IPL Win probability predictor Web App

December 12, 2022

[2]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

[3]:
```
matches = pd.read_csv('matches.csv')
```

[4]:
```
matches.head()
```

[4]:

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets	player_of_match	venue
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
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
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
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
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

```
          umpire1        umpire2 umpire3
0     AY Dandekar       NJ Llong     NaN
1  A Nand Kishore         S Ravi     NaN
2     Nitin Menon      CK Nandan     NaN
3    AK Chaudhary  C Shamshuddin     NaN
4             NaN            NaN     NaN
```

[5]:
```
delivery = pd.read_csv('deliveries.csv')
```

[6]:
```
delivery.head()
```

[6]:
```
   match_id  inning         batting_team                 bowling_team  over  \
0         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1
1         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1
2         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1
3         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1
4         1       1  Sunrisers Hyderabad  Royal Challengers Bangalore     1

   ball    batsman non_striker    bowler  is_super_over  ...  bye_runs  \
0     1  DA Warner    S Dhawan  TS Mills              0  ...         0
1     2  DA Warner    S Dhawan  TS Mills              0  ...         0
2     3  DA Warner    S Dhawan  TS Mills              0  ...         0
3     4  DA Warner    S Dhawan  TS Mills              0  ...         0
4     5  DA Warner    S Dhawan  TS Mills              0  ...         0

   legbye_runs  noball_runs  penalty_runs  batsman_runs  extra_runs  \
0            0            0             0             0           0
1            0            0             0             0           0
2            0            0             0             4           0
3            0            0             0             0           0
4            0            0             0             0           2

   total_runs  player_dismissed dismissal_kind fielder
0           0               NaN            NaN     NaN
1           0               NaN            NaN     NaN
2           4               NaN            NaN     NaN
3           0               NaN            NaN     NaN
4           2               NaN            NaN     NaN

[5 rows x 21 columns]
```

[7]:
```
matches.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 756 entries, 0 to 755
Data columns (total 18 columns):
 #   Column           Non-Null Count  Dtype
```

```
---  ------           --------------  -----
 0   id               756 non-null    int64
 1   Season           756 non-null    object
 2   city             749 non-null    object
 3   date             756 non-null    object
 4   team1            756 non-null    object
 5   team2            756 non-null    object
 6   toss_winner      756 non-null    object
 7   toss_decision    756 non-null    object
 8   result           756 non-null    object
 9   dl_applied       756 non-null    int64
 10  winner           752 non-null    object
 11  win_by_runs      756 non-null    int64
 12  win_by_wickets   756 non-null    int64
 13  player_of_match  752 non-null    object
 14  venue            756 non-null    object
 15  umpire1          754 non-null    object
 16  umpire2          754 non-null    object
 17  umpire3          119 non-null    object
dtypes: int64(4), object(14)
memory usage: 106.4+ KB
```

[8]:
```
delivery.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 179078 entries, 0 to 179077
Data columns (total 21 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   match_id          179078 non-null  int64
 1   inning            179078 non-null  int64
 2   batting_team      179078 non-null  object
 3   bowling_team      179078 non-null  object
 4   over              179078 non-null  int64
 5   ball              179078 non-null  int64
 6   batsman           179078 non-null  object
 7   non_striker       179078 non-null  object
 8   bowler            179078 non-null  object
 9   is_super_over     179078 non-null  int64
 10  wide_runs         179078 non-null  int64
 11  bye_runs          179078 non-null  int64
 12  legbye_runs       179078 non-null  int64
 13  noball_runs       179078 non-null  int64
 14  penalty_runs      179078 non-null  int64
 15  batsman_runs      179078 non-null  int64
 16  extra_runs        179078 non-null  int64
 17  total_runs        179078 non-null  int64
 18  player_dismissed  8834 non-null    object
```

```
 19  dismissal_kind    8834 non-null    object
 20  fielder           6448 non-null    object
dtypes: int64(13), object(8)
memory usage: 28.7+ MB
```

[9]:
```
matches.describe()
```

[9]:

	id	dl_applied	win_by_runs	win_by_wickets
count	756.000000	756.000000	756.000000	756.000000
mean	1792.178571	0.025132	13.283069	3.350529
std	3464.478148	0.156630	23.471144	3.387963
min	1.000000	0.000000	0.000000	0.000000
25%	189.750000	0.000000	0.000000	0.000000
50%	378.500000	0.000000	0.000000	4.000000
75%	567.250000	0.000000	19.000000	6.000000
max	11415.000000	1.000000	146.000000	10.000000

[10]:
```
delivery.describe()
```

[10]:

	match_id	inning	over	ball	\
count	179078.000000	179078.000000	179078.000000	179078.000000	
mean	1802.252957	1.482952	10.162488	3.615587	
std	3472.322805	0.502074	5.677684	1.806966	
min	1.000000	1.000000	1.000000	1.000000	
25%	190.000000	1.000000	5.000000	2.000000	
50%	379.000000	1.000000	10.000000	4.000000	
75%	567.000000	2.000000	15.000000	5.000000	
max	11415.000000	5.000000	20.000000	9.000000	

	is_super_over	wide_runs	bye_runs	legbye_runs	\
count	179078.000000	179078.000000	179078.000000	179078.000000	
mean	0.000452	0.036721	0.004936	0.021136	
std	0.021263	0.251161	0.116480	0.194908	
min	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	0.000000	0.000000	0.000000	
50%	0.000000	0.000000	0.000000	0.000000	
75%	0.000000	0.000000	0.000000	0.000000	
max	1.000000	5.000000	4.000000	5.000000	

	noball_runs	penalty_runs	batsman_runs	extra_runs	\
count	179078.000000	179078.000000	179078.000000	179078.000000	
mean	0.004183	0.000056	1.246864	0.067032	
std	0.070492	0.016709	1.608270	0.342553	
min	0.000000	0.000000	0.000000	0.000000	
25%	0.000000	0.000000	0.000000	0.000000	
50%	0.000000	0.000000	1.000000	0.000000	
75%	0.000000	0.000000	1.000000	0.000000	
max	5.000000	5.000000	7.000000	7.000000	

```
          total_runs
count  179078.000000
mean        1.313897
std         1.605422
min         0.000000
25%         0.000000
50%         1.000000
75%         1.000000
max        10.000000
```

1 Data Cleaning

[11]:
```
total_score_df = delivery.groupby(['match_id','inning']).sum()['total_runs'].
↪reset_index()
```

[12]:
```
firstinnings_total = total_score_df[total_score_df['inning']==1]
```

[13]:
```
firstinnings_total.head()
```

[13]:
```
   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
```

[14]:
```
match_df = matches.
↪merge(firstinnings_total[['match_id','total_runs']],left_on='id',right_on='match_id')
```

[15]:
```
match_df['team1'].unique()
```

[15]:
```
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)
```

[16]:
```
teams = ['Sunrisers Hyderabad', 'Mumbai Indians',
         'Royal Challengers Bangalore','Kolkata Knight Riders',
        'Chennai Super Kings', 'Rajasthan Royals',
        'Kings XI Punjab','Delhi Capitals']
```

[17]:
```
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')
```

[18]:
```
match_df = match_df[match_df['team1'].isin(teams)]
match_df = match_df[match_df['team2'].isin(teams)]
```

[19]:
```
match_df['team1'].unique()
```

[19]:
```
array(['Sunrisers Hyderabad', 'Royal Challengers Bangalore',
       'Kolkata Knight Riders', 'Kings XI Punjab', 'Delhi Capitals',
       'Mumbai Indians', 'Chennai Super Kings', 'Rajasthan Royals'],
      dtype=object)
```

[20]:
```
match_df = match_df[match_df['dl_applied']==0]
```

[21]:
```
match_df.head()
```

[21]:

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	win_by_runs	win_by_wickets	player_of_match	venue
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
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
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
7	8	IPL-2017	Indore	10-04-2017	Royal Challengers Bangalore	Kings XI Punjab	Royal Challengers Bangalore	bat	normal	0	Kings XI Punjab	0	0		
9	10	IPL-2017	Mumbai	12-04-2017	Sunrisers Hyderabad	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0	0		

```
7               8        AR Patel                      Holkar Cricket Stadium
9               4       JJ Bumrah                            Wankhede Stadium

        umpire1        umpire2 umpire3  match_id  total_runs
0   AY Dandekar       NJ Llong     NaN         1         207
4           NaN            NaN     NaN         5         157
6   Nitin Menon      CK Nandan     NaN         7         178
7  AK Chaudhary  C Shamshuddin     NaN         8         148
9   Nitin Menon      CK Nandan     NaN        10         158
```

[22]:
```
delivery_df = match_df.merge(delivery,left_on='id',right_on='match_id')
```

[23]:
```
delivery_df = delivery_df[delivery_df['inning']==2]
```

[24]:
```
delivery_df.head()
```

[24]:
```
     id    Season       city        date               team1  \
125   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
126   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
127   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
128   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
129   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad

                          team2                  toss_winner toss_decision  \
125  Royal Challengers Bangalore  Royal Challengers Bangalore         field
126  Royal Challengers Bangalore  Royal Challengers Bangalore         field
127  Royal Challengers Bangalore  Royal Challengers Bangalore         field
128  Royal Challengers Bangalore  Royal Challengers Bangalore         field
129  Royal Challengers Bangalore  Royal Challengers Bangalore         field

     result  dl_applied  ... bye_runs  legbye_runs  noball_runs penalty_runs  \
125  normal           0  ...        0            0            0            0
126  normal           0  ...        0            0            0            0
127  normal           0  ...        0            0            0            0
128  normal           0  ...        0            0            0            0
129  normal           0  ...        0            0            0            0

    batsman_runs extra_runs total_runs_y player_dismissed  dismissal_kind  \
125            1          0            1              NaN             NaN
126            0          0            0              NaN             NaN
127            0          0            0              NaN             NaN
128            2          0            2              NaN             NaN
129            4          0            4              NaN             NaN

     fielder
125      NaN
126      NaN
```

```
127      NaN
128      NaN
129      NaN

[5 rows x 41 columns]
```

[25]:
```
delivery_df['current_score'] = delivery_df.groupby('id').cumsum()['total_runs_y']
```

[26]:
```
delivery_df['runs_left'] =␣
↪delivery_df['total_runs_x']+1-delivery_df['current_score']
```

[27]:
```
delivery_df.head()
```

[27]:
```
     id    Season       city        date                team1  \
125   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
126   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
127   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
128   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
129   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad

                          team2                 toss_winner toss_decision  \
125  Royal Challengers Bangalore  Royal Challengers Bangalore         field
126  Royal Challengers Bangalore  Royal Challengers Bangalore         field
127  Royal Challengers Bangalore  Royal Challengers Bangalore         field
128  Royal Challengers Bangalore  Royal Challengers Bangalore         field
129  Royal Challengers Bangalore  Royal Challengers Bangalore         field

     result  dl_applied  ... noball_runs  penalty_runs  batsman_runs  \
125  normal           0  ...           0             0             1
126  normal           0  ...           0             0             0
127  normal           0  ...           0             0             0
128  normal           0  ...           0             0             2
129  normal           0  ...           0             0             4

    extra_runs total_runs_y player_dismissed dismissal_kind fielder  \
125          0            1              NaN            NaN     NaN
126          0            0              NaN            NaN     NaN
127          0            0              NaN            NaN     NaN
128          0            2              NaN            NaN     NaN
129          0            4              NaN            NaN     NaN

     current_score  runs_left
125              1        207
126              1        207
127              1        207
128              3        205
129              7        201
```

```
[5 rows x 43 columns]
```

[28]:
```
delivery_df['balls_left'] = 120 - ((delivery_df['over']-1)*6 +␣
↪delivery_df['ball'])
```

[29]:
```
delivery_df.head()
```

[29]:
```
      id    Season       city        date                team1  \
125    1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
126    1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
127    1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
128    1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
129    1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad

                          team2                 toss_winner toss_decision  \
125  Royal Challengers Bangalore  Royal Challengers Bangalore         field
126  Royal Challengers Bangalore  Royal Challengers Bangalore         field
127  Royal Challengers Bangalore  Royal Challengers Bangalore         field
128  Royal Challengers Bangalore  Royal Challengers Bangalore         field
129  Royal Challengers Bangalore  Royal Challengers Bangalore         field

     result  dl_applied  ... penalty_runs  batsman_runs  extra_runs  \
125  normal           0  ...            0             1           0
126  normal           0  ...            0             0           0
127  normal           0  ...            0             0           0
128  normal           0  ...            0             2           0
129  normal           0  ...            0             4           0

    total_runs_y player_dismissed dismissal_kind fielder current_score  \
125            1              NaN            NaN     NaN             1
126            0              NaN            NaN     NaN             1
127            0              NaN            NaN     NaN             1
128            2              NaN            NaN     NaN             3
129            4              NaN            NaN     NaN             7

     runs_left  balls_left
125        207         119
126        207         118
127        207         117
128        205         116
129        201         115

[5 rows x 44 columns]
```

[30]:
```
delivery_df['player_dismissed'] = delivery_df['player_dismissed'].fillna('0')
```

```
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('id').cumsum()['player_dismissed'].values
delivery_df['wickets'] = 10-wickets
```

[31]:
```
delivery_df.tail()
```

[31]:
```
           id    Season       city        date           team1  \
149573  11415  IPL-2019  Hyderabad  12-05-2019  Mumbai Indians
149574  11415  IPL-2019  Hyderabad  12-05-2019  Mumbai Indians
149575  11415  IPL-2019  Hyderabad  12-05-2019  Mumbai Indians
149576  11415  IPL-2019  Hyderabad  12-05-2019  Mumbai Indians
149577  11415  IPL-2019  Hyderabad  12-05-2019  Mumbai Indians

                      team2     toss_winner toss_decision  result  dl_applied  \
149573  Chennai Super Kings  Mumbai Indians           bat  normal           0
149574  Chennai Super Kings  Mumbai Indians           bat  normal           0
149575  Chennai Super Kings  Mumbai Indians           bat  normal           0
149576  Chennai Super Kings  Mumbai Indians           bat  normal           0
149577  Chennai Super Kings  Mumbai Indians           bat  normal           0

        ... batsman_runs  extra_runs  total_runs_y player_dismissed  \
149573  ...            1           0             1                0
149574  ...            2           0             2                0
149575  ...            1           0             1                1
149576  ...            2           0             2                0
149577  ...            0           0             0                1

       dismissal_kind    fielder current_score runs_left  balls_left  wickets
149573            NaN        NaN           152         1           4        5
149574            NaN        NaN           154        -1           3        5
149575        run out  KH Pandya           155        -2           2        4
149576            NaN        NaN           157        -4           1        4
149577            lbw        NaN           157        -4           0        3

[5 rows x 45 columns]
```

[32]:
```
#crr = runs/overs
delivery_df['crr'] = delivery_df['current_score']*6/
 ↪(120-delivery_df['balls_left'])
```

[33]:
```
delivery_df.head()
```

[33]:
```
     id    Season       city        date                team1  \
125   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
126   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
```

```
127   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
128   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad
129   1  IPL-2017  Hyderabad  05-04-2017  Sunrisers Hyderabad

                          team2                 toss_winner toss_decision  \
125  Royal Challengers Bangalore  Royal Challengers Bangalore         field
126  Royal Challengers Bangalore  Royal Challengers Bangalore         field
127  Royal Challengers Bangalore  Royal Challengers Bangalore         field
128  Royal Challengers Bangalore  Royal Challengers Bangalore         field
129  Royal Challengers Bangalore  Royal Challengers Bangalore         field

     result  dl_applied  ... extra_runs  total_runs_y  player_dismissed  \
125  normal           0  ...          0             1                 0
126  normal           0  ...          0             0                 0
127  normal           0  ...          0             0                 0
128  normal           0  ...          0             2                 0
129  normal           0  ...          0             4                 0

    dismissal_kind fielder current_score runs_left balls_left  wickets  crr
125            NaN     NaN             1       207        119       10  6.0
126            NaN     NaN             1       207        118       10  3.0
127            NaN     NaN             1       207        117       10  2.0
128            NaN     NaN             3       205        116       10  4.5
129            NaN     NaN             7       201        115       10  8.4

[5 rows x 46 columns]
```

[34]:
```
#rrr = runs left/  balls left
delivery_df['rrr'] = delivery_df['runs_left']*6/delivery_df['balls_left']
```

[35]:
```
def result(row):
    if row['batting_team'] == row['winner']:
        return 1
    else:
        return 0
```

[36]:
```
delivery_df['result'] = delivery_df.apply(result,axis=1)
```

[37]:
```
delivery_df.columns
```

[37]:
```
Index(['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_x', 'total_runs_x', 'match_id_y', 'inning',
       'batting_team', 'bowling_team', 'over', 'ball', 'batsman',
       'non_striker', 'bowler', 'is_super_over', 'wide_runs', 'bye_runs',
       'legbye_runs', 'noball_runs', 'penalty_runs', 'batsman_runs',
```

```
       'extra_runs', 'total_runs_y', 'player_dismissed', 'dismissal_kind',
       'fielder', 'current_score', 'runs_left', 'balls_left', 'wickets', 'crr',
       'rrr'],
      dtype='object')
```

[38]:
```
final_df = delivery_df[['batting_team','bowling_team','city','runs_left',
                        ␣
 ↪'balls_left','wickets','total_runs_x','crr','rrr','result']]
```

[39]:
```
final_df.head()
```

[39]:
```
                  batting_team         bowling_team       city  runs_left  \
125  Royal Challengers Bangalore  Sunrisers Hyderabad  Hyderabad        207
126  Royal Challengers Bangalore  Sunrisers Hyderabad  Hyderabad        207
127  Royal Challengers Bangalore  Sunrisers Hyderabad  Hyderabad        207
128  Royal Challengers Bangalore  Sunrisers Hyderabad  Hyderabad        205
129  Royal Challengers Bangalore  Sunrisers Hyderabad  Hyderabad        201

     balls_left  wickets  total_runs_x  crr        rrr  result
125         119       10           207  6.0  10.436975       0
126         118       10           207  3.0  10.525424       0
127         117       10           207  2.0  10.615385       0
128         116       10           207  4.5  10.603448       0
129         115       10           207  8.4  10.486957       0
```

[40]:
```
final_df = final_df.sample(final_df.shape[0])
```

[41]:
```
final_df.sample()
```

[41]:
```
           batting_team    bowling_team   city  runs_left  balls_left  \
35518  Delhi Daredevils  Mumbai Indians  Delhi        100          27

       wickets  total_runs_x       crr        rrr  result
35518        2           218  7.677419  22.222222       0
```

[42]:
```
final_df.isnull().sum()
```

[42]:
```
batting_team      0
bowling_team      0
city            832
runs_left         0
balls_left        0
wickets           0
total_runs_x      0
crr               0
rrr               5
result            0
dtype: int64
```

[43]:
```
final_df.dropna(inplace=True)
```

[44]:
```
final_df.describe()
```

[44]:
```
          runs_left    balls_left       wickets  total_runs_x           crr  \
count  71576.000000  71576.000000  71576.000000  71576.000000  71576.000000
mean      93.383313     62.673159      7.537904    165.767841      7.449324
std       50.058881     33.398497      2.139570     29.291819      2.277979
min      -15.000000     -2.000000      0.000000     65.000000      0.000000
25%       54.000000     35.000000      6.000000    147.000000      6.272727
50%       93.000000     63.000000      8.000000    165.000000      7.500000
75%      131.000000     92.000000      9.000000    185.000000      8.696629
max      250.000000    119.000000     10.000000    250.000000     42.000000

                rrr        result
count  7.157600e+04  71576.000000
mean            NaN      0.525246
std             NaN      0.499366
min            -inf      0.000000
25%    7.272727e+00      0.000000
50%    9.000000e+00      1.000000
75%    1.106422e+01      1.000000
max             inf      1.000000
```

[45]:
```
final_df.isnull().sum()
```

[45]:
```
batting_team    0
bowling_team    0
city            0
runs_left       0
balls_left      0
wickets         0
total_runs_x    0
crr             0
rrr             0
result          0
dtype: int64
```

[46]:
```
final_df = final_df[final_df['balls_left']!=0]
```

2 Model Building

[47]:
```
from sklearn.model_selection import train_test_split
```

[48]:
```
X = final_df.drop(columns=['result'],axis=1)
y = final_df['result']
```

[49]:
```
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size = 0.25)
```

[50]:
```
y_train.isnull().sum()
```

[50]:
```
0
```

[51]:
```
from sklearn.compose import ColumnTransformer
from sklearn.preprocessing import OneHotEncoder
```

[52]:
```
trf = ColumnTransformer(transformers = [
    ␣
 ↪('trf1',OneHotEncoder(sparse=False,drop='first'),['batting_team','bowling_team','city'])
    ], remainder='passthrough')
```

[53]:
```
from sklearn.linear_model import LogisticRegression
```

[54]:
```
from sklearn.pipeline import Pipeline
```

[55]:
```
pipe = Pipeline(steps =␣
 ↪[('step1',trf),('step2',LogisticRegression(solver='liblinear'))])
```

[56]:
```
X_train.describe()
```

[56]:

	runs_left	balls_left	wickets	total_runs_x	crr	rrr
count	53506.000000	53506.000000	53506.000000	53506.000000	53506.000000	53506.000000
mean	93.808937	63.023324	7.559507	165.749935	7.444368	10.496812
std	49.939261	33.242529	2.121393	29.273488	2.277195	13.437271
min	-10.000000	-2.000000	0.000000	65.000000	0.000000	-516.000000
25%	54.000000	35.000000	6.000000	147.000000	6.260870	7.272727
50%	93.000000	64.000000	8.000000	165.000000	7.482353	9.000000
75%	131.000000	92.000000	9.000000	185.000000	8.696385	11.013439
max	250.000000	119.000000	10.000000	250.000000	42.000000	684.000000

[57]:
```
pipe.fit(X_train,y_train)
```

[57]:
```
Pipeline(steps=[('step1',
                 ColumnTransformer(remainder='passthrough',
                                   transformers=[('trf1',
```

```
                                                  OneHotEncoder(drop='first',
                                                                sparse=False),
                                                  ['batting_team',
                                                   'bowling_team', 'city'])])),
                  ('step2', LogisticRegression(solver='liblinear'))])
```

[58]:
```
X_train.head()
```

[58]:

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr
122392	Sunrisers Hyderabad	Rajasthan Royals	Hyderabad	54	71	9	133	9.795918	4.563380
132291	Royal Challengers Bangalore	Kings XI Punjab	Indore	48	94	10	93	10.615385	3.063830
73146	Chennai Super Kings	Mumbai Indians	Chennai	135	106	9	148	6.000000	7.641509
135893	Mumbai Indians	Delhi Capitals	Mumbai	127	59	7	219	9.147541	12.915254
144450	Sunrisers Hyderabad	Kolkata Knight Riders	Hyderabad	22	42	9	164	11.000000	3.142857

[59]:
```
pred = pipe.predict(X_test)
```

[60]:
```
from sklearn.metrics import accuracy_score, confusion_matrix
```

[62]:
```
accuracy_score(pred,y_test)
```

[62]:
```
0.8023099349629962
```

[66]:
```
X_test.head(10)
```

[66]:

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr
93150	Kolkata Knight Riders	Delhi Daredevils	Delhi	160	117	10	160	2.000000	8.205128
78253	Chennai Super Kings	Sunrisers Hyderabad	Chennai	115	91	10	159	9.310345	7.582418
94664	Rajasthan Royals	Royal Challengers Bangalore	Bangalore	129	68	8	190	7.153846	11.382353
105126	Kolkata Knight Riders	Delhi Daredevils	Delhi						
90319	Delhi Daredevils	Mumbai Indians	Sharjah						
58698	Delhi Daredevils	Kolkata Knight Riders	Kolkata						
62278	Rajasthan Royals	Deccan Chargers	Jaipur						
112489	Chennai Super Kings	Kings XI Punjab	Chandigarh						
50838	Delhi Daredevils	Deccan Chargers	Delhi						
40591	Deccan Chargers	Kolkata Knight Riders	Kolkata						

```
105126          10          17          7           146    7.980583    3.529412
90319           45          41          8           125    6.151899    6.585366
58698           91         111         10            97    4.666667    4.918919
62278          133          82          9           196   10.105263    9.731707
112489          19          28          7           130    7.304348    4.071429
50838           76          43          6           168    7.246753   10.604651
40591           57          29          7           181    8.241758   11.793103
```

[71]:
```
pipe.predict_proba(X_test)[0]
```

[71]:
```
array([0.33279468, 0.66720532])
```

[91]:
```
teams
```

[91]:
```
['Sunrisers Hyderabad',
 'Mumbai Indians',
 'Royal Challengers Bangalore',
 'Kolkata Knight Riders',
 'Chennai Super Kings',
 'Rajasthan Royals',
 'Kings XI Punjab',
 'Delhi Capitals']
```

[92]:
```
delivery_df['city'].unique()
```

[92]:
```
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)
```

[65]:
```
import pickle
```

[69]:
```
pickle.dump(pipe,open('ipl_pipe.pkl','wb'))
```

[ ]:
```

```