Meerut Institute of Engineering and Technology, Meerut

miet
GROUP OF INSTITUTIONS

2022-2023

MINI PROJECT REPORT

On

"IPL   WIN   PREDICTOR"

BACHELOR OF TECHNOLOGY IN COMPUTER
SCIENCE AND ENGINEERING

Submitted to-

Mr. Shivam Singhal

(Department of Computer Science & Engineering)

Submitted by-

Student name : ANSHUL SHARMA   Roll No:2000680100057

5 th SEMESTER
DEPARTMENT OF COMPUTER
SCIENCE AND ENGINEERING
MEERUT INSTITUTE OF ENGINEERING AND TECHNOLOGY, MEERUT

<u>DECLARATION</u>

We hereby declare that the project entitled -"**IPL WIN PREDICTOR**", which is being submitted as Mini Project in department of Computer Science and engineering to Meerut Institute of Engineering and Technology, Meerut (U.P.) is an authentic record of our genuine work done under

the guidance of Mr."SHIVAM SINGHAL " of Computer Science and Engineering, Meerut Institute of Engineering and Technology, Meerut.

Date: 16 NOVEMBER 2022   ANSHUL SHARMA

Place: MEERUT   2000680100057
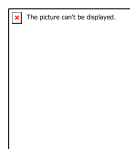
CERTIFICATE

This is to certify that mini project report entitled – "**IPL WIN PREDICTOR**" submitted by "Anshul Sharma" has been carried out under the guidance of Mr."SHIVAM SINGHAL" of Computer Science and Engineering, Meerut Institute of Engineering and Technology, Meerut. This project report is approved for Mini Project (KCS 554) in 5TH semester in "MACHINE LEARNING" from Meerut Institute of Engineering and Technology, Meerut.

Mr. SHIVAM SINGHAL

Date: 16 NOVEMBER 2022

Certificate

ACKNOWLEDGEMENT

I express my sincere indebtedness towards our guide Mr."SHIVAM SINGHAL " of Computer Science and Engineering, Meerut Institute of Engineering and Technology, Meerut for his valuable suggestion, guidance and supervision throughout the work. Without his king patronage and guidance the project would not have taken shape. I would also like to express my gratitude and sincere regards for his kind approval of the project. Time to time counseling and advises.

I would also like to thank to our HoD Dr. "MUKESH RAWAT", Department of Computer Science and engineering, Meerut Institute of Engineering and Technology, Meerut for his expert advice and counseling from time to time.

I owe sincere thanks to all the faculty members in the department of Computer Science and engineering for their kind guidance and encouragement time to time.

Date: 16 NOVEMBER 2022   ANSHUL SHARMA

Table of contents

Description

DESCRIPTION OF MACHINE LEARNING :

Machine learning (ML) is a type of artificial intelligence (AI) allows software applications to become more accurate at Predicting outcomes without being explicitly programmed to do so. Machine learning algorithms use historical data as input to predict new output values.

Recommendation engines are a common use case for machine learning. Other popular uses include fraud detection, spam filtering, malware threat detect business process automation (BPA) and Predictive maintenance.

TOOLBOX

TYPES OF MACHINE LEARNING

Supervised Machine Learning

Unsupervised Machine Learning

Semi-Supervised Learning

Reinforcement Learning

Neural networks

Deep learning

Machine learning

AI
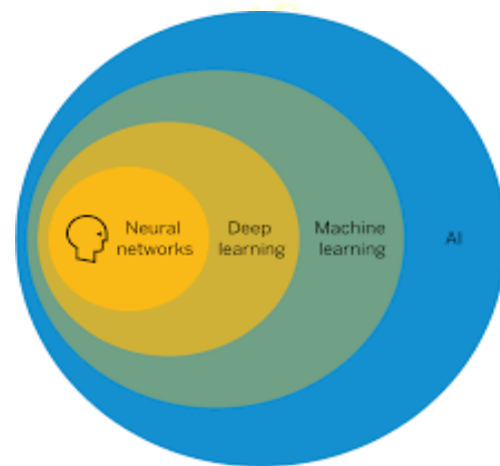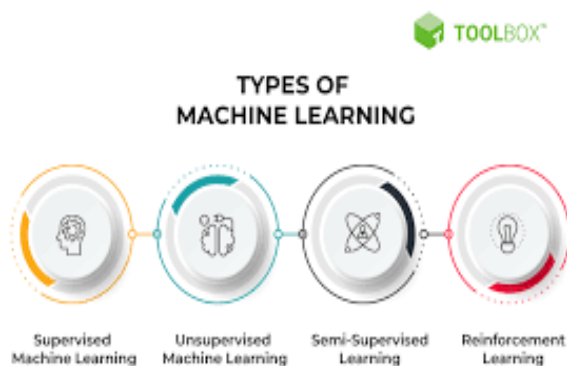
. NumPy and Pandas

NumPy stands for 'Numerical Python' or 'Numeric Python'. It is an open source module of Python which provides fast mathematical computation on arrays and matrices. Since, arrays and matrices are an essential part of the Machine Learning ecosystem, NumPy along with Machine Learning modules like Scikit-learn, Pandas, Matplotlib, TensorFlow, etc. complete the Python Machine Learning Ecosystem.

NumPy provides the essential multi-dimensional array-oriented computing functionalities designed for high-level mathematical functions and scientific computation. Numpy can be imported into the notebook using

Some of the important attributes of a NumPy object are:

1-**Ndim:** displays the dimension of the array

2-**Shape:** returns a tuple of integers indicating the size of the array

3-**Size:** returns the total number of elements in the NumPy array

4-**Dtype**: returns the type of elements in the array, i.e., int64, character

5- **Itemsize:** returns the size in bytes of each item

NumPy

```
>>> import numpy as np
```

Similar to NumPy, Pandas is one of the most widely used python libraries in data science. It provide high-performance, easy to use structures and data analysis tools. Unlike NumPy library which provides objects for multi-dimensional arrays, Pandas provides in-memory 2d table object called Dataframe.
It is like a spreadsheet with column names and rowlabel

Some commonly used data structures in pandas:

Series objects: 1D array, similar to a column in a spreadsheet

DataFrame objects: 2D table, similar to a spreadsheet

Panel objects: Dictionary of DataFrames, similar to sheet in MS Excel.

CRICKET
DATASET

FLOWCHART

DATA PREPROCESSING

FEATURE EXTRACTION

MACHINE
LEARNING
ALGORITHM

DECISION
TREE
ALGORITHM

RANDOM
FOREST
ALGORITHM

ACCURACY
80 PERCENT

ACCURACY
90 PERCENT

OUTPUT   SCREEN

IPL Win Predictor

IPL

TAKE ANY 2 TEAM AND FILL DATA AND FIND THE WINNING PERCENTAGE:

Select the batting team

| Chennai Super Kings ▼ |

Select the bowling team

| Chennai Super Kings ▼ |

Select host city

| Abu Dhabi ▼ |

Target

| 0.00 − + |

Score

| 0.00 − + |

Overs completed

| 0.00 − + |

Wickets out

| 0.00 − + |

Predict Probability

Select the batting team

| Chennai Super Kings ▼ |

Select the bowling team

| Royal Challengers Bangalore ▼ |

Select host city

| Sharjah ▼ |

Target

| 210.00 − + |

Score	Overs completed	Wickets out
90.00 − +	10.00 − +	2.00 − +

Predict Probability

Chennai Super Kings- 58%

Royal Challengers Bangalore- 42%

·

CHECK PROJECT HELP OF THIS EXAMPLE:

LIVE

44th Match (N), Sharjah, Sep 30 2021, Indian Premier League

Sunrisers Hyderabad   134/7

Chennai Super Kings •   (10.4/20 ov, target 135) **79/1**

Super Kings need 56 runs in 56 balls.

Current RR: 7.40 • Required RR: 6.00 • Last 5 ov (RR): 33/1 (6.60)   Win Probability: CSK 96% • SRH 4%

Live   Scorecard   Live Blog   Commentary   Live Stats   Overs   Playing XI   News   Videos   Photos   Table

								T20 CAREER			
BATTERS	R	B	4s	6s	SR	THIS BOWLER	LAST 5 BALLS	MAT	RUNS	HS	AVE
Moeen Ali* (lhb)	1	1	0	0	100.00	1 (lb)	1 (lb)	192	4151	121*	25.01

CREATE IN THIS TABLE

batting_team	bowling_team	city	runs_left	balls_left	wickets_left	total_runs_x	crr	rrr	result
Kings XI Punjab	Rajasthan Royals	Durban	208	108	8	211	1.500000	11.555556	0
Royal Challengers Bangalore	Mumbai Indians	Mumbai	91	66	9	141	5.555556	8.272727	1
Kolkata Knight Riders	Delhi Daredevils	Delhi	97	83	8	146	7.945946	7.012048	1
Mumbai Indians	Chennai Super Kings	Mumbai	69	34	6	168	6.906977	12.176471	0
Rajasthan Royals	Chennai Super Kings	Chennai	113	81	9	157	6.769231	8.370370	0
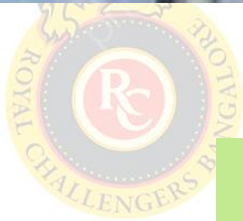
+ Code + Text

```
import numpy as np
import pandas as pd
```

+ Code   + Text

```
[2] match = pd.read_csv('/content/drive/MyDrive/ipl/matches.csv')
    delivery = pd.read_csv('/content/drive/MyDrive/ipl/deliveries.csv')
```

```
[3] match.head()
```

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets	player_of_match	venue	umpire1	umpire2	umpire3
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	35	0	Yuvraj Singh	Rajiv Gandhi International Stadium, Uppal	AY Dandekar	NJ Llong	NaN
1	2	IPL-2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Rising Pune Supergiant	0	7	SPD Smith	Maharashtra Cricket Association Stadium	A Nand Kishore	S Ravi	NaN
2	3	IPL-2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	Kolkata Knight Riders	0	10	CA Lynn	Saurashtra Cricket Association Stadium	Nitin Menon	CK Nandan	NaN
3	4	IPL-2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	0	6	GJ Maxwell	Holkar Cricket Stadium	AK Chaudhary	C Shamshuddin	NaN
4	5	IPL-2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	15	0	KM Jadhav	M Chinnaswamy Stadium	NaN	NaN	NaN

```
[4] match.shape
    (756, 18)
```

```
[5] delivery.head()
```

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler	is_super_over	...	bye_runs	legbye_runs	noball_runs	penalty_runs	batsman_runs	extra_runs	total_runs	player_dismissed	dismissal_kind	fielder
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	0	0	0	0	NaN	NaN	NaN
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	0	0	0	0	NaN	NaN	NaN
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	0	4	0	4	NaN	NaN	NaN
3	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	0	0	0	0	NaN	NaN	NaN
4	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	S Dhawan	TS Mills	0	...	0	0	0	0	0	2	2	NaN	NaN	NaN

5 rows × 21 columns

+ Code + Text

```
[6] # from current dataset some feature extract
    # total score find match all innings
    # delivery.groupby(['match_id','inning']).sum()['total_runs']
    # convert this dataframe
    total_score_df = delivery.groupby(['match_id','inning']).sum()['total_runs'].reset_index()
```

```
total_score_df
```

	match_id	inning	total_runs
0	1	1	207
1	1	2	172
2	2	1	184
3	2	2	187
4	3	1	183
...	...	...	...
1523	11413	2	170
1524	11414	1	155
1525	11414	2	182
1526	11415	1	162
1527	11415	2	157

1528 rows × 3 columns

```
[8] # 1528 row because 756 matches and all matches 2 innings
```

```
[9] # inning 2 total run  remove because 2nd inning team chase 1st inning total run (filter the data )
```

```
[10] total_score_df=total_score_df[total_score_df['inning'] == 1]
```

`total_score_df`

	match_id	inning	total_runs
0	1	1	207
2	2	1	184
4	3	1	183
6	4	1	163
8	5	1	157
...	...	...	...
1518	11347	1	143
1520	11412	1	136
1522	11413	1	171
1524	11414	1	155
1526	11415	1	152

756 rows × 3 columns

```python
[12] # with matches dataframe merge total_score_df
     # match_id and id common both dataframe
```

```python
[13] match_df=match.merge(total_score_df[['match_id','total_runs']],left_on='id',right_on='match_id')
```

`match_df`

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets	player_of_match	venue	umpire1	umpire2	umpire3	match_id	total_runs
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	35	0	Yuvraj Singh	Rajiv Gandhi International Stadium, Uppal	AY Dandekar	NJ Llong	NaN	1	207
1	2	IPL-2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Rising Pune Supergiant	0	7	SPD Smith	Maharashtra Cricket Association Stadium	A Nand Kishore	S Ravi	NaN	2	184
2	3	IPL-2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	Kolkata Knight Riders	0	10	CA Lynn	Saurashtra Cricket Association Stadium	Nitin Menon	CK Nandan	NaN	3	183
3	4	IPL-2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	0	6	GJ Maxwell	Holkar Cricket Stadium	AK Chaudhary	C Shamshuddin	NaN	4	163
4	5	IPL-2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	15	0	KM Jadhav	M Chinnaswamy Stadium	NaN	NaN	NaN	5	157
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
751	11347	IPL-2019	Mumbai	05-05-2019	Kolkata Knight Riders	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0	9	HH Pandya	Wankhede Stadium	Nanda Kishore	O Nandan	S Ravi	11347	143
752	11412	IPL-2019	Chennai	07-05-2019	Chennai Super Kings	Mumbai Indians	Chennai Super Kings	bat	normal	0	Mumbai Indians	0	6	AS Yadav	M. A. Chidambaram Stadium	Nigel Llong	Nitin Menon	Ian Gould	11412	136
753	11413	IPL-2019	Visakhapatnam	08-05-2019	Sunrisers Hyderabad	Delhi Capitals	Delhi Capitals	field	normal	0	Delhi Capitals	0	2	RR Pant	ACA-VDCA Stadium	NaN	NaN	NaN	11413	171
754	11414	IPL-2019	Visakhapatnam	10-05-2019	Delhi Capitals	Chennai Super Kings	Chennai Super Kings	field	normal	0	Chennai Super Kings	0	6	F du Plessis	ACA-VDCA Stadium	Sundaram Ravi	Bruce Oxenford	Chettithody Shamshuddin	11414	155
755	11415	IPL-2019	Hyderabad	12-05-2019	Mumbai Indians	Chennai Super Kings	Mumbai Indians	bat	normal	0	Mumbai Indians	1	0	JJ Bumrah	Rajiv Gandhi Intl. Cricket Stadium	Nitin Menon	Ian Gould	Nigel Llong	11415	152

756 rows × 20 columns

```
[15] # preprocessing --> some teams not ipl play in this time eg-->Rising Pune Supergiant,gujrat titans.so only take matches jo team ab bhi khel rhi hai
```

```
[16] # all team name
     match_df['team1'].unique()
```

```
array(['Sunrisers Hyderabad', 'Mumbai Indians', 'Gujarat Lions',
       'Rising Pune Supergiant', 'Royal Challengers Bangalore',
       'Kolkata Knight Riders', 'Delhi Daredevils', 'Kings XI Punjab',
       'Chennai Super Kings', 'Rajasthan Royals', 'Deccan Chargers',
       'Kochi Tuskers Kerala', 'Pune Warriors', 'Rising Pune Supergiants',
       'Delhi Capitals'], dtype=object)
```

```
▶  # remove=Gujarat Lions ,Rising Pune Supergiant,Kochi Tuskers Kerala, Pune Warriors, Rising Pune Supergiants
   # Delhi Daredevils replace Delhi Capitals,Sunrisers Hyderabad replace Deccan Chargers
```

```
[18] teams=[
        'Sunrisers Hyderabad',
        'Mumbai Indians',
        'Royal Challengers Bangalore',
        'Kolkata Knight Riders',
        'Kings XI Punjab',
         'Chennai Super Kings',
        'Rajasthan Royals',
        'Delhi Capitals'
        ]
```

```
[19] match_df['team1'] = match_df['team1'].str.replace('Delhi Daredevils','Delhi Capitals')
     match_df['team2'] = match_df['team2'].str.replace('Delhi Daredevils','Delhi Capitals')

     match_df['team1'] = match_df['team1'].str.replace('Deccan Chargers','Sunrisers Hyderabad')
     match_df['team2'] = match_df['team2'].str.replace('Deccan Chargers','Sunrisers Hyderabad')
```

```
[20] # jo team play vhi team 1 and 2 column me put
     match_df = match_df[match_df['team1'].isin(teams)]
     match_df = match_df[match_df['team2'].isin(teams)]
```

```
[21] match_df.shape
```

```
(641, 20)
```

```
[22] # Duckworth-Lewis (due rain match cancel ) remove from column
```

```
[23] match_df['dl_applied'].value_counts()
```

```
0    626
1     15
Name: dl_applied, dtype: int64
```

```
[24] # so 15 match dl_applied remove
     match_df = match_df[match_df['dl_applied'] == 0]
```

```
▶  match_df
```

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets	player_of_match	venue	umpire1	umpire2	umpire3	match_id	total_runs
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	35	0	Yuvraj Singh	Rajiv Gandhi International Stadium, Uppal	AY Dandekar	NJ Llong	NaN	1	207
4	5	IPL-2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Capitals	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	15	0	KM Jadhav	M Chinnaswamy Stadium	NaN	NaN	NaN	5	157
6	7	IPL-2017	Mumbai	09-04-2017	Kolkata Knight Riders	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0	4	N Rana	Wankhede Stadium	Nitin Menon	CK Nandan	NaN	7	178
7	8	IPL-2017	Indore	10-04-2017	Royal Challengers Bangalore	Kings XI Punjab	Royal Challengers Bangalore	bat	normal	0	Kings XI Punjab	0	8	AR Patel	Holkar Cricket Stadium	AK Chaudhary	C Shamshuddin	NaN	8	148
9	10	IPL-2017	Mumbai	12-04-2017	Sunrisers Hyderabad	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0	4	JJ Bumrah	Wankhede Stadium	Nitin Menon	CK Nandan	NaN	10	158
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
751	11347	IPL-2019	Mumbai	05-05-2019	Kolkata Knight Riders	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0	9	HH Pandya	Wankhede Stadium	Nanda Kishore	O Nandan	S Ravi	11347	143
752	11412	IPL-2019	Chennai	07-05-2019	Chennai Super Kings	Mumbai Indians	Chennai Super Kings	bat	normal	0	Mumbai Indians	0	6	AS Yadav	M. A. Chidambaram Stadium	Nigel Llong	Nitin Menon	Ian Gould	11412	136
753	11413	IPL-2019	Visakhapatnam	08-05-2019	Sunrisers Hyderabad	Delhi Capitals	Delhi Capitals	field	normal	0	Delhi Capitals	0	2	RR Pant	ACA-VDCA Stadium	NaN	NaN	NaN	11413	171
754	11414	IPL-2019	Visakhapatnam	10-05-2019	Delhi Capitals	Chennai Super Kings	Chennai Super Kings	field	normal	0	Chennai Super Kings	0	6	F du Plessis	ACA-VDCA Stadium	Sundaram Ravi	Bruce Oxenford	Chettithody Shamshuddin	11414	155
755	11415	IPL-2019	Hyderabad	12-05-2019	Mumbai Indians	Chennai Super Kings	Mumbai Indians	bat	normal	0	Mumbai Indians	1	0	JJ Bumrah	Rajiv Gandhi Intl. Cricket Stadium	Nitin Menon	Ian Gould	Nigel Llong	11415	152

626 rows × 20 columns

+ Code   + Text

```
[26] # match_df se important column take and join with delivery dataframe because all feature extraction perform in delivery df
     # imp column match df se kon kon se le see screenhot me
```

```
[27] match_df = match_df[['match_id','city','winner','total_runs']]
```

```
[28] # join help of match_id
     delivery_df=match_df.merge(delivery,on='match_id')
```

```
delivery_df
# in delivery_df only row need inning value =2 because chase kerte time ye sab kerna hai .
```

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	bye_runs	legbye_runs	noball_runs	penalty_runs	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder
0	1	Hyderabad	Sunrisers Hyderabad	207	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	...	0	0	0	0	0	0	0	NaN	NaN	NaN
1	1	Hyderabad	Sunrisers Hyderabad	207	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	...	0	0	0	0	0	0	0	NaN	NaN	NaN
2	1	Hyderabad	Sunrisers Hyderabad	207	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	...	0	0	0	0	4	0	4	NaN	NaN	NaN
3	1	Hyderabad	Sunrisers Hyderabad	207	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	...	0	0	0	0	0	0	0	NaN	NaN	NaN
4	1	Hyderabad	Sunrisers Hyderabad	207	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	...	0	0	0	0	0	2	2	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	0	0	0	0	1	0	1	NaN	NaN	NaN
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	0	0	0	0	2	0	2	NaN	NaN	NaN
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	0	0	0	0	1	0	1	SR Watson	run out	KH Pandya
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	0	0	0	0	2	0	2	NaN	NaN	NaN
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	0	0	0	0	0	0	SN Thakur	lbw	NaN

149578 rows × 24 columns

+ Code   + Text

```
[30] delivery_df = delivery_df[delivery_df['inning'] == 2]
```

```
[31] delivery_df.shape
     (72413, 24)
```

```
delivery_df
```

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	bye_runs	legbye_runs	noball_runs	penalty_runs	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	0	0	0	0	1	0	1	NaN	NaN	NaN
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	0	0	0	0	0	NaN	NaN	NaN
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	0	0	0	0	0	NaN	NaN	NaN
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	0	0	0	0	2	0	2	NaN	NaN	NaN
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	0	0	0	0	4	0	4	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	0	0	0	0	1	0	1	NaN	NaN	NaN
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	0	0	0	0	2	0	2	NaN	NaN	NaN
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	0	0	0	0	1	0	1	SR Watson	run out	KH Pandya
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	0	0	0	0	2	0	2	NaN	NaN	NaN
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	0	0	0	0	0	0	SN Thakur	lbw	NaN

72413 rows × 24 columns

CO model ipl predictor ☆
File Edit View Insert Runtime Tools Help All changes saved

Comment Share ⚙
+ Code + Text
RAM
Disk
Editing

[33] # screenshot me first 3 column and total_run column se alag all column nikalne hai

[34] # runs_left,balls_left column find firstly
    # find all balls ke badd kitne total run hai

```
delivery_df.groupby('match_id').cumsum()['total_runs_y']
```

```
125      1
126      1
127      1
128      3
129      7
        ...
149573   152
149574   154
149575   155
149576   157
149577   157
Name: total_runs_y, Length: 72413, dtype: int64
```

[36] delivery_df['current_score'] = delivery_df.groupby('match_id').cumsum()['total_runs_y']

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

CO model ipl predictor ☆
File Edit View Insert Runtime Tools Help All changes saved

Comment Share ⚙
+ Code + Text
RAM
Disk
Editing

[37] delivery_df
    # current_score column add

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	legbye_runs	noball_runs	penalty_runs	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	0	0	0	1	0	1	NaN	NaN	NaN	1
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	0	0	0	0	NaN	NaN	NaN	1
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	0	0	0	0	NaN	NaN	NaN	1
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	0	0	0	2	0	2	NaN	NaN	NaN	3
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	0	0	0	4	0	4	NaN	NaN	NaN	7
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	0	0	0	1	0	1	NaN	NaN	NaN	152
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	0	0	0	2	0	2	NaN	NaN	NaN	154
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	0	0	0	1	0	1	SR Watson	run out	KH Pandya	155
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	0	0	0	2	0	2	NaN	NaN	NaN	157
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	0	0	0	0	0	SN Thakur	lbw	NaN	157

72413 rows × 25 columns

+ Code   + Text

[38] # runs_left= total_run or target_run or 1st inning run - current_score

[39] delivery_df['runs_left']=delivery_df['total_runs_x']-delivery_df['current_score']

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

delivery_df

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	noball_runs	penalty_runs	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score	runs_left
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	0	0	1	0	1	NaN	NaN	NaN	1	206
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	0	0	0	NaN	NaN	NaN	1	206
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	0	0	0	NaN	NaN	NaN	1	206
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	0	0	2	0	2	NaN	NaN	NaN	3	204
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	0	0	4	0	4	NaN	NaN	NaN	7	200
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	0	0	1	0	1	NaN	NaN	NaN	152	0
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	0	0	2	0	2	NaN	NaN	NaN	154	-2
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	0	0	1	0	1	SR Watson	run out	KH Pandya	155	-3
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	0	0	2	0	2	NaN	NaN	NaN	157	-5
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	0	0	0	0	SN Thakur	lbw	NaN	157	-5

72413 rows × 26 columns

+ Code   + Text

[41] # balls_left column find
delivery_df['balls_left'] = 126 - (delivery_df['over']*6 + delivery_df['ball'])

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

delivery_df

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	penalty_runs	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score	runs_left	balls_left
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	0	1	0	1	NaN	NaN	NaN	1	206	119
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	0	0	NaN	NaN	NaN	1	206	118
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	0	0	NaN	NaN	NaN	1	206	117
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	0	2	0	2	NaN	NaN	NaN	3	204	116
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	0	4	0	4	NaN	NaN	NaN	7	200	115
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	0	1	0	1	NaN	NaN	NaN	152	0	4
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	0	2	0	2	NaN	NaN	NaN	154	-2	3
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	0	1	0	1	SR Watson	run out	KH Pandya	155	-3	2
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	0	2	0	2	NaN	NaN	NaN	157	-5	1
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	0	0	0	SN Thakur	lbw	NaN	157	-5	0

72413 rows × 27 columns

```
[43] # find wicket left column
     #  player_dismissed column given which name of player if out if not out nan
```

```
# nan value replace 0
delivery_df['player_dismissed'] = delivery_df['player_dismissed'].fillna("0")
# if 0 no change if not 0 (player name ) make 1
delivery_df['player_dismissed'] = delivery_df['player_dismissed'].apply(lambda x:x if x == "0" else "1")
delivery_df['player_dismissed'] = delivery_df['player_dismissed'].astype('int')
wickets = delivery_df.groupby('match_id').cumsum()['player_dismissed'].values
delivery_df['wickets'] = 10 - wickets
delivery_df.head()
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:4: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  after removing the cwd from sys.path.
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:5: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  import sys
```

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score	runs_left	balls_left	wickets
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	1	0	1	0	NaN	NaN	1	206	119	10
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	0	0	NaN	NaN	1	206	118	10
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	0	0	NaN	NaN	1	206	117	10
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	2	0	2	0	NaN	NaN	3	204	116	10
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	4	0	4	0	NaN	NaN	7	200	115	10

5 rows × 28 columns

✓ 0s   completed at 6:14 PM

```
delivery_df.tail()
```

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score	runs_left	balls_left	wickets
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	1	0	1	0	NaN	NaN	152	0	4	5
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	2	0	2	0	NaN	NaN	154	-2	3	5
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	1	0	1	1	run out	KH Pandya	155	-3	2	4
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	2	0	2	0	NaN	NaN	157	-5	1	4
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	0	0	1	lbw	NaN	157	-5	0	3

5 rows × 28 columns

```
[46] delivery_df.head()
```

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	batsman_runs	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score	runs_left	balls_left	wickets
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	1	0	1	0	NaN	NaN	1	206	119	10
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	0	0	NaN	NaN	1	206	118	10
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	0	0	NaN	NaN	1	206	117	10
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	2	0	2	0	NaN	NaN	3	204	116	10
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	4	0	4	0	NaN	NaN	7	200	115	10

5 rows × 28 columns

+ Code + Text

RAM
Disk Editing

[47] # current run rate find column
```
# crr=runs/overs
delivery_df['crr']=delivery_df['current_score']*6/(120-delivery_df['balls_left'])
```

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:3: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  This is separate from the ipykernel package so we can avoid doing imports until

delivery_df

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	extra_runs	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score	runs_left	balls_left	wickets	crr
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	0	0	1	0	NaN	NaN	1	206	119	10	6.000000
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	0	0	NaN	NaN	1	206	118	10	3.000000
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	0	0	NaN	NaN	1	206	117	10	2.000000
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	0	2	0	0	NaN	NaN	3	204	116	10	4.500000
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	0	4	0	0	NaN	NaN	7	200	115	10	8.400000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	0	1	0	NaN	NaN	152	0	4	5	7.862069
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	0	2	0	NaN	NaN	154	-2	3	5	7.897436
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	0	1	1	run out	KH Pandya	155	-3	2	4	7.881356
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	0	2	0	NaN	NaN	157	-5	1	4	7.915966
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	0	1	lbw	NaN	157	-5	0	3	7.850000

72413 rows × 29 columns

+ Code + Text

RAM
Disk Editing

[49] # required run rate column find
```
delivery_df['rrr'] = (delivery_df['runs_left']*6)/delivery_df['balls_left']
```

/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

delivery_df

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	total_runs_y	player_dismissed	dismissal_kind	fielder	current_score	runs_left	balls_left	wickets	crr	rrr
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	1	0	NaN	NaN	1	206	119	10	6.000000	10.386555
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	0	NaN	NaN	1	206	118	10	3.000000	10.474576
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	0	NaN	NaN	1	206	117	10	2.000000	10.564103
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	2	0	NaN	NaN	3	204	116	10	4.500000	10.551724
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	4	0	NaN	NaN	7	200	115	10	8.400000	10.434783
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	1	0	NaN	NaN	152	0	4	5	7.862069	0.000000
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	2	0	NaN	NaN	154	-2	3	5	7.897436	-4.000000
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	1	1	run out	KH Pandya	155	-3	2	4	7.881356	-9.000000
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	2	0	NaN	NaN	157	-5	1	4	7.915966	-30.000000
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	0	1	lbw	NaN	157	-5	0	3	7.850000	-inf

72413 rows × 30 columns

+ Code + Text

```python
[51] # result column find
     # if winner==batting_team print 1 else 0 so make function
     def result(row):
         return 1 if row['batting_team']==row['winner'] else 0
```

```python
delivery_df['result']=delivery_df.apply(result,axis=1)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
```

```python
[53] delivery_df
```

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	player_dismissed	dismissal_kind	fielder	current_score	runs_left	balls_left	wickets	crr	rrr	result
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	0	NaN	NaN	1	206	119	10	6.000000	10.386555	0
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	0	NaN	NaN	1	206	118	10	3.000000	10.474576	0
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	0	NaN	NaN	1	206	117	10	2.000000	10.564103	0
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	0	NaN	NaN	3	204	116	10	4.500000	10.551724	0
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	0	NaN	NaN	7	200	115	10	8.400000	10.434783	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	0	NaN	NaN	152	0	4	5	7.862069	0.000000	0
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	0	NaN	NaN	154	-2	3	5	7.897436	-4.000000	0
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	1	run out	KH Pandya	155	-3	2	4	7.881356	-9.000000	0
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	0	NaN	NaN	157	-5	1	4	7.915966	-30.000000	0
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	1	lbw	NaN	157	-5	0	3	7.850000	-inf	0

72413 rows × 31 columns

```python
[54] final_df = delivery_df[['batting_team','bowling_team','city','runs_left','balls_left','wickets','total_runs_x','crr','rrr','result']]
```

```python
final_df
```

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr	result
125	Royal Challengers Bangalore	Sunrisers Hyderabad	Hyderabad	206	119	10	207	6.000000	10.386555	0
126	Royal Challengers Bangalore	Sunrisers Hyderabad	Hyderabad	206	118	10	207	3.000000	10.474576	0
127	Royal Challengers Bangalore	Sunrisers Hyderabad	Hyderabad	206	117	10	207	2.000000	10.564103	0
128	Royal Challengers Bangalore	Sunrisers Hyderabad	Hyderabad	204	116	10	207	4.500000	10.551724	0
129	Royal Challengers Bangalore	Sunrisers Hyderabad	Hyderabad	200	115	10	207	8.400000	10.434783	0
...	...	...	...	...	...	...	...	...	...	...
149573	Chennai Super Kings	Mumbai Indians	Hyderabad	0	4	5	152	7.862069	0.000000	0
149574	Chennai Super Kings	Mumbai Indians	Hyderabad	-2	3	5	152	7.897436	-4.000000	0
149575	Chennai Super Kings	Mumbai Indians	Hyderabad	-3	2	4	152	7.881356	-9.000000	0
149576	Chennai Super Kings	Mumbai Indians	Hyderabad	-5	1	4	152	7.915966	-30.000000	0
149577	Chennai Super Kings	Mumbai Indians	Hyderabad	-5	0	3	152	7.850000	-inf	0

72413 rows × 10 columns

```python
[56] # suffle -->koi bhi match ka ball khi bhi ja sakta hai row me
     final_df = final_df.sample(final_df.shape[0])
```

```python
[57] final_df
```

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr	result
42079	Deccan Chargers	Rajasthan Royals	Nagpur	3	2	2	159	7.932203	9.000000	0
413	Delhi Daredevils	Royal Challengers Bangalore	Bangalore	105	79	8	157	7.809756	7.974684	0
93898	Royal Challengers Bangalore	Kings XI Punjab	Bangalore	190	105	8	198	3.200000	10.857143	0
8119	Kolkata Knight Riders	Deccan Chargers	Kolkata	90	98	8	110	5.454545	5.510204	1
66633	Kings XI Punjab	Rajasthan Royals	Chandigarh	126	78	8	177	7.285714	9.692308	0
...	...	...	...	...	...	...	...	...	...	...
110885	Delhi Daredevils	Sunrisers Hyderabad	Raipur	69	38	6	163	6.878049	10.894737	0
53741	Deccan Chargers	Kolkata Knight Riders	Hyderabad	128	86	9	169	7.235294	8.930233	0
64986	Royal Challengers Bangalore	Kolkata Knight Riders	Kolkata	113	44	7	190	6.078947	15.409091	0
28133	Royal Challengers Bangalore	Mumbai Indians	Port Elizabeth	100	62	7	157	5.896552	9.677419	0
10988	Royal Challengers Bangalore	Delhi Daredevils	Delhi	128	74	7	191	8.217391	10.378378	0

72413 rows × 10 columns

```python
[58] # random row see
     final_df.sample()
```

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr	result
145442	Kings XI Punjab	Royal Challengers Bangalore	Bengaluru	25	1	3	213	9.478992	150.0	0

```python
[59] # nan values (missing values)
     final_df.isnull().sum()
```

```
batting_team      0
bowling_team      0
city            832
runs_left         0
balls_left        0
wickets           0
total_runs_x      0
crr               0
rrr               7
result            0
dtype: int64
```

CO ▲ model ipl predictor ☆
File Edit View Insert Runtime Tools Help   All changes saved
💬 Comment 👥 Share ⚙ 🧑
+ Code + Text   RAM █▮ ▾ ✎ Editing ∧
                                                                        Disk █▮

```
[60] # remove missing value rows
     final_df.dropna(inplace=True)
```

```
[61] final_df = final_df[final_df['balls_left'] != 0]
```

```
[62] # model building
     # firstly data split
```

```
[63] X = final_df.iloc[:,:-1] #all row and last column se aalag all column
     y = final_df.iloc[:,-1]  # all row and last column
     from sklearn.model_selection import train_test_split
     X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.2,random_state=1)
     # here 20 percent data use testing and 80 percent training
```

```
[64] X_train
```

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr
124333	Delhi Daredevils	Kolkata Knight Riders	Kolkata	113	70	8	206	11.160000	9.685714
83050	Kings XI Punjab	Sunrisers Hyderabad	Chandigarh	93	55	6	150	5.261538	10.145455
31126	Rajasthan Royals	Delhi Daredevils	Bloemfontein	31	11	1	150	6.550459	16.909091
59884	Mumbai Indians	Deccan Chargers	Visakhapatnam	123	90	8	138	3.000000	8.200000
116096	Mumbai Indians	Royal Challengers Bangalore	Mumbai	123	84	9	170	7.833333	8.785714
...	...	...	...	...	...	...	...	...	...
131173	Rajasthan Royals	Chennai Super Kings	Jaipur	47	17	6	188	8.213592	16.588235
51378	Chennai Super Kings	Mumbai Indians	Mumbai	26	10	2	164	7.527273	15.600000
30801	Deccan Chargers	Kings XI Punjab	Johannesburg	85	81	8	134	7.538462	6.296296
51828	Kings XI Punjab	Delhi Daredevils	Delhi	119	50	8	231	9.600000	14.280000
43459	Deccan Chargers	Chennai Super Kings	Nagpur	102	89	8	138	6.967742	6.876404

57073 rows × 9 columns

```
[65] # here 3 column string me hai so onehotencoding apply
```

CO ▲ model ipl predictor ☆
File Edit View Insert Runtime Tools Help   All changes saved
💬 Comment 👥 Share ⚙ 🧑
+ Code + Text   RAM █▮ ▾ ✎ Editing ∧
                                                                        Disk █▮

```
[66] from sklearn.compose import ColumnTransformer
     from sklearn.preprocessing import OneHotEncoder

     trf = ColumnTransformer([
         ('trf',OneHotEncoder(sparse=False,drop='first'),['batting_team','bowling_team','city'])
     ]
     ,remainder='passthrough')
```

```
[67] # pipeline make
     from sklearn.linear_model import LogisticRegression
     # from sklearn.ensemble import RandomForestClassifier
     from sklearn.pipeline import Pipeline
```

```
[68] # pipe = Pipeline(steps=[
     #     ('step1',trf),
     #     ('step2',RandomForestClassifier())
     # ])
     pipe = Pipeline(steps=[
         ('step1',trf),
         ('step2',LogisticRegression(solver='liblinear'))
     ])
```

```
[69] # model training
     pipe.fit(X_train,y_train)

     Pipeline(steps=[('step1',
                      ColumnTransformer(remainder='passthrough',
                                        transformers=[('trf',
                                                       OneHotEncoder(drop='first',
                                                                     sparse=False),
                                                       ['batting_team',
                                                        'bowling_team', 'city'])])),
                     ('step2', LogisticRegression(solver='liblinear'))])
```

CO ⬢ model ipl predictor ☆
File Edit View Insert Runtime Tools Help   All changes saved
💬 Comment 👥 Share ⚙
+ Code + Text   RAM ▮ Disk ▮ ▾   ✏ Editing ⌄

```
[70] # some value infinite so remove row .rrr infinite when balls 0 and some runs  need.so balls_left!=0 row take only
     X_train.describe()
```

	runs_left	balls_left	wickets	total_runs_x	crr	rrr
count	57073.000000	57073.000000	57073.000000	57073.000000	57073.000000	57073.000000
mean	92.679854	62.808684	7.548315	165.764022	7.448015	10.417316
std	49.959364	33.245806	2.127675	29.280412	2.275618	13.999837
min	-11.000000	-2.000000	0.000000	66.000000	0.000000	-510.000000
25%	53.000000	35.000000	6.000000	147.000000	6.260870	7.153846
50%	92.000000	63.000000	8.000000	165.000000	7.486726	8.884615
75%	130.000000	92.000000	9.000000	185.000000	8.700000	10.894737
max	249.000000	119.000000	10.000000	250.000000	36.000000	678.000000

```
[71] y_pred = pipe.predict(X_test)
```

```
[72] from sklearn.metrics import accuracy_score
     accuracy_score(y_test,y_pred)
     # 80 percent accuracy

     0.8027191814422875
```

```
[73] # if random forest algo use 99 percent accuracy
     # if puchne chasing team win or not use random forest .but hame chahiye all stage per dono team ke jitne ki probability kya hai use logistic regression.
```

```
[74] pipe.predict_proba(X_test)[10]# [match no ]

     array([0.98360382, 0.01639618])
```

```
[75] # logistic regression five best probality here.
```

```
[76] # ham ek match ko leker iss algo me bhejege 2nd inning and match situation analysis percentage  by percentage
```

```
[77] def match_summary(row):
         print("Batting Team-" + row['batting_team'] + " | Bowling Team-" + row['bowling_team'] + " | Target- " + str(row['total_runs_x']))
```

CO ⬢ model ipl predictor ☆
File Edit View Insert Runtime Tools Help   All changes saved
💬 Comment 👥 Share ⚙
+ Code + Text   RAM ▮ Disk ▮ ▾   ✏ Editing ⌄

```
[78] def match_progression(x_df,match_id,pipe):
         match = x_df[x_df['match_id'] == match_id]
         match = match[(match['ball'] == 6)]
         temp_df = match[['batting_team','bowling_team','city','runs_left','balls_left','wickets','total_runs_x','crr','rrr']].dropna()
         temp_df = temp_df[temp_df['balls_left'] != 0]
         result = pipe.predict_proba(temp_df)
         temp_df['lose'] = np.round(result.T[0]*100,1)
         temp_df['win'] = np.round(result.T[1]*100,1)
         temp_df['end_of_over'] = range(1,temp_df.shape[0]+1)

         target = temp_df['total_runs_x'].values[0]
         runs = list(temp_df['runs_left'].values)
         new_runs = runs[:]
         runs.insert(0,target)
         temp_df['runs_after_over'] = np.array(runs)[:-1] - np.array(new_runs)
         wickets = list(temp_df['wickets'].values)
         new_wickets = wickets[:]
         new_wickets.insert(0,10)
         wickets.append(0)
         w = np.array(wickets)
         nw = np.array(new_wickets)
         temp_df['wickets_in_over'] = (nw - w)[0:temp_df.shape[0]]

         print("Target-",target)
         temp_df = temp_df[['end_of_over','runs_after_over','wickets_in_over','lose','win']]
         return temp_df,target
```
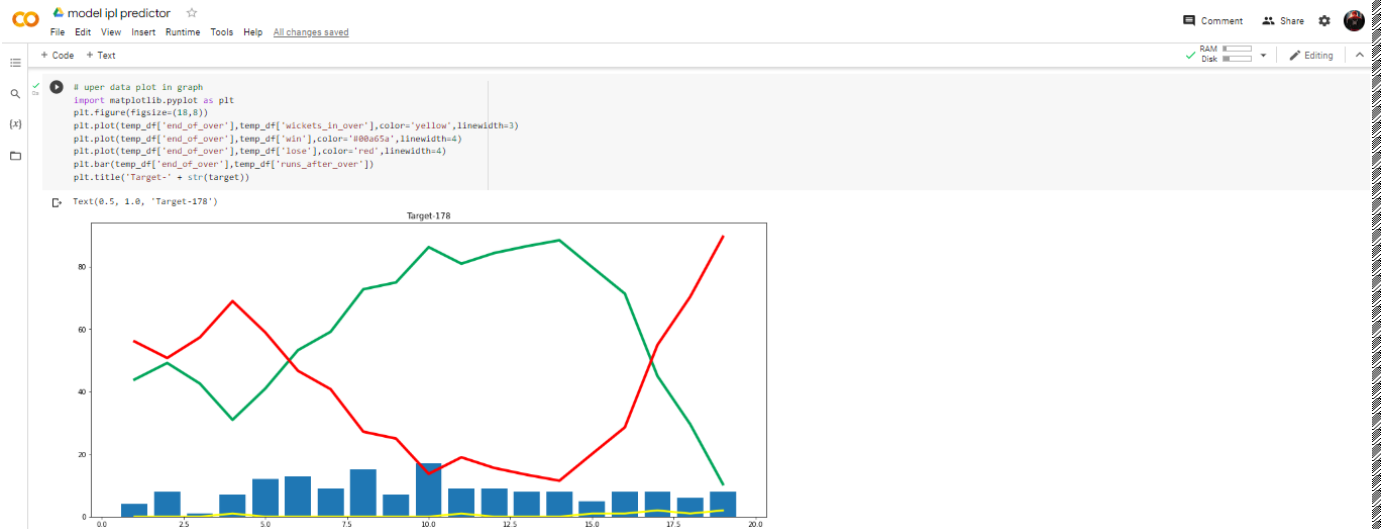
```
temp_df,target = match_progression(delivery_df,74,pipe)#(all match ,match no,pipe)
temp_df
```

Target- 178

	end_of_over	runs_after_over	wickets_in_over	lose	win
10459	1	4	0	56.1	43.9
10467	2	8	0	50.8	49.2
10473	3	1	0	57.4	42.6
10479	4	7	1	69.0	31.0
10485	5	12	0	59.0	41.0
10491	6	13	0	46.7	53.3
10497	7	9	0	40.8	59.2
10505	8	15	0	27.2	72.8
10511	9	7	0	25.0	75.0
10518	10	17	0	13.7	86.3
10524	11	9	1	19.0	81.0
10530	12	9	0	15.6	84.4
10536	13	8	0	13.4	86.6
10542	14	8	0	11.5	88.5
10548	15	5	1	20.1	79.9
10555	16	8	1	28.6	71.4
10561	17	8	2	55.0	45.0
10567	18	6	1	70.4	29.6
10573	19	8	2	89.6	10.4

```
# uper data plot in graph
import matplotlib.pyplot as plt
plt.figure(figsize=(18,8))
plt.plot(temp_df['end_of_over'],temp_df['wickets_in_over'],color='yellow',linewidth=3)
plt.plot(temp_df['end_of_over'],temp_df['win'],color='#00a65a',linewidth=4)
plt.plot(temp_df['end_of_over'],temp_df['lose'],color='red',linewidth=4)
plt.bar(temp_df['end_of_over'],temp_df['runs_after_over'])
plt.title('Target-' + str(target))
```

Text(0.5, 1.0, 'Target-178')

```
[81]  # matches study help of this
      # website make --> teams name ,all city name ,pipe object require
```

```
[82]  teams
```

```
['Sunrisers Hyderabad',
 'Mumbai Indians',
 'Royal Challengers Bangalore',
 'Kolkata Knight Riders',
 'Kings XI Punjab',
 'Chennai Super Kings',
 'Rajasthan Royals',
 'Delhi Capitals']
```

```
[83]  delivery_df['city'].unique()
```

```
array(['Hyderabad', 'Bangalore', 'Mumbai', 'Indore', 'Kolkata', 'Delhi',
       'Chandigarh', 'Jaipur', 'Chennai', 'Cape Town', 'Port Elizabeth',
       'Durban', 'Centurion', 'East London', 'Johannesburg', 'Kimberley',
       'Bloemfontein', 'Ahmedabad', 'Cuttack', 'Nagpur', 'Dharamsala',
       'Visakhapatnam', 'Pune', 'Raipur', 'Ranchi', 'Abu Dhabi',
       'Sharjah', nan, 'Mohali', 'Bengaluru'], dtype=object)
```

```
[84]  import pickle
      pickle.dump(pipe,open('pipe.pkl','wb'))
```

```
[85]  # streamlit is a library jiske use se easily ananlysis ke web application make . flask me more code write but in streamlit have inbuild function .pip install streamlit use install this.
```

```
# run code help of -->streamlit run file_name
```

```
[1]  ! pip install streamlit -q
```

```
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 9.2 MB 21.2 MB/s
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 237 kB 58.1 MB/s
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 182 kB 64.4 MB/s
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 78 kB 7.7 MB/s
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 164 kB 81.9 MB/s
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 4.7 MB 70.7 MB/s
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 63 kB 2.1 MB/s
                    ▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮▮ 51 kB 7.9 MB/s
         Building wheel for validators (setup.py) ... done
```

```python
%%writefile app.py
import streamlit as st
import pickle
import pandas as pd

teams = ['Sunrisers Hyderabad',
 'Mumbai Indians',
 'Royal Challengers Bangalore',
 'Kolkata Knight Riders',
 'Kings XI Punjab',
 'Chennai Super Kings',
 'Rajasthan Royals',
 'Delhi Capitals']

cities = ['Hyderabad', 'Bangalore', 'Mumbai', 'Indore', 'Kolkata', 'Delhi',
       'Chandigarh', 'Jaipur', 'Chennai', 'Cape Town', 'Port Elizabeth',
       'Durban', 'Centurion', 'East London', 'Johannesburg', 'Kimberley',
       'Bloemfontein', 'Ahmedabad', 'Cuttack', 'Nagpur', 'Dharamsala',
       'Visakhapatnam', 'Pune', 'Raipur', 'Ranchi', 'Abu Dhabi',
       'Sharjah', 'Mohali', 'Bengaluru']

pipe = pickle.load(open('pipe.pkl','rb'))


st.title('IPL Win Predictor')
from PIL import Image
img = Image.open('python.jpg')
st.image(img,caption='IPL')
col1, col2 = st.columns(2)
```

Executing (4s) Cell > system() > _system_compat() > _run_command() > _monitor_process() > _poll_process()

```python
with col1:
    batting_team = st.selectbox('Select the batting team',sorted(teams))
with col2:
    bowling_team = st.selectbox('Select the bowling team',sorted(teams))
selected_city=st.selectbox('Select host city',sorted(cities))
target = st.number_input('Target')
col3,col4,col5 = st.columns(3)
with col3:
    score = st.number_input('Score')
with col4:
    overs = st.number_input('Overs completed')
with col5:
    wickets = st.number_input('Wickets out')
if st.button('Predict Probability'):
    runs_left = target - score
    balls_left = 120 - (overs*6)
    wickets = 10 - wickets
    crr = score/overs
    rrr = (runs_left*6)/balls_left

    input_df = pd.DataFrame({'batting_team':[batting_team],'bowling_team':[bowling_team],'city':[selected_city],
    'runs_left':[runs_left],'balls_left':[balls_left],'wickets':[wickets],'total_runs_x':[target],'crr':[crr],'rrr':[rrr]})

    result = pipe.predict_proba(input_df)
    loss = result[0][0]
    win = result[0][1]
    st.header(batting_team + "- " + str(round(win*100)) + "%")
    st.header(bowling_team + "- " + str(round(loss*100)) + "%")
```

Writing app.py

```
!streamlit run app.py & npx localtunnel --port 8501
```

```
[..................] / rollbackFailedOptional: verb npm-session c8837ef60fd8f0b
Collecting usage statistics. To deactivate, set browser.gatherUsageStats to False.

npx: installed 22 in 3.512s
your url is: https://violet-lamps-wear-35-194-89-231.loca.lt

  You can now view your Streamlit app in your browser.

  Network URL: http://172.28.0.2:8501
  External URL: http://35.194.89.231:8501
```

REFERENCES

[1]. W3 School Website. Link: https://www.w3schools.com/

[2]. Mr. SHIVAM SINGHAL, who helped us with his guidance and knowledge.