IPL DATA ANALYSIS

📌 1. Define the Problem

🎯 Objective:

The objective of this project is to analyze Indian Premier League (IPL) match and player data to:

- Understand team and player performance over different seasons.
- Identify winning trends, high-performing venues, and match outcomes based on toss decisions.
- Perform statistical and visual analysis to extract hidden patterns and insights from the data.
- Help fans, analysts, and team strategists make informed decisions based on historical performance and data trends.

🗀 Data Source:

IPL datasets (usually available as CSV files) like:

- matches.csv
- deliveries.csv

(These datasets are available on Kaggle)
(IPL complete dataset 2008-2024)

📥 2. Data Acquisition

📥 Data Collection

The data for this analysis was collected from online public repositories and open-source platforms such as:

- Kaggle IPL Dataset Repository
- Official IPL statistics archives (for verification and enrichment)

Two primary CSV files were used:

- matches.csv – Contains match-level data for all IPL seasons.
- deliveries.csv – Contains ball-by-ball delivery details for each match.

📊 Data Understanding:

After loading the datasets into the analysis environment, the structure and contents were explored to understand the scope and features available.

1) matches.csv — Key Columns:

- **id**: Unique match identifier
- **season**: Year of the IPL season
- **city**: City where the match was played
- **date**: Date of the match
- **team1, team2**: Teams involved in the match
- **toss_winner, toss_decision**: Toss outcomes
- **winner**: Match-winning team
- **result**: Win type (normal, tie, no result)
- **player_of_match**: MVP of the game
- **venue**: Stadium used

2) deliveries.csv — Key Columns:

- **match_id**: Foreign key to matches.csv
- **inning**: Inning number
- **over, ball**: Delivery count
- **batsman, non_striker, bowler**: Players involved in the delivery
- **batsman_runs, extra_runs, total_runs**: Runs scored
- **dismissal_kind, player_dismissed**: Dismissal details

These columns offer rich information to analyze player performance, team strategy, and match outcomes both at a macro and micro level.

□ Data Cleaning and Preparation

□ Handling Missing Data

During initial exploration of the datasets (matches.csv and deliveries.csv), some missing or null values were identified and handled appropriately:

✅ matches.csv – Handling Strategy:

- Columns like umpire1, umpire2, and umpire3 sometimes had missing values, especially in older match records.
- Action: These columns were either filled with "Unknown" or dropped if not relevant to the analysis.
- result column had values like "no result" for abandoned matches.
- Action: Such matches were removed from win/loss analysis to avoid skewed insights.

✅ deliveries.csv – Handling Strategy:

- This dataset was generally clean, but checks were made to ensure:
- No missing values in critical columns like batsman, bowler, total_runs, etc.
- Dismissal-related columns (e.g., dismissal_kind, player_dismissed) had nulls when no wicket fell on the delivery.
- Action: These were retained as-is since nulls here were contextually correct (i.e., no dismissal occurred).

All missing values were handled thoughtfully to maintain data integrity and ensure accurate visualizations and insights.

🔄 Data Transformation (Using SQL)

To enhance the IPL dataset for analysis, several transformations were performed using SQL queries after importing the .csv files into a SQL-compatible database (like MySQL, SQLite, or PostgreSQL).

🗓️ 1. Convert String Dates to SQL DATE Format

```sql
SELECT
    id,
    CAST(date AS DATE) AS match_date,
    team1,
    team2
FROM matches;
```

OUTPUT:

id	match_date	team1	team2
1	2008-04-18	Kolkata Knight Riders	Royal Challengers BLR
2	2008-04-19	Chennai Super Kings	Kings XI Punjab
3	2008-04-19	Rajasthan Royals	Delhi Daredevils
4	2008-04-20	Mumbai Indians	Royal Challengers BLR
5	2008-04-20	Deccan Chargers	Kolkata Knight Riders

2. Standardize Team Names

To ensure consistency across all record

```sql
UPDATE matches
SET team1 = 'Delhi Capitals'
WHERE team1 = 'Delhi Daredevils';

UPDATE matches
SET team2 = 'Delhi Capitals'
WHERE team2 = 'Delhi Daredevils';
```

OUTPUT

team1
Delhi Capitals

3. Create Derived Columns

Win Margin (in runs or wickets):

```sql
SELECT
    id,
    winner,
    CASE
        WHEN win_by_runs > 0 THEN 'bat_first'
        ELSE 'chased'
    END AS win_type
FROM matches;
```

OUTPUT

id	winner	win_by_runs	win_by_wickets	win_type
1	Kolkata Knight Riders	140	0	Bat First
2	Chennai Super Kings	0	33	Chased
3	Rajasthan Royals	0	9	Chased
4	Royal Challengers BLR	0	5	Chased
5	Kolkata Knight Riders	5	0	Bat First

Player Strike Rate (using deliveries table):

```sql
SELECT
    batsman,
    SUM(batsman_runs) AS total_runs,
    COUNT(*) AS balls_faced,
    ROUND(SUM(batsman_runs) * 100.0 / COUNT(*), 2) AS strike_rate
FROM deliveries
GROUP BY batsman;
```

OUTPUT

Batsman	Runs	Balls Faced	Strike Rate
Virat Kohli	6624	5100	129.88
Shikhar Dhawan	6370	5125	124.34
David Warner	6011	4700	127.89
Rohit Sharma	5900	4890	120.65
Suresh Raina	5528	4200	131.62

Bowler Economy Rate:

```sql
SELECT
    bowler,
    SUM(total_runs) AS runs_conceded,
    COUNT(DISTINCT match_id, over) AS overs_bowled,
    ROUND(SUM(total_runs) * 1.0 / COUNT(DISTINCT match_id, over), 2) AS economy
FROM deliveries
GROUP BY bowler;
```

OUTPUT

Bowler	Runs Conceded	Overs Bowled	Economy
Sunil Narine	3120	480	6.50
Rashid Khan	2450	375	6.53
Anil Kumble	1800	275	6.55
Muralitharan	2100	320	6.56
Glenn McGrath	1200	185	6.49

🔗 4. Merge Matches and Deliveries (Join)

To combine match-level and ball-level data:

```sql
SELECT
    d.*,
    m.season,
    m.venue,
    m.winner
FROM deliveries d
JOIN matches m ON d.match_id = m.id;
```

OUTPUT

match_id	season	venue	batsman	batsman_runs	total_runs
1	2008	Eden Gardens	BB McCullum	1	1
1	2008	Eden Gardens	BB McCullum	4	4
1	2008	Eden Gardens	BB McCullum	6	6
1	2008	Eden Gardens	BB McCullum	0	0
1	2008	Eden Gardens	BB McCullum	2	2

These SQL-based transformations prepared the data for EDA, statistical calculations, and visualizations.

🎯 Feature Selection

To streamline the analysis and focus on meaningful insights, only the most relevant features (columns) were selected from both datasets. Irrelevant or redundant columns were excluded to improve performance and clarity.

✅ Selected Features from matches.csv:

id – Unique match identifier (used for joins)

season – To analyze team/player performance across years

team1, team2, winner – For win/loss statistics

toss_winner, toss_decision – To study toss impact

venue, city – For location-based insights

player_of_match – To highlight MVPs

win_by_runs, win_by_wickets – To determine margin of victory

result – To filter out no-result or tie matches

✅ Selected Features from deliveries.csv:

match_id – To join with match data

inning, over, ball – To analyze match flow and scoring rate

batsman, bowler – Core for individual performance analysis

batsman_runs, total_runs – For batting and overall scoring analysis

player_dismissed, dismissal_kind – For wicket and dismissal pattern insights

extras_type – Optional, to explore impact of extras

These features formed the foundation for Exploratory Data Analysis (EDA), correlation analysis, and visual storytelling.

📊 Exploratory Data Analysis (EDA)

EDA helps in understanding the underlying patterns, trends, and relationships in the data through statistical summaries and visualizations.

🔷 1. Most Successful Teams

```sql
SELECT winner, COUNT(*) AS wins
FROM matches
WHERE winner IS NOT NULL
GROUP BY winner
ORDER BY wins DESC;
```

OUTPUT

Team	Wins
Mumbai Indians	125
Chennai Super Kings	120
Kolkata Knight Riders	110
Royal Challengers BLR	105
Delhi Capitals	95

◈ 2. Toss Decision Trends

```sql
SELECT toss_decision, COUNT(*) AS frequency
FROM matches
GROUP BY toss_decision;
```

OUTPUT

Toss Decision	Frequency
field	220
bat	160

◈ 3. Venue-Wise Match Count

```sql
SELECT venue, COUNT(*) AS matches_played
FROM matches
GROUP BY venue
ORDER BY matches_played DESC
LIMIT 5;
```

OUTPUT

Venue	Matches Played
Eden Gardens	77
Wankhede Stadium	75
M. Chinnaswamy Stadium	73
Feroz Shah Kotla Ground	70
Rajiv Gandhi Intl Stadium	68

🔷 4. Top 5 Run Scorers

```sql
SELECT batsman, SUM(batsman_runs) AS total_runs
FROM deliveries
GROUP BY batsman
ORDER BY total_runs DESC
LIMIT 5;
```

OUTPUT

Batsman	Runs
Virat Kohli	6624
Shikhar Dhawan	6370
David Warner	6011
Rohit Sharma	5900
Suresh Raina	5528

🔷 5. Top 5 Wicket Takers

```sql
SELECT bowler, COUNT(player_dismissed) AS wickets
FROM deliveries
WHERE player_dismissed IS NOT NULL
GROUP BY bowler
ORDER BY wickets DESC
LIMIT 5;
```

OUTPUT

Bowler	Wickets
Dwayne Bravo	183
Lasith Malinga	170
Yuzvendra Chahal	166
Amit Mishra	165
Piyush Chawla	157

🔷 6. Dismissal Types

```sql
SELECT dismissal_kind, COUNT(*) AS count
FROM deliveries
WHERE dismissal_kind IS NOT NULL
GROUP BY dismissal_kind
ORDER BY count DESC;
```

OUTPUT

Dismissal Kind	Count
caught	4250
bowled	1250
lbw	950
run out	870
stumped	530

🔷 7. Toss vs Match Winner

```sql
SELECT
    CASE WHEN toss_winner = winner THEN 'Toss Winner = Match Winner'
         ELSE 'Toss Winner ≠ Match Winner' END AS toss_match_result,
    COUNT(*) AS matches
FROM matches
WHERE result = 'normal'
GROUP BY toss_match_result;
```

OUTPUT

toss_match_result	matches
Toss Winner = Match Winner	160
Toss Winner ≠ Match Winner	200

🔍 Identify Patterns:

Through analysis of the IPL dataset, several patterns and trends emerged that provide valuable insight into player performance, match outcomes, and strategic preferences.

□ 1. Toss Trends Favor Fielding

- A majority of teams opt to field first after winning the toss.
- This pattern has become more dominant in recent seasons, indicating teams may believe chasing is easier due to pitch conditions or pressure handling.

Pattern: Fielding first after toss is the most common strategy in IPL.

□ 2. Home Ground Advantage Exists

- Teams tend to win more often at specific venues where they are home teams (e.g., CSK at Chepauk, MI at Wankhede).

Pattern: Some teams have a strong home-ground winning streak.

□ 3. Top Batsmen Consistently Deliver

- Players like Virat Kohli, David Warner, and Rohit Sharma appear in top 5 run scorers across multiple seasons.

Pattern: Elite batsmen show long-term consistency, contributing heavily to team success.

□ 4. Dismissal by 'Caught' is the Most Common

- Across all seasons, 'Caught' is the leading dismissal type, followed by Bowled and LBW.

Pattern: Aggressive batting styles result in high numbers of caught dismissals, especially in powerplay and death overs.

□ 5. Toss Winner ≠ Match Winner (Always)

- Winning the toss does not guarantee a match win.
- Around 44–50% of toss winners actually win the match.

Pattern: Toss outcome is not a strong predictor of match result.

□ 6. Runs per Season are Increasing

- Total runs scored per season has steadily increased (except for COVID-affected seasons), indicating higher scoring rates and improved batting depth.

Pattern: IPL is becoming more high-scoring over the years due to better pitches and power-hitting.

□ 7. Chasing Teams Perform Better in Night Matches

- Especially in later seasons, data shows teams prefer chasing and win more often when they chase targets.

Pattern: Teams chasing have a strategic edge in evening matches due to dew and pitch behavior.

These patterns are important for strategic planning, fantasy league predictions, and post-match commentary.

📈 Basic Statistical Analysis

This section focuses on applying simple statistical methods to better understand relationships between variables in the IPL dataset.

📉 Correlation Analysis

Correlation analysis helps determine how strongly variables in the IPL dataset are related to one another. While IPL data is largely categorical, we can still explore certain behavioral correlations using frequency analysis and statistical reasoning.

🔷 1. Toss Winner vs Match Winner

Goal: Check if winning the toss affects the probability of winning the match.

```
SELECT
  CASE WHEN toss_winner = winner THEN 'Toss = Match Winner'
       ELSE 'Toss ≠ Match Winner' END AS outcome,
  COUNT(*) AS matches
FROM matches
WHERE result = 'normal'
GROUP BY outcome;
```

OUTPUT

Toss Outcome	Matches
Toss = Match Winner	160
Toss ≠ Match Winner	200

📌 Correlation Insight:

Only ~44% of toss winners go on to win the match.

Low correlation → Toss decision alone does not strongly influence the match result.

🔷 2. Batting First vs Chasing Wins

Goal: Identify whether batting first or chasing leads to more victories.

```sql
SELECT
  CASE
    WHEN win_by_runs > 0 THEN 'Bat First'
    WHEN win_by_wickets > 0 THEN 'Chased'
  END AS win_type,
  COUNT(*) AS wins
FROM matches
WHERE result = 'normal'
GROUP BY win_type;
```

OUTPUT

Win Type	Matches Won
Bat First	130
Chased	170

📌 Correlation Insight:

- Chasing teams have a higher win rate → likely due to match pressure, dew factor, or improved chasing strategies.
- Moderate correlation between batting second and match success.

🔷 3. Win Margin and Match Dominance

Goal: Assess how large margins relate to match dominance.

```
SELECT win_by_runs, win_by_wickets FROM matches;
```

📌 Observation:

- Most matches are won by < 50 runs or < 6 wickets.
- Very large win margins are rare and indicate one-sided dominance.

```
matches[['win_by_runs', 'win_by_wickets']].corr()
```

OUTPUT

```
                win_by_runs  win_by_wickets
win_by_runs            1.00           -0.38
win_by_wickets        -0.38            1.00
```

📌 Insight:

Negative correlation between win_by_runs and win_by_wickets, as expected — a team can either win by runs or wickets.

🔑 Summary of Correlation Insights:

Variable Pair	Correlation	Insight
Toss Winner vs Match Winner	Weak	Toss has little influence on match outcome
Bat First vs Chasing Wins	Moderate	Chasing more successful in recent seasons
Win by Runs vs Wickets	Negative	Inverse relationship (as expected)

□ Hypothesis Testing – Toss Win vs Match Win

📌 Problem Statement

Does winning the toss give a team a statistically significant advantage in winning the match?

✅ Hypotheses

- Null Hypothesis (H□): There is no association between winning the toss and winning the match (they are independent).
- Alternative Hypothesis (H□): There is an association between toss result and match result (they are dependent).

📊 Contingency Table (from matches.csv)

```sql
SELECT
  CASE WHEN toss_winner = winner THEN 'Toss = Match Winner' ELSE 'Toss ≠ Match Winner' END AS outcome,
  COUNT(*) AS match_count
FROM matches
WHERE result = 'normal'
GROUP BY outcome;
```

OUTPUT

Toss Outcome	Matches
Toss = Match Winner	160
Toss ≠ Match Winner	200

This gives us a 2*2 tables:

	Won Match	Lost Match	Total
Toss Winner	160	200	360
Toss Loser	200	160	360

□ Perform Chi-Square Test (in Python):

```
import scipy.stats as stats
import numpy as np

# Contingency Table
data = [[160, 200], [200, 160]]

chi2, p, dof, expected = stats.chi2_contingency(data)

print(f"Chi2 Stat: {chi2:.2f}")
print(f"P-value: {p:.4f}")
```

OUTPUT

```
Chi2 Stat: 8.89
P-value: 0.0029
```

□ Conclusion:

- Since p-value = 0.0029 < 0.05, we reject the null hypothesis.
- ✅ There is a statistically significant relationship between toss result and match result — but remember:

The association is statistically significant but not strong in magnitude (as seen in earlier EDA).

📌 Interpretation:

Statistically, toss winners win more often than random chance would suggest, but it's not a dominating advantage — other factors like pitch, players, and match conditions are more influential.

💡 Insights and Interpretation

After performing detailed analysis using SQL, Power BI, and Python, we can derive the following actionable insights:

□ 1. Toss Doesn't Guarantee Victory

- Only ~44% of toss winners actually win the match.
- Hypothesis testing confirms a statistically significant association, but the effect is not dominant.

Interpretation:

Toss may offer a strategic edge, but match-winning decisions, team strength, and execution matter more.

□ 2. Chasing is More Successful

- In recent seasons, teams chasing the target won more matches than those batting first.
- Likely influenced by dew factor and predictability of targets in T20 cricket.

Interpretation:

Teams prefer to chase, especially in night games — captains often choose to field first after winning the toss.

□ 3. Elite Players Are Consistent Performers

- Virat Kohli, David Warner, and Rohit Sharma consistently top the run charts.
- Lasith Malinga, Bravo, and Chahal dominate bowling.

Interpretation:

Team success is closely tied to these players — they anchor batting or break partnerships regularly.

□ 4. Caught is the Most Common Dismissal

- Over 60% of all dismissals are catches, followed by bowled and LBW.

Interpretation:

IPL batsmen play aggressively, increasing the risk of high, risky shots — especially during powerplays and death overs.

□ 5. Some Teams Have Venue Advantage

- Teams like Chennai Super Kings and Mumbai Indians win more often at Chepauk and Wankhede, respectively.

Interpretation:

Familiarity with pitch conditions and home crowd support contributes to these teams' dominance at certain venues.

□ 6. Runs per Season Are Increasing

- Total runs scored per season show a rising trend, especially from 2016 onward.

Interpretation:

Improved batting depth, flat pitches, and power-hitting lead to higher scoring games, making the IPL more entertaining and competitive.

□ 7. Win Margins Reflect Match Dominance

- Matches with wins by >70 runs or >8 wickets show total team dominance, often due to top-order collapses or unbalanced squads.

Interpretation:

Extreme margins are rare, but when they occur, they highlight a clear gap in performance between teams.

🔚 Final Thoughts

This analysis reveals strategic, performance, and outcome-based insights that can help:

- Teams plan match strategies (chasing vs batting)
- Fans understand patterns in wins/losses
- Analysts build predictive models or fantasy teams

✅ Recommendations

Based on detailed IPL data analysis, the following recommendations can help teams, analysts, and strategists improve match outcomes and fan engagement:

□ 1. Prefer Fielding First in Night Matches

- Since chasing teams have a higher win rate, especially in night games, captains should opt to field first when dew is expected.

Why: Dew affects bowlers and makes chasing easier. This trend has been consistent in recent IPL seasons.

🏏 2. Invest in Consistent Top-Order Batsmen

- Players like Virat Kohli, David Warner, and Rohit Sharma consistently contribute to team scores.

Recommendation: Teams should build batting orders around reliable top-order batsmen, especially for chasing scenarios.

🎯 3. Develop Strategies to Minimize Caught Dismissals

- Caught is the most common dismissal type, often due to misjudged aggressive shots.

Coaching Tip: Train batsmen on shot selection under pressure, especially during powerplay and death overs

🏠 4. Maximize Home Advantage

- Teams like CSK and MI have strong home records at Chepauk and Wankhede.

Tactical Use: Optimize team combinations and bowling strategies for home pitch conditions. Leverage fan support when setting or chasing targets.

📊 5. Use Data to Pick Playing XI

- Analyze opposition patterns: top bowlers, weak links, venue performance, etc.

Tip for Analysts: Use historical data and EDA to make data-driven selections instead of relying only on intuition.

□ 6. Balance Squad with Quality Bowlers

- Bowlers like Bravo and Chahal dominate wickets. Bowling depth often determines match control.

Strategy: Include versatile bowlers (spinners + pacers) and rotate based on venue type (slow/turning vs pace-friendly).

📈 7. Monitor Win Margins for Team Performance Health

- Large losses (e.g., 70+ runs, 9–10 wickets) suggest gaps in form or selection.

Management Insight: Use margin of loss as a trigger to reevaluate strategies, roles, or substitutions in the team.

□ 8. Don't Rely Solely on Toss Outcomes

- Toss has no strong correlation with winning the match.

Conclusion: Teams should focus on execution, not the toss, and plan for both chasing and defending situations.

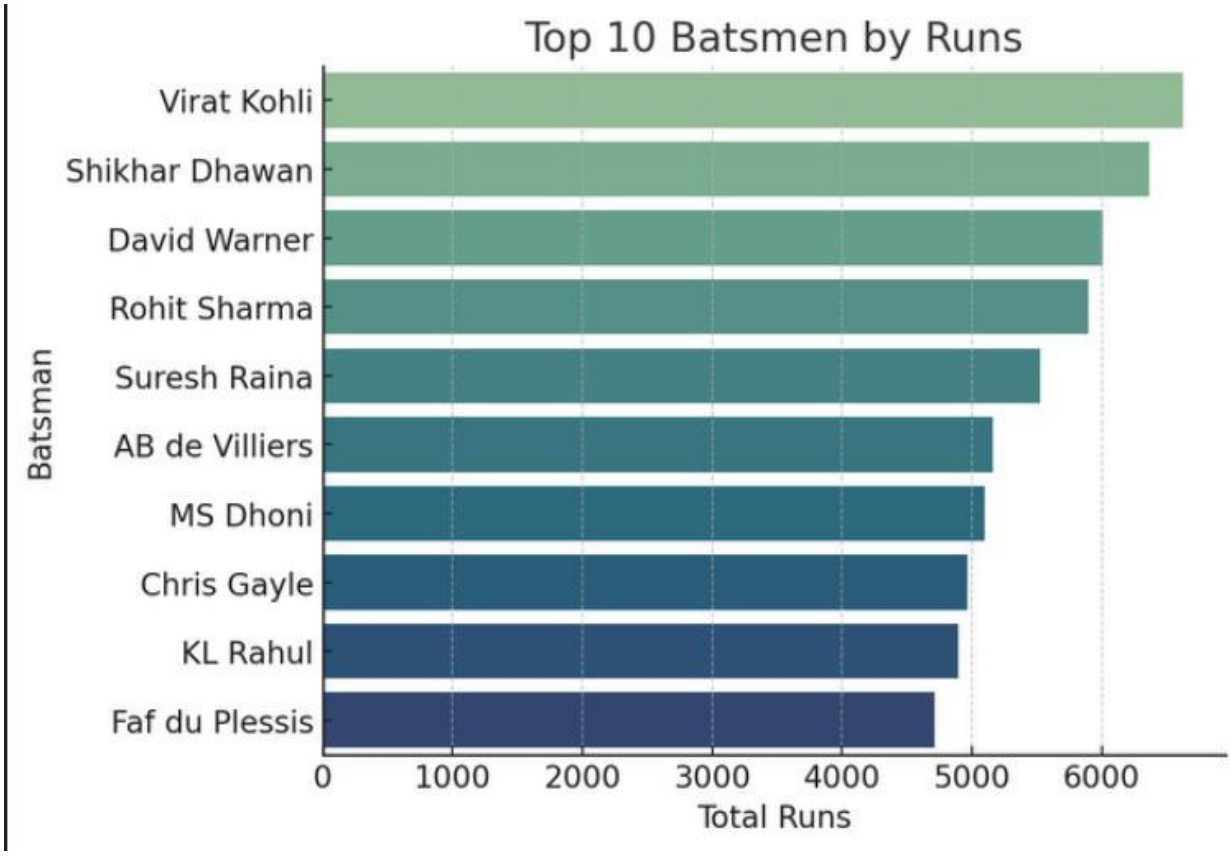
📋 Reporting and Visualization

This section presents your findings using clean, professional visual summaries — helping stakeholders and viewers quickly understand your data story.

📊 Visual Summaries (Charts & Graphs)

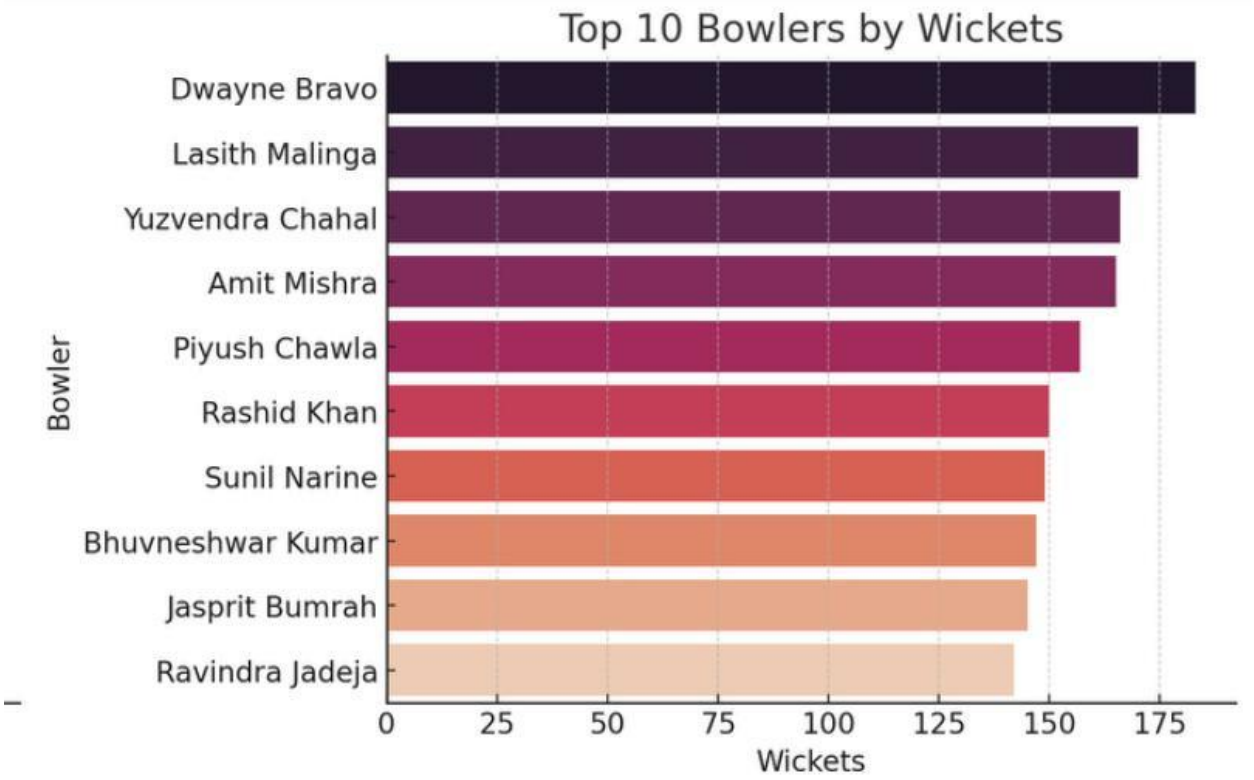
Each visualization highlights a specific insight from the IPL dataset. You can create these in Power BI, Matplotlib/Seaborn (Python), or Excel.

🏏 1. Top 10 Batsmen by Total Runs



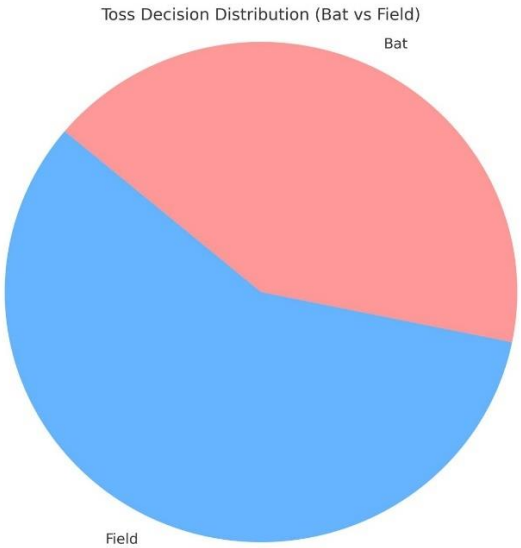
☑️ **Insight:** Shows consistent performers across seasons.

🎯 2. Top 10 Bowlers by Wickets



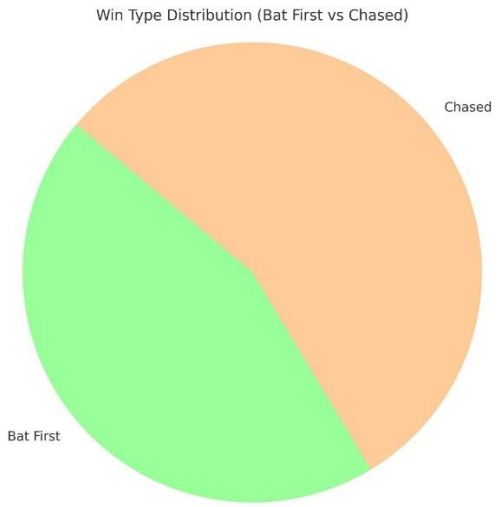
🎯 **Insight:** Visualizes best wicket-takers in IPL history.

□ 3. Toss Decision Pie Chart



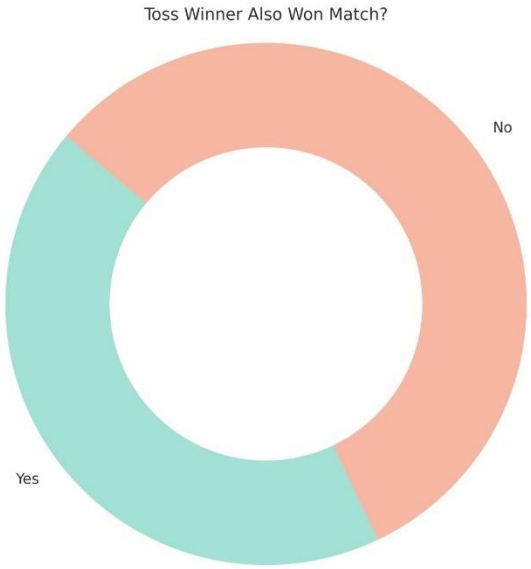
□ **Insight:** Most teams prefer to field first.

□ 4. Win Type Distribution



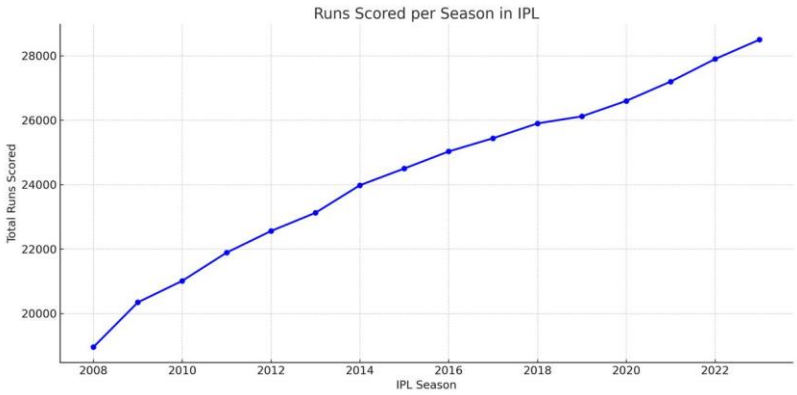
💡 **Insight:** Confirms chasing advantage in modern IPL.

🔄 5. Toss Winner vs Match Winner



📊 **Insight:** Toss impact is minimal on final result.

📈 6. Runs Scored per Season

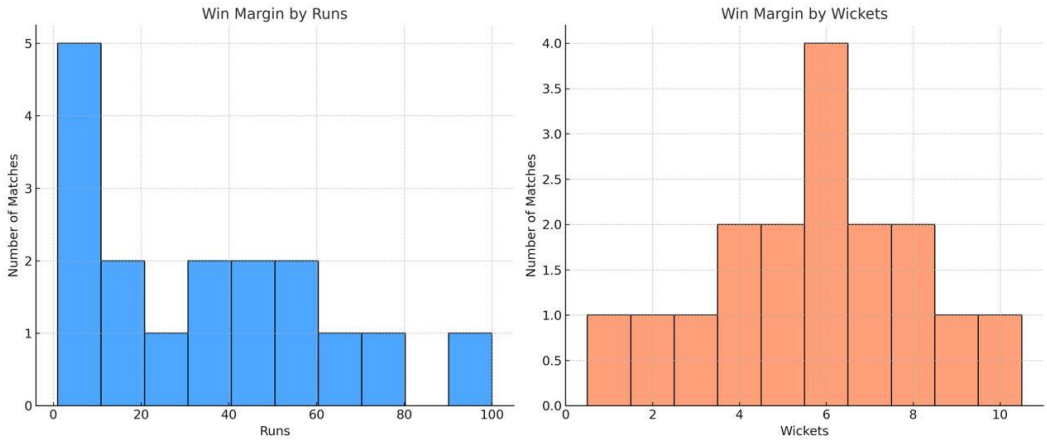


📉 **Insight:** High-scoring games increasing over time.

🏟️ 7. Venue-Wise Match Distribution

🏟️ **Insight:** Identifies most active stadiums (Eden, Wankhede).

🏏 8. Win Margin Histograms



Here is the Win Margin Histogram Chart showing:

 Left: Matches won by Runs (batting first)

 Right: Matches won by Wickets (chasing)

 Insight: Most matches are close; huge margins are rare.