MATCH

In [295]:
```
import numpy as np
import pandas as pd
```

In [296]:
```
match = pd.read_csv('matches.csv')
delivery = pd.read_csv('deliveries.csv')
```

In [297]:
```
match.head()
```

Out[297]:

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	winner	wi
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sunrisers Hyderabad	
1	2	IPL-2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Rising Pune Supergiant	
2	3	IPL-2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	Kolkata Knight Riders	
3	4	IPL-2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	
4	5	IPL-2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	

In [298]:
```
match.shape
```

Out[298]: (756, 18)

In [299]:
```
delivery.head()
```

Out[299]:

	match_id	inning	batting_team	bowling_team	over	ball	batsman	non_striker	bowler	is_super_over	...	bye_runs
0	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	1	DA Warner	S Dhawan	TS Mills	0	...	0
1	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	2	DA Warner	S Dhawan	TS Mills	0	...	0
2	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	3	DA Warner	S Dhawan	TS Mills	0	...	0
3	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	4	DA Warner	S Dhawan	TS Mills	0	...	0
4	1	1	Sunrisers Hyderabad	Royal Challengers Bangalore	1	5	DA Warner	S Dhawan	TS Mills	0	...	0

5 rows × 21 columns

In [300]:
```
total_score_df = delivery.groupby(['match_id','inning']).sum()['total_runs'].reset_index()
```

In [301]:
```
total_score_df = total_score_df[total_score_df['inning'] == 1]
```

In [302]:
```
total_score_df
```

Out[302]:

	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

In [303]:
```
match_df = match.merge(total_score_df[['match_id','total_runs']],left_on='id',right_on='match_id'
```

In [304]:
```
match_df
```

Out[304]:

	id	Season	city	date	team1	team2	toss_winner	toss_decision	result	dl_applied	w
0	1	IPL-2017	Hyderabad	05-04-2017	Sunrisers Hyderabad	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Sur Hyde
1	2	IPL-2017	Pune	06-04-2017	Mumbai Indians	Rising Pune Supergiant	Rising Pune Supergiant	field	normal	0	Supe
2	3	IPL-2017	Rajkot	07-04-2017	Gujarat Lions	Kolkata Knight Riders	Kolkata Knight Riders	field	normal	0	K I I
3	4	IPL-2017	Indore	08-04-2017	Rising Pune Supergiant	Kings XI Punjab	Kings XI Punjab	field	normal	0	Ki F
4	5	IPL-2017	Bangalore	08-04-2017	Royal Challengers Bangalore	Delhi Daredevils	Royal Challengers Bangalore	bat	normal	0	Challe Ban
...	...	...	...	...	...	...	...	...	...	...	
751	11347	IPL-2019	Mumbai	05-05-2019	Kolkata Knight Riders	Mumbai Indians	Mumbai Indians	field	normal	0	M Ir
752	11412	IPL-2019	Chennai	07-05-2019	Chennai Super Kings	Mumbai Indians	Chennai Super Kings	bat	normal	0	M Ir
753	11413	IPL-2019	Visakhapatnam	08-05-2019	Sunrisers Hyderabad	Delhi Capitals	Delhi Capitals	field	normal	0	Ca
754	11414	IPL-2019	Visakhapatnam	10-05-2019	Delhi Capitals	Chennai Super Kings	Chennai Super Kings	field	normal	0	Ch
755	11415	IPL-2019	Hyderabad	12-05-2019	Mumbai Indians	Chennai Super Kings	Mumbai Indians	bat	normal	0	M Ir

756 rows × 20 columns

In [305]:
```
match_df['team1'].unique()
```

Out[305]:
```
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)
```

In [306]:
```
teams = [
    'Sunrisers Hyderabad',
    'Mumbai Indians',
    'Royal Challengers Bangalore',
    'Kolkata Knight Riders',
    'Kings XI Punjab',
    'Chennai Super Kings',
    'Rajasthan Royals',
    'Delhi Capitals'
]
```

In [307]:
```
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')
```

In [308]:
```
match_df = match_df[match_df['team1'].isin(teams)]
match_df = match_df[match_df['team2'].isin(teams)]
```

In [309]:
```
match_df.shape
```

Out[309]: (641, 20)

In [310]:
```
match_df = match_df[match_df['dl_applied'] == 0]
```

In [311]:
```
match_df = match_df[['match_id','city','winner','total_runs']]
```

In [312]:
```
delivery_df = match_df.merge(delivery,on='match_id')
```

In [313]:
```
delivery_df = delivery_df[delivery_df['inning'] == 2]
```

In [314]:
```
delivery_df
```

Out[314]:

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	bye
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	
...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	

72413 rows × 24 columns

In [315]:
```
delivery_df['current_score'] = delivery_df.groupby('match_id')['total_runs_y'].cumsum()
```

In [316]:
```
delivery_df['runs_left'] = delivery_df['total_runs_x'] - delivery_df['current_score']
```

In [317]:
```
delivery_df['balls_left'] = 126 - (delivery_df['over']*6 + delivery_df['ball'])
```

In [318]:
```
delivery_df
```

129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	
...	...	...	...	...	...	...	...	...	...	...	...	...
149573	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	2	RA Jadeja	...	
149574	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	3	SR Watson	...	
149575	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	4	SR Watson	...	
149576	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	5	SN Thakur	...	
149577	11415	Hyderabad	Mumbai Indians	152	2	Chennai Super Kings	Mumbai Indians	20	6	SN Thakur	...	

In [319]:
```
delivery_df['player_dismissed'] = delivery_df['player_dismissed'].apply(lambda x: 0 if x == "0" e

# Calculate cumulative sum within each group
delivery_df['wickets'] = delivery_df.groupby('match_id')['player_dismissed'].cumsum()

# Calculate remaining wickets (assuming a total of 10 wickets in a match)
delivery_df['remaining_wickets'] = 10 - delivery_df['wickets']

# Display the DataFrame
delivery_df.head()
```

Out[319]:

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	extra_r
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	

5 rows × 29 columns

In [320]:
```
delivery_df.head()
```

Out[320]:

	match_id	city	winner	total_runs_x	inning	batting_team	bowling_team	over	ball	batsman	...	extra_r
125	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	1	CH Gayle	...	
126	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	2	Mandeep Singh	...	
127	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	3	Mandeep Singh	...	
128	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	4	Mandeep Singh	...	
129	1	Hyderabad	Sunrisers Hyderabad	207	2	Royal Challengers Bangalore	Sunrisers Hyderabad	1	5	Mandeep Singh	...	

5 rows × 29 columns

In [321]:
```
delivery_df['crr'] = (delivery_df['current_score']*6)/(120 - delivery_df['balls_left'])
```

In [322]:
```
delivery_df['rrr'] = (delivery_df['runs_left']*6)/delivery_df['balls_left']
```

In [323]:
```
def result(row):
    return 1 if row['batting_team'] == row['winner'] else 0
```

In [324]:
```
delivery_df['result'] = delivery_df.apply(result,axis=1)
```

In [325]:
```
l_df = delivery_df[['batting_team','bowling_team','city','runs_left','balls_left','wickets','total
```

In [326]:
```
final_df = final_df.sample(final_df.shape[0])
```

In [327]:
```
final_df.sample()
```

Out[327]:

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr	result
30599	Deccan Chargers	Kolkata Knight Riders	Johannesburg	58	34	87	160	7.116279	10.235294	1

In [328]:
```
final_df.dropna(inplace=True)
```

In [329]:
```
final_df = final_df[final_df['balls_left'] != 0]
```

In [330]:
```
X = final_df.iloc[:,:-1]
y = final_df.iloc[:,-1]
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)
```

In [331]:
```
X_train
```

Out[331]:

	batting_team	bowling_team	city	runs_left	balls_left	wickets	total_runs_x	crr	rrr
91985	Rajasthan Royals	Delhi Daredevils	Delhi	75	53	69	152	6.895522	8.490566
65235	Rajasthan Royals	Delhi Daredevils	Delhi	56	40	82	152	7.200000	8.400000
19954	Chennai Super Kings	Kings XI Punjab	Mumbai	46	58	65	112	6.387097	4.758621
9071	Rajasthan Royals	Deccan Chargers	Hyderabad	179	95	25	214	8.400000	11.305263
120161	Royal Challengers Bangalore	Kolkata Knight Riders	Kolkata	168	107	13	183	6.923077	9.420561
...	...	...	...	...	...	...	...	...	...
90558	Chennai Super Kings	Sunrisers Hyderabad	Sharjah	31	38	82	145	8.341463	4.894737
51761	Kings XI Punjab	Delhi Daredevils	Delhi	227	115	5	231	4.800000	11.843478
84529	Rajasthan Royals	Mumbai Indians	Mumbai	34	15	109	166	7.542857	13.600000
98074	Royal Challengers Bangalore	Kolkata Knight Riders	Kolkata	150	79	41	195	6.585366	11.392405
55906	Mumbai Indians	Deccan Chargers	Mumbai	103	72	49	135	4.000000	8.583333

57073 rows × 9 columns

In [332]:
```
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')
```

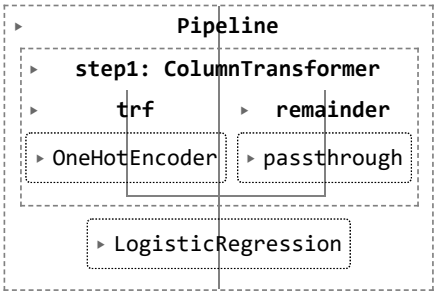
In [333]:
```
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier
from sklearn.pipeline import Pipeline
```

In [334]:
```
pipe = Pipeline(steps=[
    ('step1',trf),
    ('step2',LogisticRegression(solver='liblinear'))
])
```

In [335]:
```
pipe.fit(X_train,y_train)
```

```
C:\Users\Aditya\anaconda3\Lib\site-packages\sklearn\preprocessing\_encoders.py:972: FutureWarnin
g: `sparse` was renamed to `sparse_output` in version 1.2 and will be removed in 1.4. `sparse_ou
tput` is ignored unless you leave `sparse` to its default value.
  warnings.warn(
```

Out[335]:



In [336]:
```
y_pred = pipe.predict(X_test)
```

In [337]:
```
from sklearn.metrics import accuracy_score
accuracy_score(y_test,y_pred)
```

Out[337]:
```
0.7884925362674329
```

In [338]:
```
pipe.predict_proba(X_test)[10]
```

Out[338]:
```
array([0.03268684, 0.96731316])
```

In [339]:
```
def match_summary(row):
    print("Batting Team-" + row['batting_team'] + " | Bowling Team-" + row['bowling_team'] + " | 
```

In [340]:
```
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','tot
    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
```

In [341]:
```
temp_df,target = match_progression(delivery_df,74,pipe)
temp_df
```

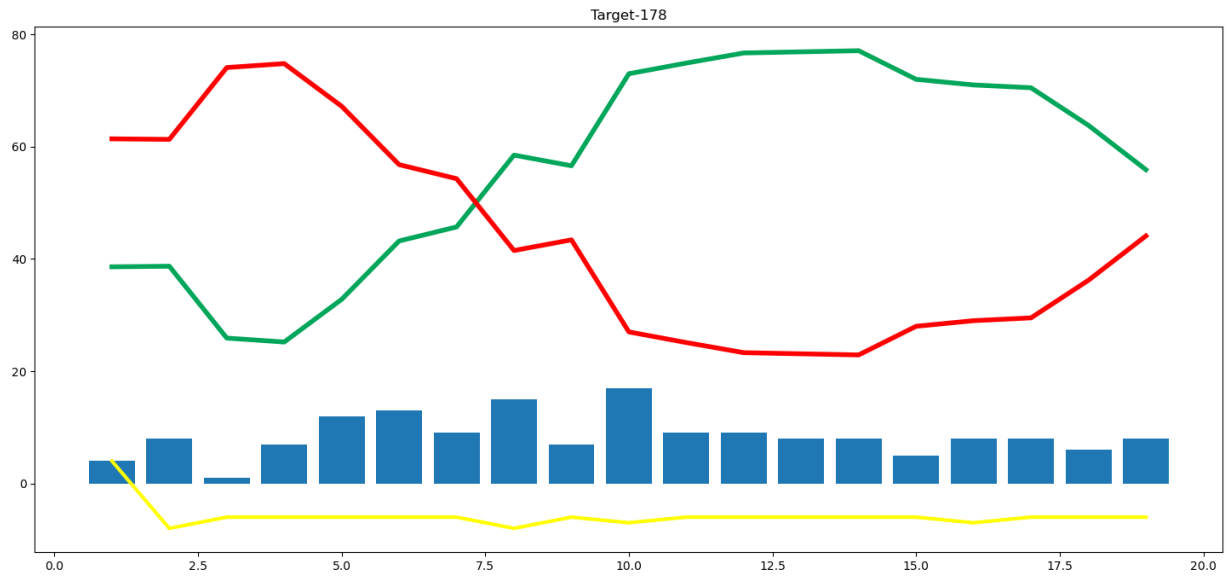
```
Target- 178
```

Out[341]:

	end_of_over	runs_after_over	wickets_in_over	lose	win
10459	1	4	4	61.4	38.6
10467	2	8	-8	61.3	38.7
10473	3	1	-6	74.1	25.9
10479	4	7	-6	74.8	25.2
10485	5	12	-6	67.2	32.8
10491	6	13	-6	56.8	43.2
10497	7	9	-6	54.3	45.7
10505	8	15	-8	41.5	58.5
10511	9	7	-6	43.4	56.6
10518	10	17	-7	27.0	73.0
10524	11	9	-6	25.1	74.9
10530	12	9	-6	23.3	76.7
10536	13	8	-6	23.1	76.9
10542	14	8	-6	22.9	77.1
10548	15	5	-6	28.0	72.0
10555	16	8	-7	29.0	71.0
10561	17	8	-6	29.5	70.5
10567	18	6	-6	36.2	63.8
10573	19	8	-6	44.1	55.9

In [342]:
```
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))
```

Out[342]:
```
Text(0.5, 1.0, 'Target-178')
```



In [343]:
```
teams
```

Out[343]:
```
['Sunrisers Hyderabad',
 'Mumbai Indians',
 'Royal Challengers Bangalore',
 'Kolkata Knight Riders',
 'Kings XI Punjab',
 'Chennai Super Kings',
 'Rajasthan Royals',
 'Delhi Capitals']
```

In [344]:
```
delivery_df['city'].unique()
```

Out[344]:
```
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)
```

In [345]:
```
import pickle
pickle.dump(pipe,open('pipe.pkl','wb'))
```

In [ ]:
```

```