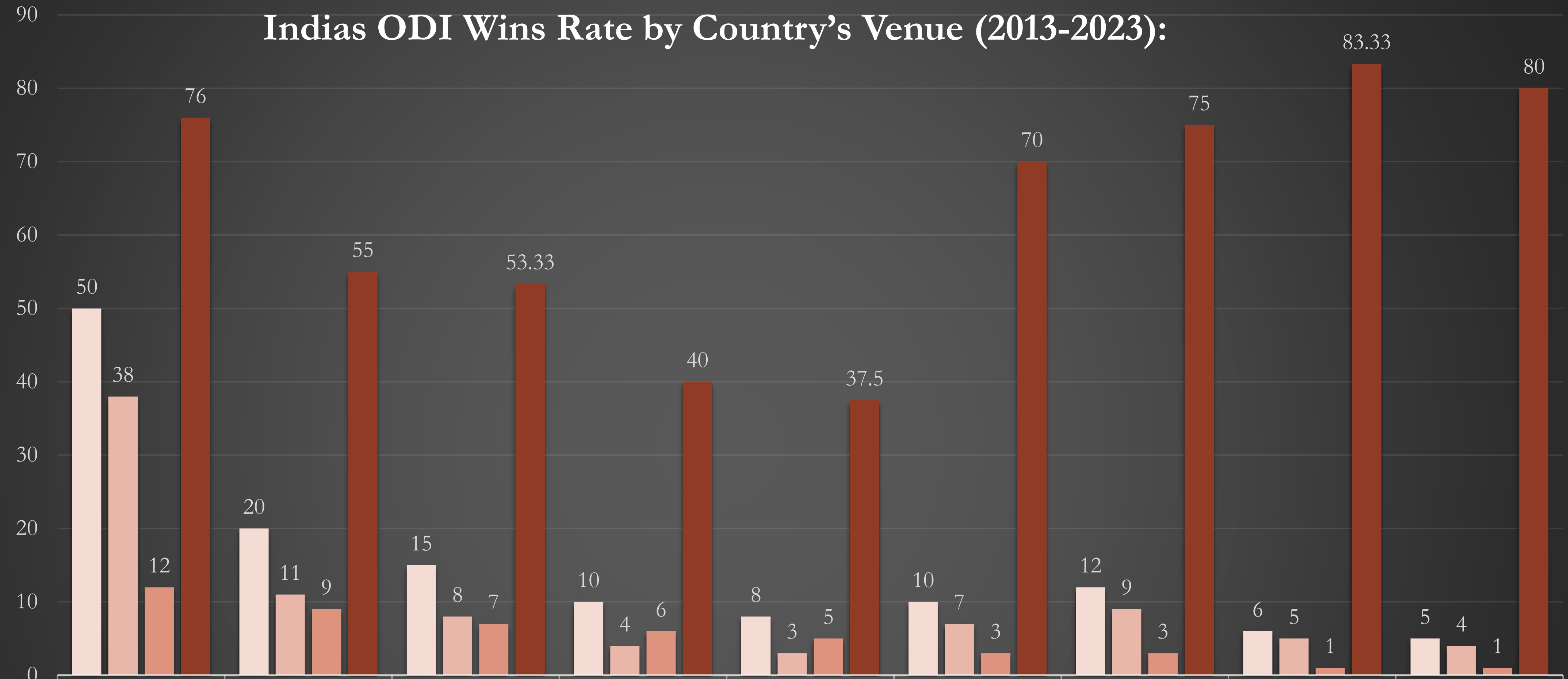
TOP 5 CRITICAL BOWLERS OF INDIA IN ODI FORMATES(Last 3 Years)



	Jasprit Bumrah	Mohammed Siraj	Kuldeep Yadav	Shardul Thakur	Yuzvendra Chahal
Wickets Taken	43	37	35	30	38
Total Matches	25	30	20	25	20
Bowling Avg.	21	22	24	28	31
Economy Rate	4.4	4.7	5.1	6.1	5

Wickets Taken Total Matches Bowling Avg. Economy Rate

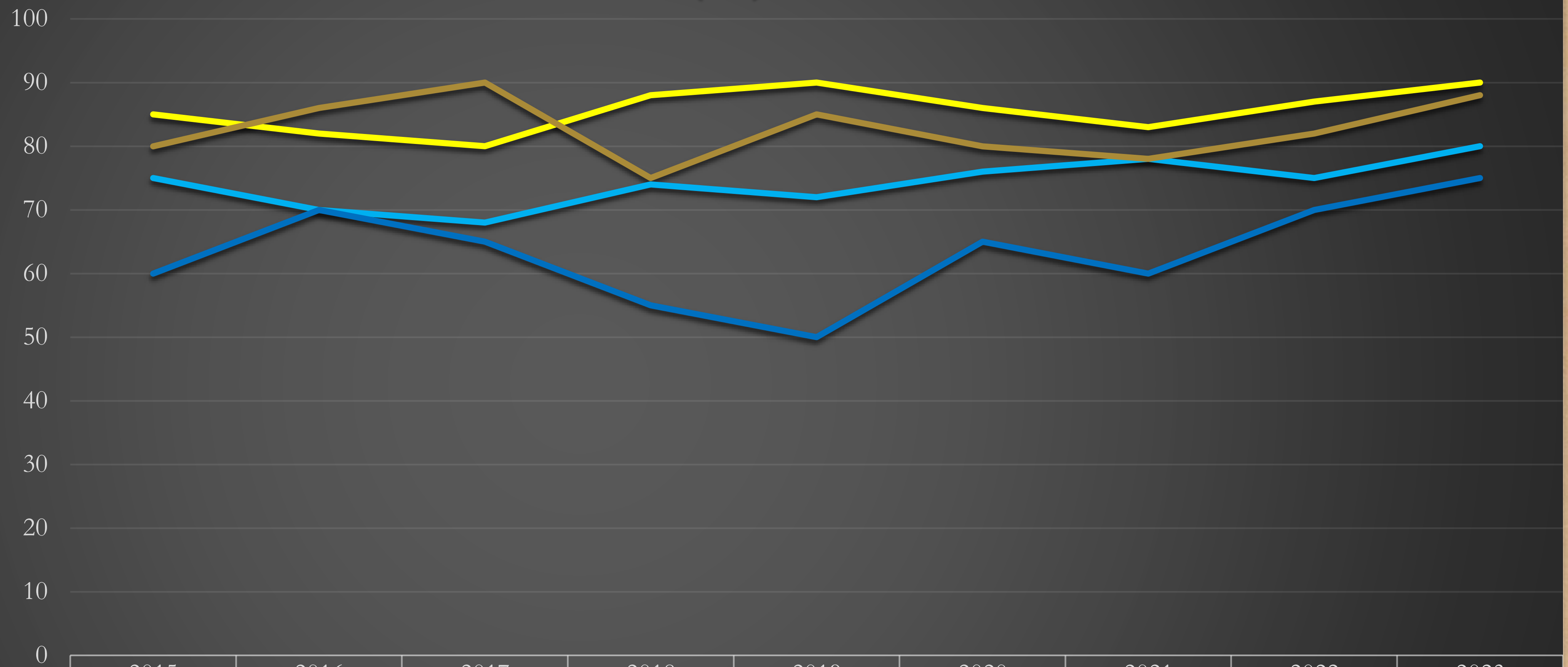
Indias ODI Wins Rate by Country's Venue (2013-2023):



	India	Australia	England	New Zealand	South Africa	Sri Lanka	UAE	Bangladesh	West Indies
Matches Played	50	20	15	10	8	10	12	6	5
Wins	38	11	8	4	3	7	9	5	4
Losses	12	9	7	6	5	3	3	1	1
Win Rate(%)	76	55	53.33	40	37.5	70	75	83.33	80

Matches Played   Wins   Losses   Win Rate(%)

Trend of India and Australia in under-pressure condition of odi matches(%)



	2015	2016	2017	2018	2019	2020	2021	2022	2023
Chasing Success (India)	75	70	68	74	72	76	78	75	80
Chasing Success (Australia)	85	82	80	88	90	86	83	87	90
Knockout Match Success (India)	60	70	65	55	50	65	60	70	75
Knockout Match Success (Australia)	80	86	90	75	85	80	78	82	88

Chasing Success (India) Chasing Success (Australia) Knockout Match Success (India) Knockout Match Success (Australia)

Weather Impact

Weather conditions significantly impact ODI cricket matches, influencing both gameplay and outcomes. Here are key points regarding its effects:

1. **Pitch and Ground Conditions**: Rain can affect the condition of the pitch and outfield, making it difficult for both batting and bowling. A damp pitch may favor bowlers, while a wet outfield can slow down the ball, reducing scoring opportunities.

2. **Duckworth-Lewis-Stern (DLS) Method**: In rain-affected matches, the DLS method is used to adjust targets, which can change the course of the match, especially when overs are lost.

3. **Visibility and Wind**: Cloud cover and wind can affect a bowler's swing, and reduced visibility can make batting harder. Both factors can alter a team's strategy during the match.Analyzing these factors with respect to Indian ODI matches can provide insights into the win-loss ratio under different weather conditions, which could help in preparation and planning.

Conclusion

- **Summary of Findings:**
  - Highlight major insights regarding player performance and match predictions.
  - Discuss how historical data impacts predictions for upcoming matches and the World Cup.
- **Future Recommendations:**
  - Suggest areas for further analysis, such as deeper dives into player fatigue or weather impacts on performance.
  - Recommend the use of real-time data integration for dynamic predictions during live matches.
  - Personal Recommandation:To enhance team versatility and adapt to dynamic match situations, it is beneficial to train bowlers in batting and batters in bowling. This cross-skill development not only increases tactical options but also provides greater flexibility in team selection. Bowlers who can contribute with the bat and batters who can offer an over or two add depth to both departments, improving overall team balance and resilience under pressure.
  - This approach can help teams better handle unexpected scenarios, reducing reliance on specialist players and fostering a more well-rounded squad.