

Statistical Insights into International Cricket Batsmen Performance

Statistical Computing (STAT 50001)

Nehana Jampanaboyana
Vaishak Balachandra
Naga Niharika



Introduction:

Why Analyze Batting Average and Strike Rate in Cricket?

Cricket, often referred to as a game of statistics.

Cricket Analytics: Highlights the broader domain of leveraging data science techniques to analyze cricket performance metrics.



Overview

- Data Collection & Preprocessing
- Exploratory Data Analysis (EDA)
- Model Development
- Insights & Validation

Motivation

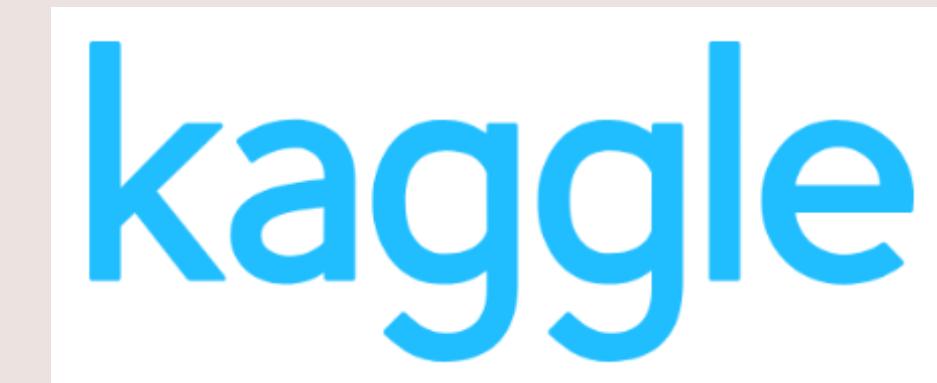
Simple Answer:- **Cricket Analytics!!**

Why Are These Metrics Important?

- Batting Average: Measures consistency and reliability over a player's career.
- Strike Rate: Highlights scoring efficiency, critical for limited-overs formats.

Data Description

- The data is sourced from the [ICC Cricket Dataset on Kaggle](#).
- Contains detailed statistics on cricket players' performances across various matches and seasons.



Variables (15)

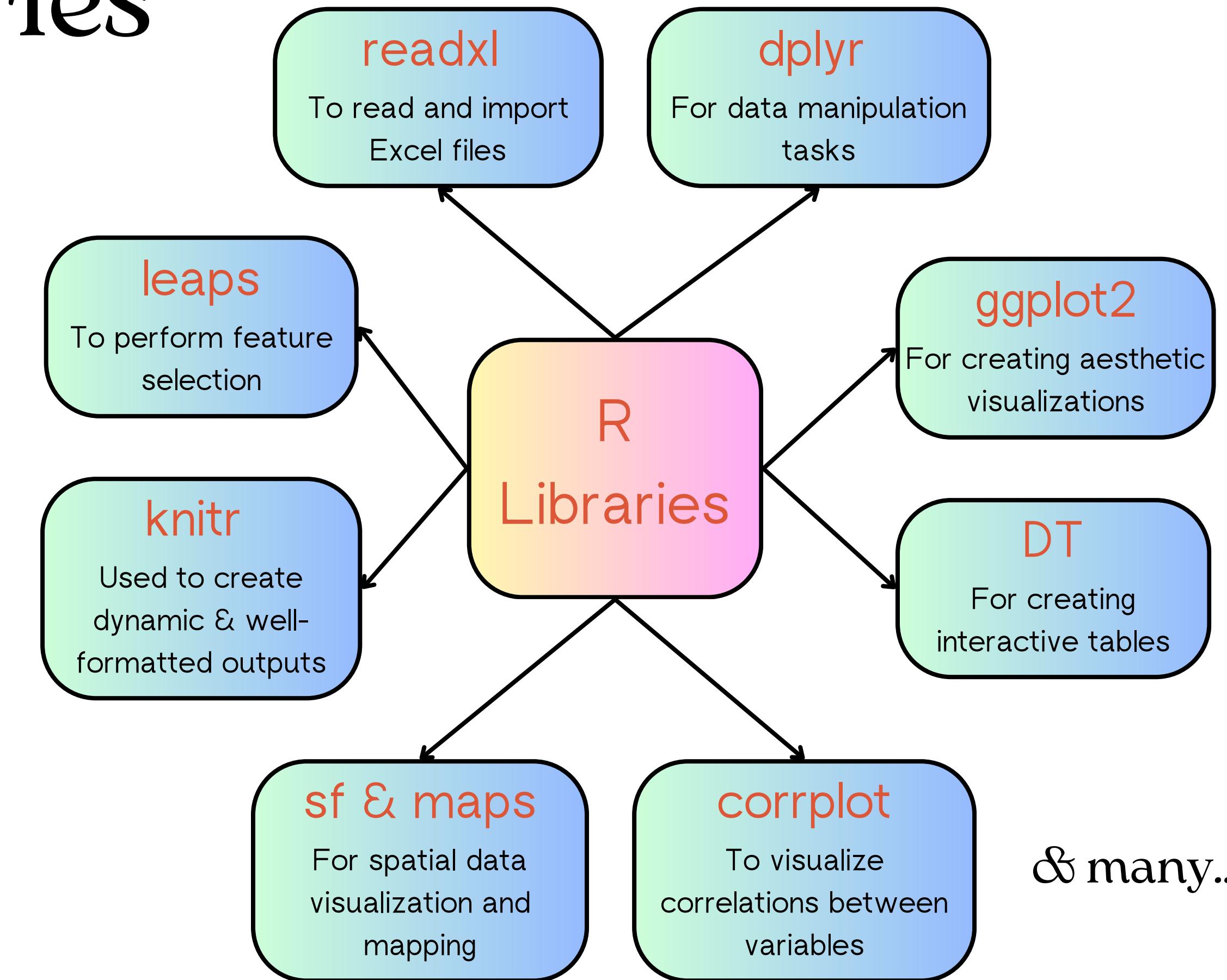
- Player: Name of the player with Country name.
- Span: Start Year and End Year.
- Mat: Number of matches played by the player.
- Inns: Number of innings played by the player.
- NO: Number of times the player remained not out.
- Runs: Total runs scored by the player.
- HS: Player's highest score in a match with out or not status.
- Ave: Average runs scored per innings.
- BF: Number of balls faced by the player.
- SR: A measure of how quickly the player scores.
- 100: Number of centuries scored by the player.
- 50: Number of half-centuries scored.
- 0: Number of times the player got out without scoring.
- 4s: Number of boundaries (4 runs) hit by the player.
- 6s: Number of sixes hit by the player.

Player	Span	Mat	Inns	NO	Runs	HS	Ave	BF	SR	100	50	0	4s	6s
V Kohli (INDIA)	2010-2019	75	70	20	2633	94*	52.66	1907	138.07	0	24	2	247	71
RG Sharma (INDIA)	2007-2019	104	96	14	2633	118	32.1	1905	138.21	4	19	6	234	120
MJ Guptill (NZ)	2009-2019	83	80	7	2436	105	33.36	1810	134.58	2	15	2	215	113
Shoaib Malik (ICC/PAK)	2006-2019	111	104	30	2263	75	30.58	1824	124.06	0	7	1	186	61
BB McCullum (NZ)	2005-2015	71	70	10	2140	123	35.66	1571	136.21	2	13	3	199	91

`Table. head(Cricket_Data)`

<https://www.kaggle.com/datasets/mahendran1/icc-cricket>

Tools & Libraries



Data Preprocessing

Data preprocessing is the process of transforming raw data into a format that's easier for machines to understand and analyze.

Column Preprocessing

- Splitting columns.

Variables (15)

- Player: Name of the player with Country name.
- Span: Start Year and End Year.
- Mat: Number of matches played by the player.
- Inns: Number of innings played by the player.
- NO: Number of times the player remained not out.
- Runs: Total runs scored by the player.
- HS: Player's highest score in a match with out or not status.
- Ave: Average runs scored per innings.
- BF: Number of balls faced by the player.
- SR: A measure of how quickly the player scores.
- 100: Number of centuries scored by the player.
- 50: Number of half-centuries scored.
- 0: Number of times the player got out without scoring.
- 4s: Number of boundaries (4 runs) hit by the player.
- 6s: Number of sixes hit by the player.



Variables (18)

```
> colnames(Batsman_clean)
[1] "Match"      "Innings"     "NotOut"      "Runs"       "HighestScore"
[6] "BattingAverage" "BallsFaced" "StrikeRate"   "Hundreds"   "Fifties"
[11] "Ducks"       "Fours"      "Sixes"      "HS_star"    "PlayerName"
[16] "Country"     "StartTime"   "EndYear"
```

Row Preprocessing

- Removing rows with NA values.
- Removing duplicate rows.

```
> dim(Batsman)
[1] 2006    15
```



```
> dim(Batsman_clean)
[1] 1672    18
```

Exploratory Data Analysis (EDA):

```
> summary(Batsman_clean)
```

Summary Statistics and Key Insights

- The median number of matches played is 7, but the maximum is 111, highlighting disparity in experience.
- Median runs (45.5) are significantly lower than maximum runs (2633), reflecting a few exceptional performers.
- Average fours (13.66) and sixes (5.51) highlight boundary play's importance in total runs.
- Average hundreds (0.0329) and a maximum of 4 indicate only a few achieve this milestone consistently.
- Average ducks (0.90 on 10) suggest batsmen contribute positively in most matches, which indicate less probability of scoring a duck.
- Median Highest Score (HS) of 22 compared to maximum (172) shows variability in top-scoring ability.
- Most players have career spans concentrated between 2010 and 2019, with few exceptions.
- Average fifty-count (0.6328) and a maximum of 24 indicate some players consistently perform.
- Players scoring centuries in a match (max 4) demonstrate elite performance levels.

Top Players Based on Performance

```
> print(top_runs)
# A tibble: 10 × 2
  PlayerName      Runs
  <chr>        <dbl>
1 V Kohli       2633
2 RG Sharma     2633
3 MJ Guptill    2436
4 Shoaib Malik  2263
5 BB McCullum   2140
6 DA Warner     2079
7 EJG Morgan    2002
8 Mohammad Shahzad 1936
9 JP Duminy     1934
10 PR Stirling   1929
```

```
> print(top_avg)
# A tibble: 10 × 3
  PlayerName    BattingAverage Match
  <chr>          <dbl>    <dbl>
1 Abu Hider      58.0     13
2 Nouman Sarwar  57.3     14
3 V Kohli        52.7     75
4 HA Varaiya     51.0     25
5 Babar Azam     50.2     36
6 A Symonds      48.1     14
7 RN ten Doeschate 44.4     22
8 KL Rahul        43.8     34
9 Hazratullah Zazai 43.2     13
10 AR Nurse       42.5     13
```

```
> print(top_strike_rate)
# A tibble: 10 × 3
  PlayerName     StrikeRate Match
  <chr>           <dbl>    <dbl>
1 Washington Sundar 217.     18
2 Hasan Ali         174.     30
3 A Symonds         169.     14
4 Hazratullah Zazai 163.     13
5 Izatullah Dawlatzai 162.     14
6 Dawlat Zadran     162.     34
7 C Munro            160.     60
8 GJ Maxwell         160.     61
9 JC Tredwell        160.     17
10 AJ Finch          156.     58
```

(Constraints/ Condition: with at least mean matches played)

Frequency Distribution

Frequency Distribution of Centuries (Hundreds)		
Number of Centuries	Frequency	
1	0	1630
2	1	33
3	2	6
4	3	2
5	4	1

Show 5 entries

Search:

Frequency Distribution of Half-Centuries (Fifties)		
Number of Half-Centuries	Frequency	
1	0	1312
2	1	161
3	2	70
4	3	38
5	4	28

Show 5 entries

Search:

Frequency Distribution of Ducks		
Number of Ducks	Frequency	
1	0	858
2	1	466
3	2	179
4	3	84
5	4	43

Show 5 entries

Search:

Showing 1 to 5 of 11 entries

Previous 1 2 3 4 Next

Showing 1 to 5 of 18 entries

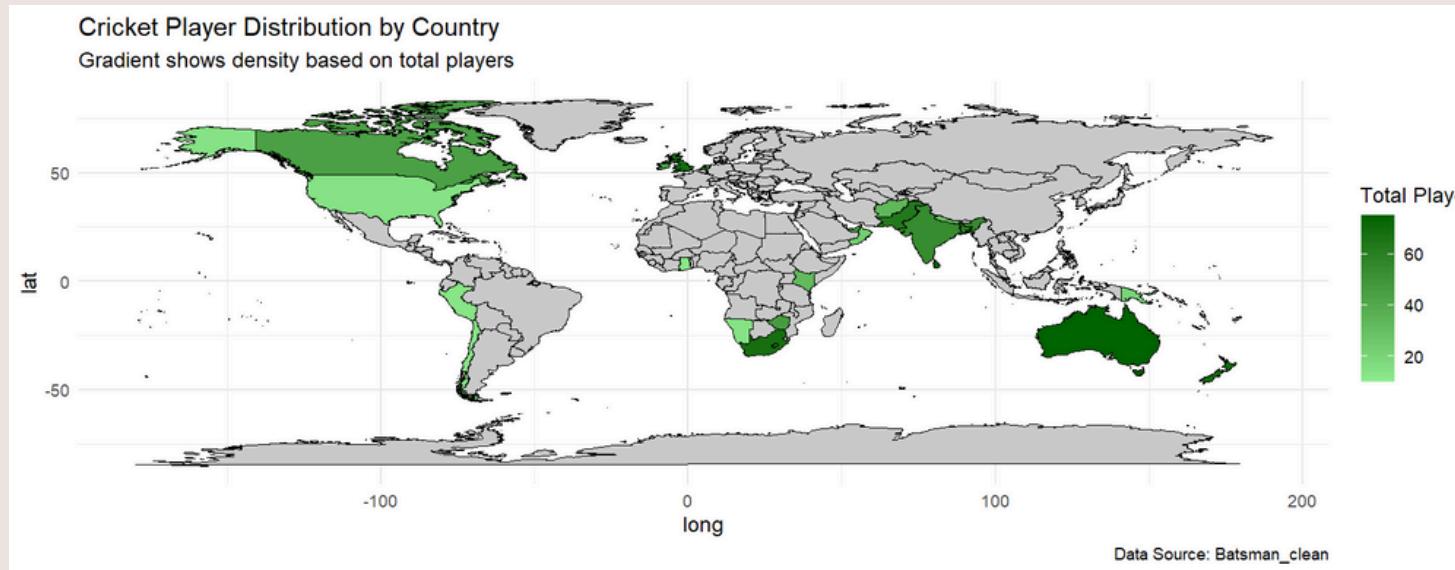
Previous 1 2 3 4 Next

Showing 1 to 5 of 5 entries

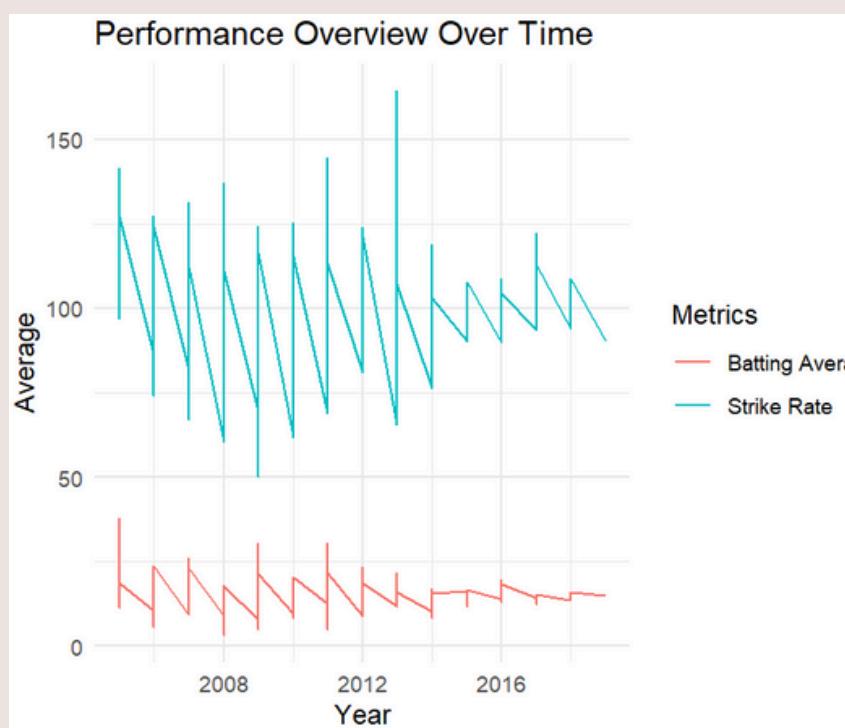
Previous 1 2 3 Next

Exploratory Data Analysis (EDA):

Country Based Analysis



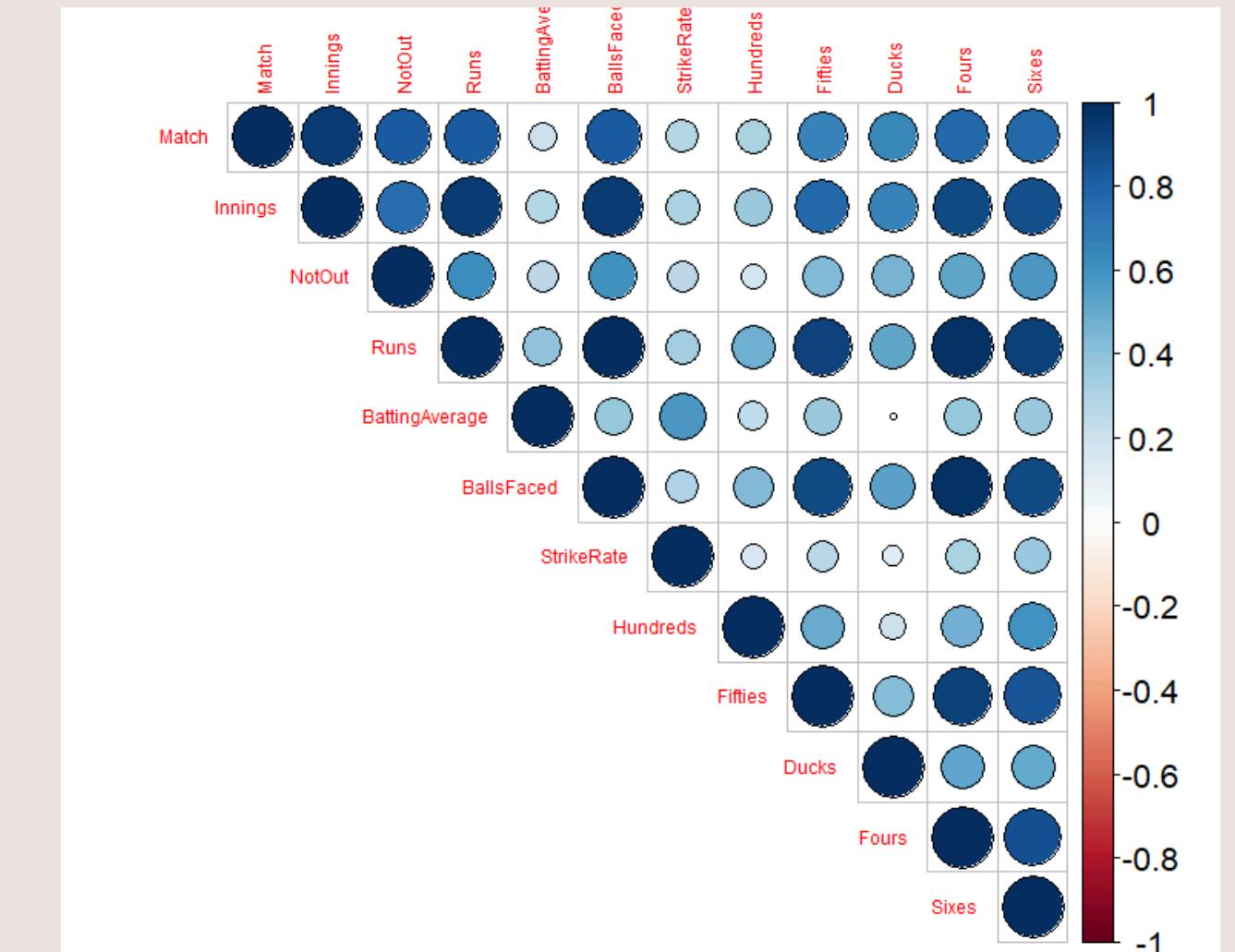
Performance Over Time



INSIGHTS:

- Batting Average showing a gradual decline and Strike Rate exhibiting more significant variability
- Performance in terms of Batting Average has steadily decreased since around 2012, while Strike Rate remains highly inconsistent over the years.

Correlation Matrix

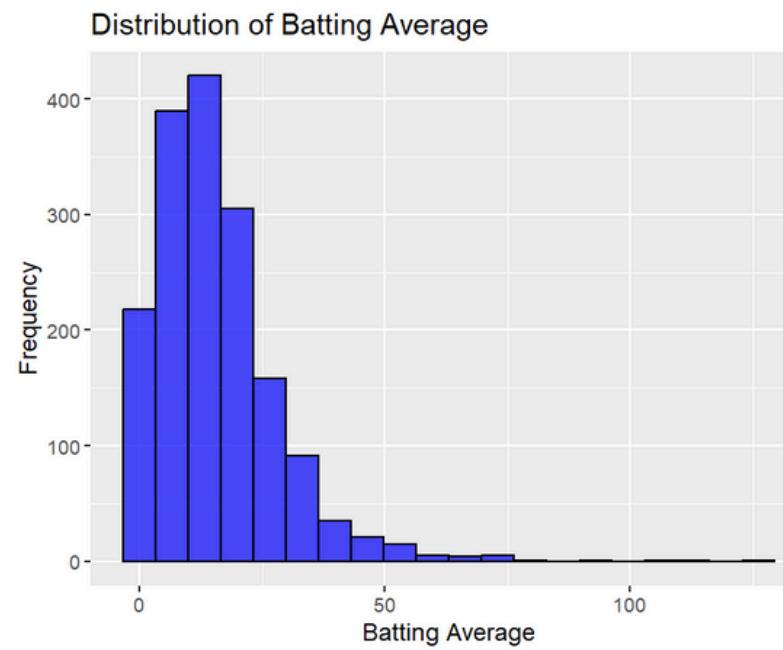


INSIGHTS:

- BATTING AVERAGE:**
 - has relation with: Balls Faced, Strike Rate, Hundreds, Fifties, Runs etc.,
- STRIKE RATE:**
 - has relation with: Fours, Sixes, Balls Faced, Ducks etc.,

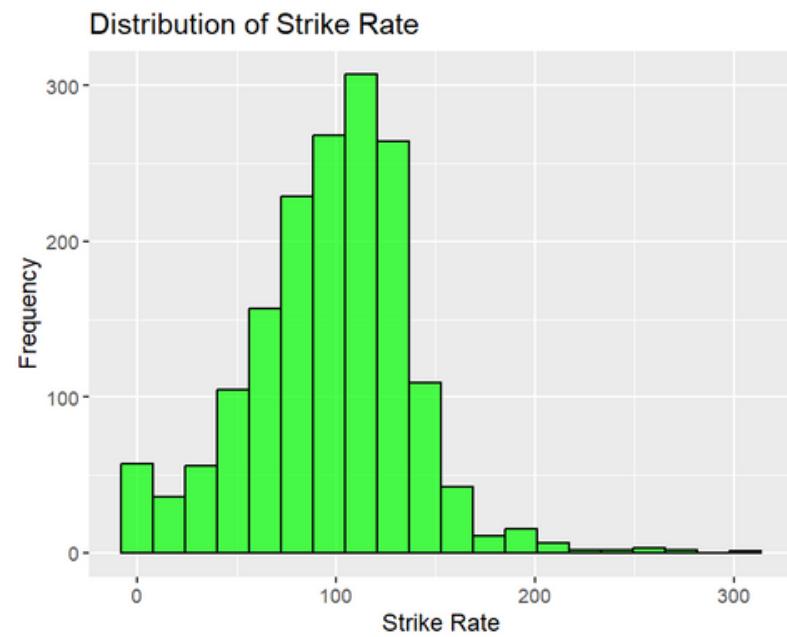
Exploratory Data Analysis (EDA):

Batting Average and Strike Rate



INSIGHTS:

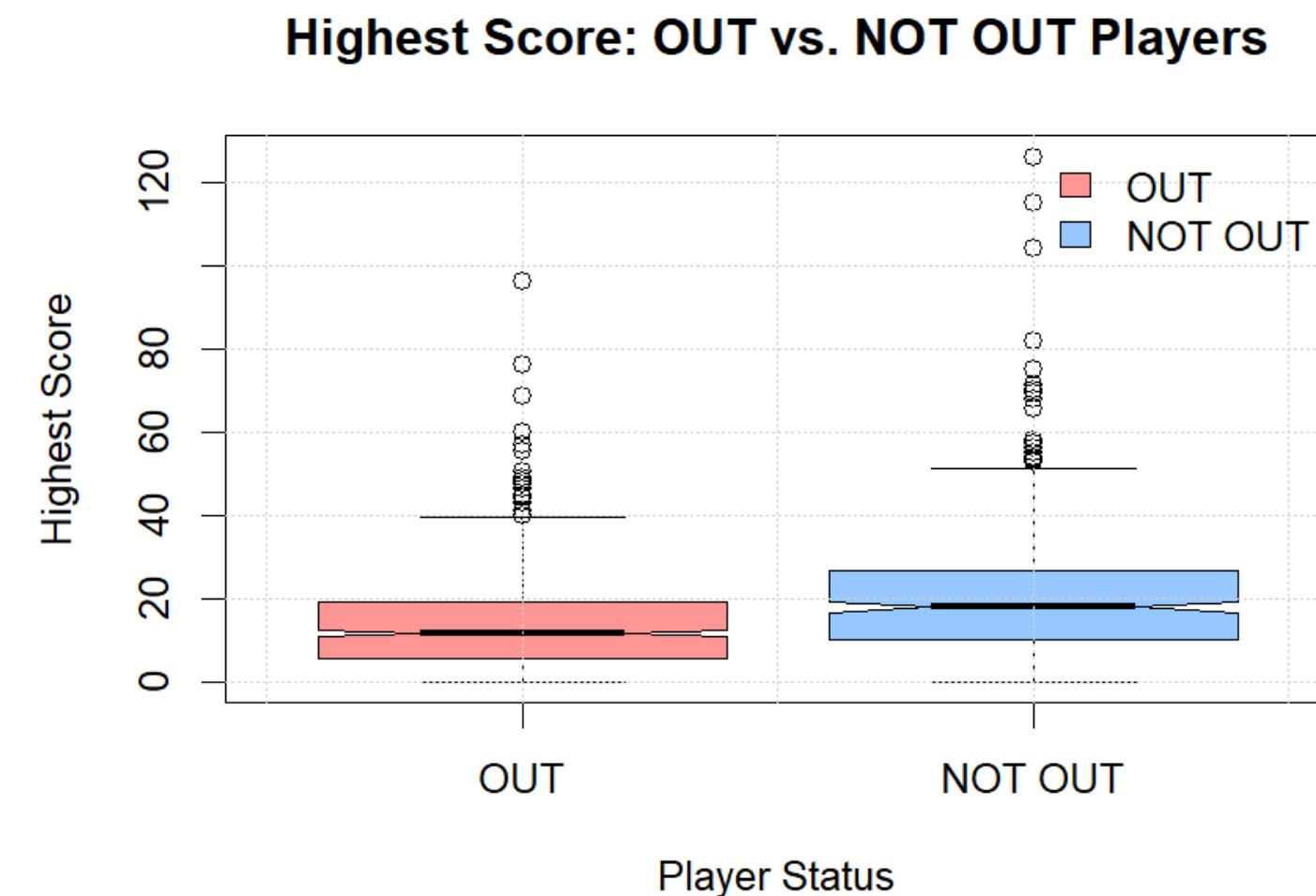
- Majority of players may contribute modestly to the team's runs, while players with a batting average above 50 are rare and may represent elite performers.



INSIGHTS:

- Strike rates around 100 are typical for average players, while exceptionally high or low strike rates are less common and might represent either elite performers or under-performers.

Highest Score (Out/ Not Out) Status



INSIGHTS:

- Players who achieve higher scores are more likely to remain "NOT OUT" at the end.

Model 1: Batting Average Prediction

Regression Analysis for Batting Average

Model

To identify the best predictors of a player's batting average using stepwise variable selection (via `regsubsets`).

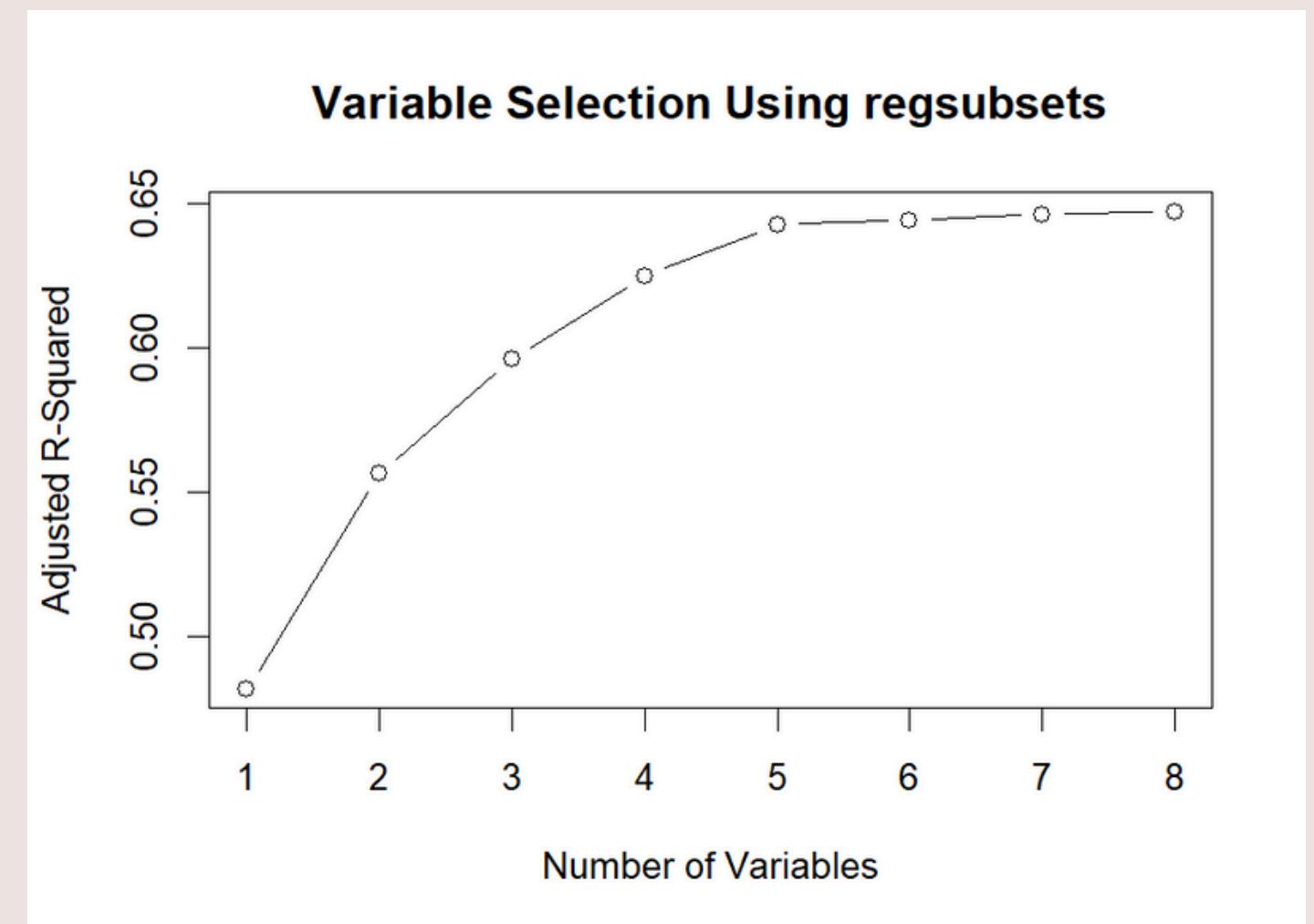
Variables Used:

`"(Intercept)", "Match", "NotOut", "Runs",
"HighestScore", "BallsFaced", "StrikeRate",
"Fifties", "Ducks".`

Flexibility:

Flexibility is provided to the stakeholder/user, allowing them to choose the number of variables to include in the model.

```
> cat("Choose the number of variables to include (1 to 8): ")  
Choose the number of variables to include (1 to 8):  
> num_vars = as.integer(readline())  
6
```



Multiple Linear Regression Model

Using Stepwise Selection

Example: Considered number of variables = 6

Variables selected: "(Intercept)", "Match", "NotOut", "HighestScore", "BallsFaced",
"StrikeRate", "Ducks".

BattingAverage ~ Match + NotOut + HighestScore + BallsFaced + StrikeRate + Ducks

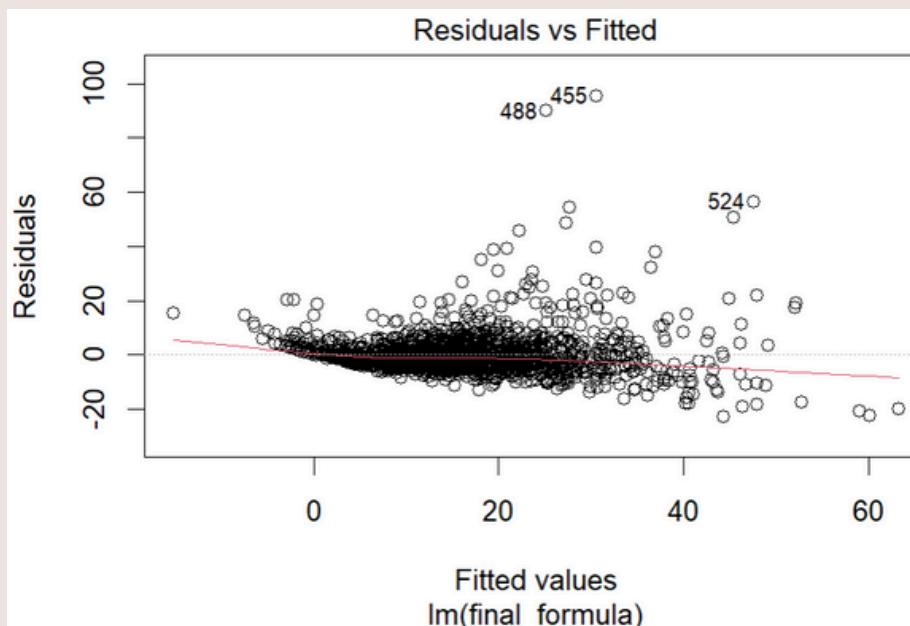
$$\text{BattingAverage} = 2.3689 - 0.4422 * \text{Match} + 1.6498 * \text{NotOut} + \\ 0.3371 * \text{HighestScore} + 0.0051 * \text{BallsFaced} + \\ 0.0641 * \text{StrikeRate} - 1.6451 * \text{Ducks}$$

Multiple Linear Regression Model

Performing BOXCox Transformation

Transformed Mode

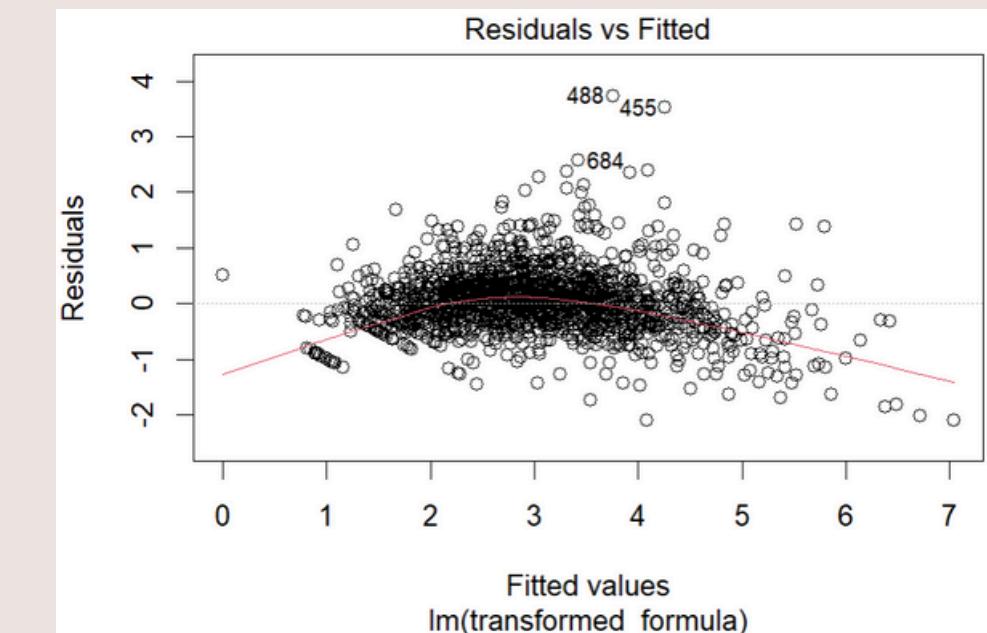
$$2.3689 - 0.4422 * \text{Match} + 1.6498 * \text{NotOut} + \\ \text{BattingAverage}^{0.42} = 0.3371 * \text{HighestScore} + 0.0051 * \text{BallsFaced} + \\ 0.0641 * \text{StrikeRate} - 1.6451 * \text{Ducks}$$



Before boxcox transformation

INFERENCE:

- All 6 variables are significant.
- Multiple R-squared: 0.6455
- Adjusted R-squared: 0.6442



after boxcox transformation

INFERENCE:

- All 6 variables are significant.
- Multiple R-squared: 0.7491
- Adjusted R-squared: 0.7482

Model 2: Strike Rate Prediction

Regression Analysis for Strike Rate

Example: Considered number of variables = 5

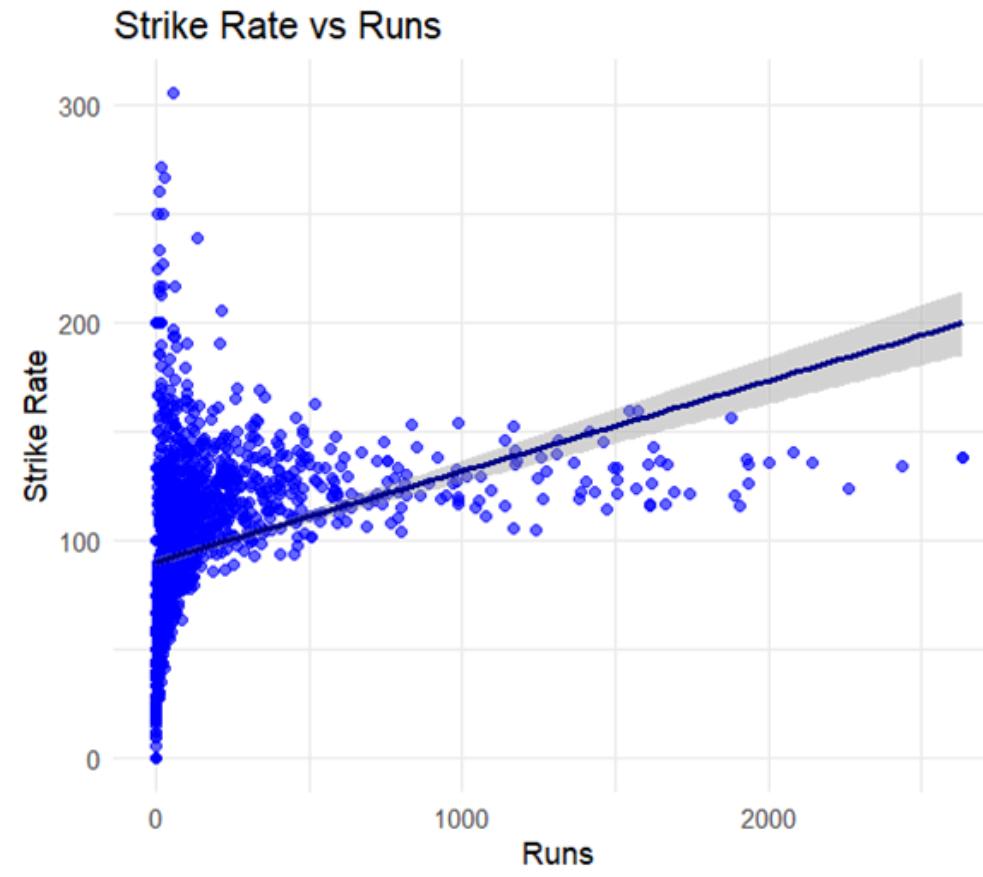
Variables selected: "(Intercept)", "Runs", "BallsFaced", "Hundreds", "Fifties", "Fours"

StrikeRate ~ Runs + BallsFaced + Hundreds + Fifties + Fours

$$92.7713 + 0.398 * \text{Runs} - 0.4141 * \text{BallsFaced}$$

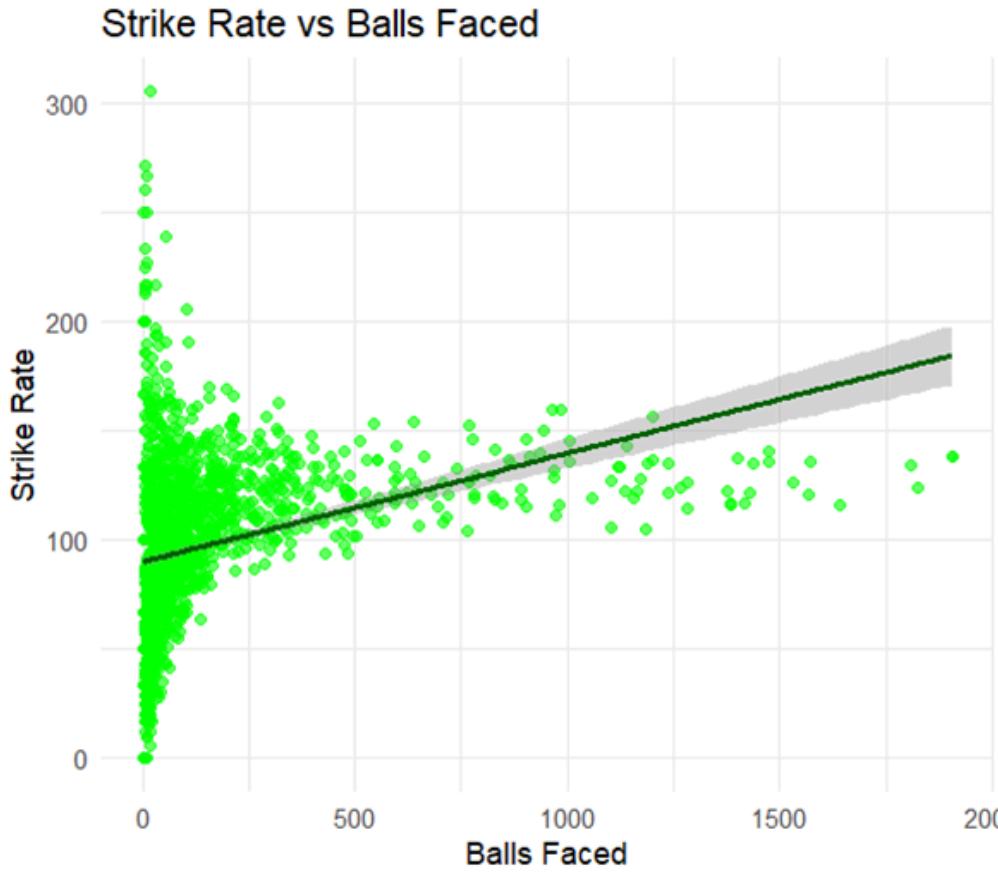
$$\begin{aligned} \text{Strike Rate Average} = & - 20.7269 * \text{Hundreds} - 7.8796 * \text{Fifties} + \\ & 0.1236 * \text{Fours} \end{aligned}$$

Visualizations



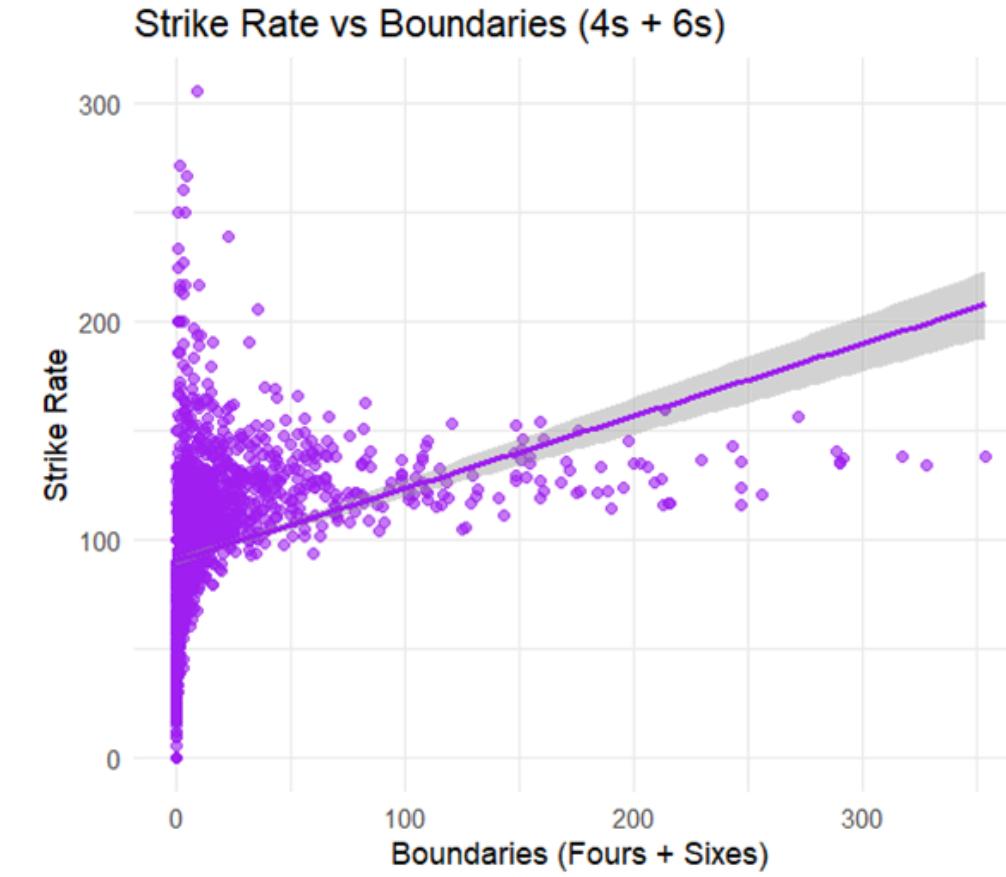
INFERENCE:

- Strong positive Correlation



INFERENCE:

- Positive Correlation



INFERENCE:

- Strong Positive Correlation

Thank you so
much!



Open to

Q&A