VIT®
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

School of Computer Science and Engineering

J Component report

Programme : M.Sc (Data Science)

Course Title : EXPLORATORY DATA ANALYSIS

Course Code : CSE5007

Slot : I7+N7

Title: Exploratory Data Analysis and Visualisation on IPL Data

Team Members: Akshay K C | 21MDT1012

Kowsalya P | 21MDT1007

Shalmia S J | 21MDT1055

Faculty: Dr. Shruti Mishra

Sign:

Date:

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

DECLARATION

We, Akshay K C, Kowsalya P, Shalmia S J hereby declare that the thesis entitled **“Exploratory Data Analysis and Visualisation on IPL Data”** submitted by us, for the completion of the course, Exploratory Data Analysis is a record of Bonafide work carried out by us under the supervision of Dr. Shruti Mishra, our course instructor. We further declare that the work reported in this document has not been submitted and will not be submitted, either in part or in full, for any other courses in this institute or any other institute or university.

Place: Chennai

Date: 05.06.2022

Signature of the Candidates:

Akshay K C

Kowsalya P

Shalmia S J

Table of Contents

1. INTRODUCTION

2. PACKAGE USED

3. EDA AND VISUALISATIONS

4. CLUSTERING ALGORITHMS USED

5. CONCLUSION

Exploratory Data Analysis and Visualisation on IPL Data

1.INTRODUCTION

1.1 Introduction

India ranks top for having the leading cricket team world wide. The Indian Premier League(IPL) is popular all over the world . IPL was started in 2008 on the basis of ICL(Indian Cricket League) and is conducted during March or April every year. By bringing top cricket players from various countries ,they are grouped into 10 teams . They are Royal Challengers Bangalore (RCB), Kolkata Knight Riders (KKR), Chennai Super Kings(CSK), Sunrisers Hyderabad(SRH), Delhi Capitals (DC), Punjab Kings (PK),Mumbai Indians(MI), Rajasthan Royals(RR), Lucknow Super Giants(LSG), Gujrat Titans(GT).

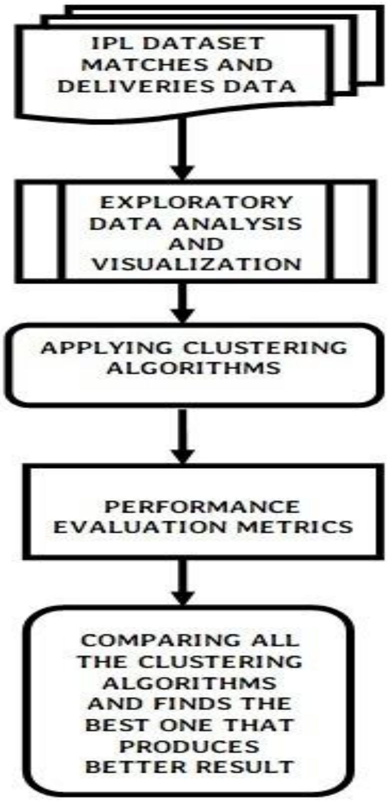
Top business people and Indian Artists owned these IPL teams by buying the players in auction. Though many problems have been raised in conducting this 20 over cricket match such as gambling and other issues from BCC, IPL didn't lose its vast audience all over the years. IPL ranked sixth for having a large audience among all other Sports League. Young players are given a chance to showcase their talents and passion for cricket.

1.2 OBJECTIVE

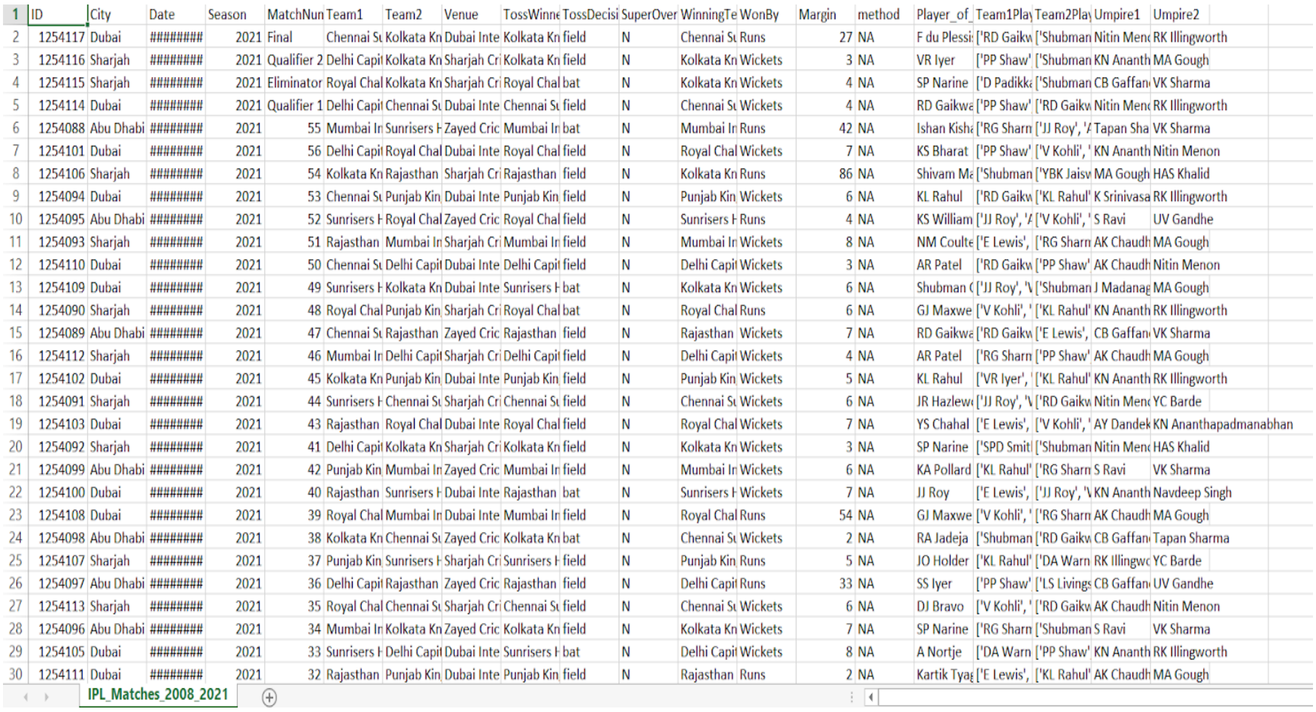
Our work contains the sports analysis of all the IPL teams played in each season from 2008-2021, all the players data. Cricket analysis forms a bridge between the players, coaches and managers. Players' performance history from the past can be an invaluable tool to select or buy the best players for the teams. The history can speak more about the players' consistency all over the years, way of approaching the game to a great extent. Existing dataset is used to perform analysis by considering various features to choose the best players for IPL. Visualisation are made to draw some conclusion from the data by ranking the players based on their runs, number of matches played, number of balls bowled etc.,. We performed various analysis on each player, team and season. Then we used clustering algorithms to group the players based on their performance. Also we have examined which algorithm works best using respective evaluation metrics.

Keywords- IPL; EDA; Visualisation; Clustering Algorithms

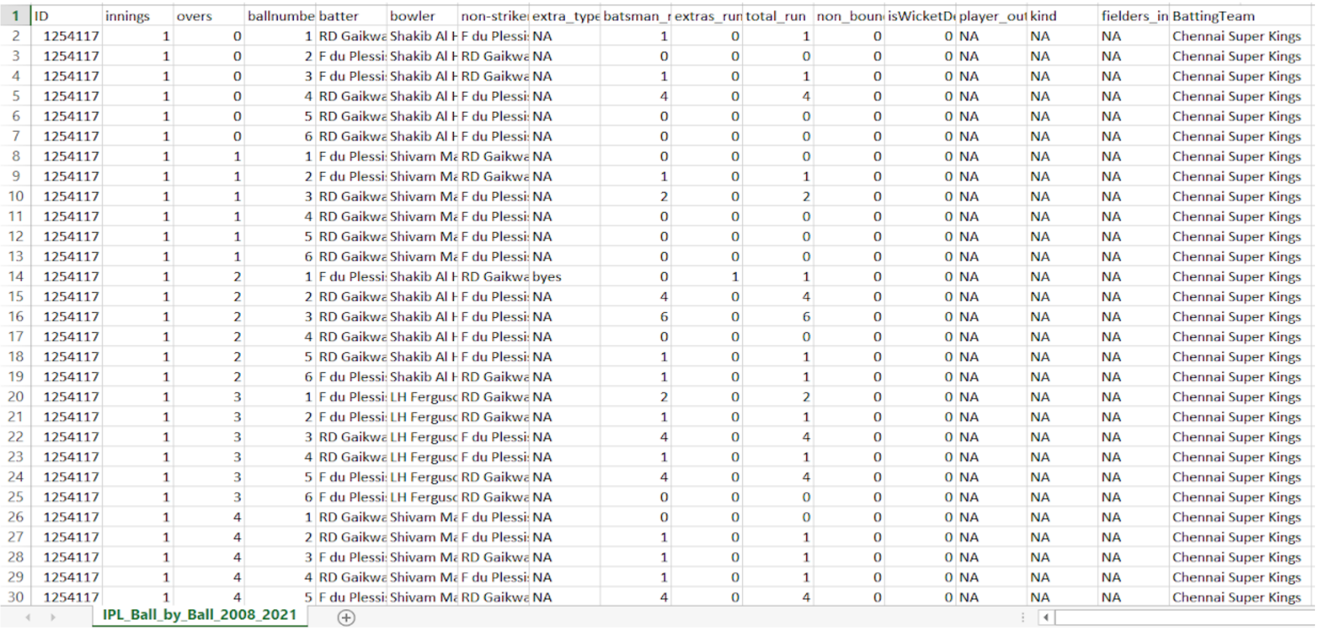
1.3 PROPOSED MODEL


IPL DATASET MATCHES AND DELIVERIES DATA
EXPLORATORY DATA ANALYSIS AND VISUALIZATION
APPLYING CLUSTERING ALGORITHMS
PERFORMANCE EVALUATION METRICS
COMPARING ALL THE CLUSTERING ALGORITHMS AND FINDS THE BEST ONE THAT PRODUCES BETTER RESULT

1.4 DATASET



1	ID	City	Date	Season	MatchNun	Team1	Team2	Venue	TossWinne	TossDecisi	SuperOver	WinningTe	WonBy	Margin	method	Player_of_	Team1Play	Team2Play	Umpire1	Umpire2
2	1254117	Dubai	########	2021	Final	Chennai Su	Kolkata Kn	Dubai Inte	Kolkata Kn	field	N	Chennai Su	Runs	27	NA	F du Plessi	['RD Gaikw	['Shubman	Nitin Menc	RK Illingworth
3	1254116	Sharjah	########	2021	Qualifier 2	Delhi Capit	Kolkata Kn	Sharjah Cri	Kolkata Kn	field	N	Kolkata Kn	Wickets	3	NA	VR Iyer	['PP Shaw'	['Shubman	KN Ananth	MA Gough
4	1254115	Sharjah	########	2021	Eliminator	Royal Chal	Kolkata Kn	Sharjah Cri	Royal Chal	bat	N	Kolkata Kn	Wickets	4	NA	SP Narine	['D Padikka	['Shubman	CB Gaffan	VK Sharma
5	1254114	Dubai	########	2021	Qualifier 1	Delhi Capit	Chennai Su	Dubai Inte	Chennai Su	field	N	Chennai Su	Wickets	4	NA	RD Gaikwa	['PP Shaw'	['RD Gaikw	Nitin Menc	RK Illingworth
6	1254088	Abu Dhabi	########	2021	55	Mumbai In	Sunrisers H	Zayed Cric	Mumbai In	bat	N	Mumbai In	Runs	42	NA	Ishan Kisha	['RG Sharn	['JJ Roy', 'A	Tapan Sha	VK Sharma
7	1254101	Dubai	########	2021	56	Delhi Capit	Royal Chal	Dubai Inte	Royal Chal	field	N	Royal Chal	Wickets	7	NA	KS Bharat	['PP Shaw'	['V Kohli', '	KN Ananth	Nitin Menon
8	1254106	Sharjah	########	2021	54	Kolkata Kn	Rajasthan	Sharjah Cri	Rajasthan	field	N	Kolkata Kn	Runs	86	NA	Shivam Ma	['Shubman	['YBK Jaisw	MA Gough	HAS Khalid
9	1254094	Dubai	########	2021	53	Chennai Su	Punjab Kin	Dubai Inte	Punjab Kin	field	N	Punjab Kin	Wickets	6	NA	KL Rahul	['RD Gaikw	['KL Rahul'	K Srinivasa	RK Illingworth
10	1254095	Abu Dhabi	########	2021	52	Sunrisers H	Royal Chal	Zayed Cric	Royal Chal	field	N	Sunrisers H	Runs	4	NA	KS William	['JJ Roy', 'A	['V Kohli', '	S Ravi	UV Gandhe
11	1254093	Sharjah	########	2021	51	Rajasthan	Mumbai In	Sharjah Cri	Mumbai In	field	N	Mumbai In	Wickets	8	NA	NM Coulte	['E Lewis',	['RG Sharn	AK Chaudh	MA Gough
12	1254110	Dubai	########	2021	50	Chennai Su	Delhi Capit	Dubai Inte	Delhi Capit	field	N	Delhi Capit	Wickets	3	NA	AR Patel	['RD Gaikw	['PP Shaw'	AK Chaudh	Nitin Menon
13	1254109	Dubai	########	2021	49	Sunrisers H	Kolkata Kn	Dubai Inte	Sunrisers H	bat	N	Kolkata Kn	Wickets	6	NA	Shubman G	['JJ Roy', 'V	['Shubman	J Madanag	MA Gough
14	1254090	Sharjah	########	2021	48	Royal Chal	Punjab Kin	Sharjah Cri	Royal Chal	bat	N	Royal Chal	Runs	6	NA	GJ Maxwe	['V Kohli', '	['KL Rahul'	KN Ananth	RK Illingworth
15	1254089	Abu Dhabi	########	2021	47	Chennai Su	Rajasthan	Zayed Cric	Rajasthan	field	N	Rajasthan	Wickets	7	NA	RD Gaikwa	['RD Gaikw	['E Lewis',	CB Gaffan	VK Sharma
16	1254112	Sharjah	########	2021	46	Mumbai In	Delhi Capit	Sharjah Cri	Delhi Capit	field	N	Delhi Capit	Wickets	4	NA	AR Patel	['RG Sharn	['PP Shaw'	AK Chaudh	MA Gough
17	1254102	Dubai	########	2021	45	Kolkata Kn	Punjab Kin	Dubai Inte	Punjab Kin	field	N	Punjab Kin	Wickets	5	NA	KL Rahul	['VR Iyer', '	['KL Rahul'	KN Ananth	RK Illingworth
18	1254091	Sharjah	########	2021	44	Sunrisers H	Chennai Su	Sharjah Cri	Chennai Su	field	N	Chennai Su	Wickets	6	NA	JR Hazlewo	['JJ Roy', 'V	['RD Gaikw	Nitin Menc	YC Barde
19	1254103	Dubai	########	2021	43	Rajasthan	Royal Chal	Dubai Inte	Royal Chal	field	N	Royal Chal	Wickets	7	NA	YS Chahal	['E Lewis',	['V Kohli', '	AY Dandek	KN Ananthapadmanabhan
20	1254092	Sharjah	########	2021	41	Delhi Capit	Kolkata Kn	Sharjah Cri	Kolkata Kn	field	N	Kolkata Kn	Wickets	3	NA	SP Narine	['SPD Smit	['Shubman	Nitin Menc	HAS Khalid
21	1254099	Abu Dhabi	########	2021	42	Punjab Kin	Mumbai In	Zayed Cric	Mumbai In	field	N	Mumbai In	Wickets	6	NA	KA Pollard	['KL Rahul'	['RG Sharn	S Ravi	VK Sharma
22	1254100	Dubai	########	2021	40	Rajasthan	Sunrisers H	Dubai Inte	Rajasthan	bat	N	Sunrisers H	Wickets	7	NA	JJ Roy	['E Lewis',	['JJ Roy', 'V	KN Ananth	Navdeep Singh
23	1254108	Dubai	########	2021	39	Royal Chal	Mumbai In	Dubai Inte	Mumbai In	field	N	Royal Chal	Runs	54	NA	GJ Maxwe	['V Kohli', '	['RG Sharn	AK Chaudh	MA Gough
24	1254098	Abu Dhabi	########	2021	38	Kolkata Kn	Chennai Su	Zayed Cric	Kolkata Kn	bat	N	Chennai Su	Wickets	2	NA	RA Jadeja	['Shubman	['RD Gaikw	CB Gaffan	Tapan Sharma
25	1254107	Sharjah	########	2021	37	Punjab Kin	Sunrisers H	Sharjah Cri	Sunrisers H	field	N	Punjab Kin	Runs	5	NA	JO Holder	['KL Rahul'	['DA Warn	RK Illingwo	YC Barde
26	1254097	Abu Dhabi	########	2021	36	Delhi Capit	Rajasthan	Zayed Cric	Rajasthan	field	N	Delhi Capit	Runs	33	NA	SS Iyer	['PP Shaw'	['LS Livings	CB Gaffan	UV Gandhe
27	1254113	Sharjah	########	2021	35	Royal Chal	Chennai Su	Sharjah Cri	Chennai Su	field	N	Chennai Su	Wickets	6	NA	DJ Bravo	['V Kohli', '	['RD Gaikw	AK Chaudh	Nitin Menon
28	1254096	Abu Dhabi	########	2021	34	Mumbai In	Kolkata Kn	Zayed Cric	Kolkata Kn	field	N	Kolkata Kn	Wickets	7	NA	SP Narine	['RG Sharn	['Shubman	S Ravi	VK Sharma
29	1254105	Dubai	########	2021	33	Sunrisers H	Delhi Capit	Dubai Inte	Sunrisers H	bat	N	Delhi Capit	Wickets	8	NA	A Nortje	['DA Warn	['PP Shaw'	KN Ananth	RK Illingworth
30	1254111	Dubai	########	2021	32	Rajasthan	Punjab Kin	Dubai Inte	Punjab Kin	field	N	Rajasthan	Runs	2	NA	Kartik Tyag	['E Lewis',	['KL Rahul'	AK Chaudh	MA Gough



1	ID	innings	overs	ballnumbe	batter	bowler	non-strike	extra_type	batsman_r	extras_run	total_run	non_boun	isWicketD	player_out	kind	fielders_in	BattingTeam
2	1254117	1	0	1	RD Gaikwa	Shakib Al H	F du Plessi	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
3	1254117	1	0	2	F du Plessi	Shakib Al H	RD Gaikwa	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
4	1254117	1	0	3	F du Plessi	Shakib Al H	RD Gaikwa	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
5	1254117	1	0	4	RD Gaikwa	Shakib Al H	F du Plessi	NA	4	0	4	0	0	NA	NA	NA	Chennai Super Kings
6	1254117	1	0	5	RD Gaikwa	Shakib Al H	F du Plessi	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
7	1254117	1	0	6	RD Gaikwa	Shakib Al H	F du Plessi	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
8	1254117	1	1	1	F du Plessi	Shivam Ma	RD Gaikwa	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
9	1254117	1	1	2	F du Plessi	Shivam Ma	RD Gaikwa	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
10	1254117	1	1	3	RD Gaikwa	Shivam Ma	F du Plessi	NA	2	0	2	0	0	NA	NA	NA	Chennai Super Kings
11	1254117	1	1	4	RD Gaikwa	Shivam Ma	F du Plessi	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
12	1254117	1	1	5	RD Gaikwa	Shivam Ma	F du Plessi	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
13	1254117	1	1	6	RD Gaikwa	Shivam Ma	F du Plessi	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
14	1254117	1	2	1	F du Plessi	Shakib Al H	RD Gaikwa	byes	0	1	1	0	0	NA	NA	NA	Chennai Super Kings
15	1254117	1	2	2	RD Gaikwa	Shakib Al H	F du Plessi	NA	4	0	4	0	0	NA	NA	NA	Chennai Super Kings
16	1254117	1	2	3	RD Gaikwa	Shakib Al H	F du Plessi	NA	6	0	6	0	0	NA	NA	NA	Chennai Super Kings
17	1254117	1	2	4	RD Gaikwa	Shakib Al H	F du Plessi	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
18	1254117	1	2	5	RD Gaikwa	Shakib Al H	F du Plessi	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
19	1254117	1	2	6	F du Plessi	Shakib Al H	RD Gaikwa	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
20	1254117	1	3	1	F du Plessi	LH Ferguso	RD Gaikwa	NA	2	0	2	0	0	NA	NA	NA	Chennai Super Kings
21	1254117	1	3	2	F du Plessi	LH Ferguso	RD Gaikwa	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
22	1254117	1	3	3	RD Gaikwa	LH Ferguso	F du Plessi	NA	4	0	4	0	0	NA	NA	NA	Chennai Super Kings
23	1254117	1	3	4	RD Gaikwa	LH Ferguso	F du Plessi	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
24	1254117	1	3	5	F du Plessi	LH Ferguso	RD Gaikwa	NA	4	0	4	0	0	NA	NA	NA	Chennai Super Kings
25	1254117	1	3	6	F du Plessi	LH Ferguso	RD Gaikwa	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
26	1254117	1	4	1	RD Gaikwa	Shivam Ma	F du Plessi	NA	0	0	0	0	0	NA	NA	NA	Chennai Super Kings
27	1254117	1	4	2	RD Gaikwa	Shivam Ma	F du Plessi	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
28	1254117	1	4	3	F du Plessi	Shivam Ma	RD Gaikwa	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
29	1254117	1	4	4	RD Gaikwa	Shivam Ma	F du Plessi	NA	1	0	1	0	0	NA	NA	NA	Chennai Super Kings
30	1254117	1	4	5	F du Plessi	Shivam Ma	RD Gaikwa	NA	4	0	4	0	0	NA	NA	NA	Chennai Super Kings

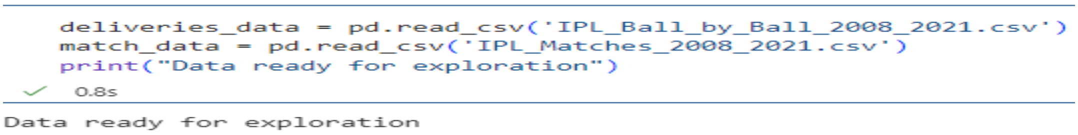
2.PACKAGES

2.1 PACKAGES USED

- Pandas
- Numpy
- Matplotlib
- Seaborn
- Sklearn
- Plotly
- Ipython

3. EXPLORATORY DATA ANALYSIS AND VISUALISATION

INPUT DATA

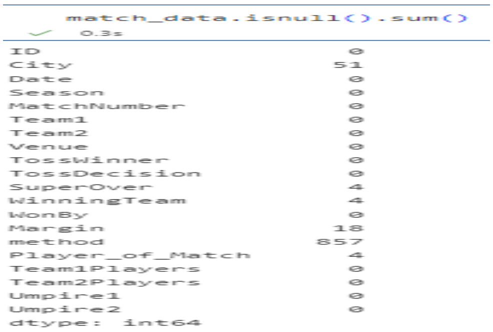


```
deliveries_data = pd.read_csv('IPL_Ball_by_Ball_2008_2021.csv')
match_data = pd.read_csv('IPL_Matches_2008_2021.csv')
print("Data ready for exploration")
```

✓ 0.8s

```
Data ready for exploration
```

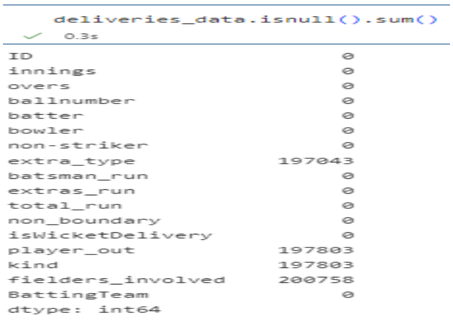
DATA EXPLORATION



```
match_data.isnull().sum()
```

✓ 0.3s

```
ID                   0
City                51
Date                 0
Season               0
MatchNumber          0
Team1                0
Team2                0
Venue                0
TossWinner           0
TossDecision         0
SuperOver            4
WinningTeam          4
WonBy                0
Margin              18
method             857
Player_of_Match      4
Team1Players         0
Team2Players         0
Umpire1              0
Umpire2              0
dtype: int64
```



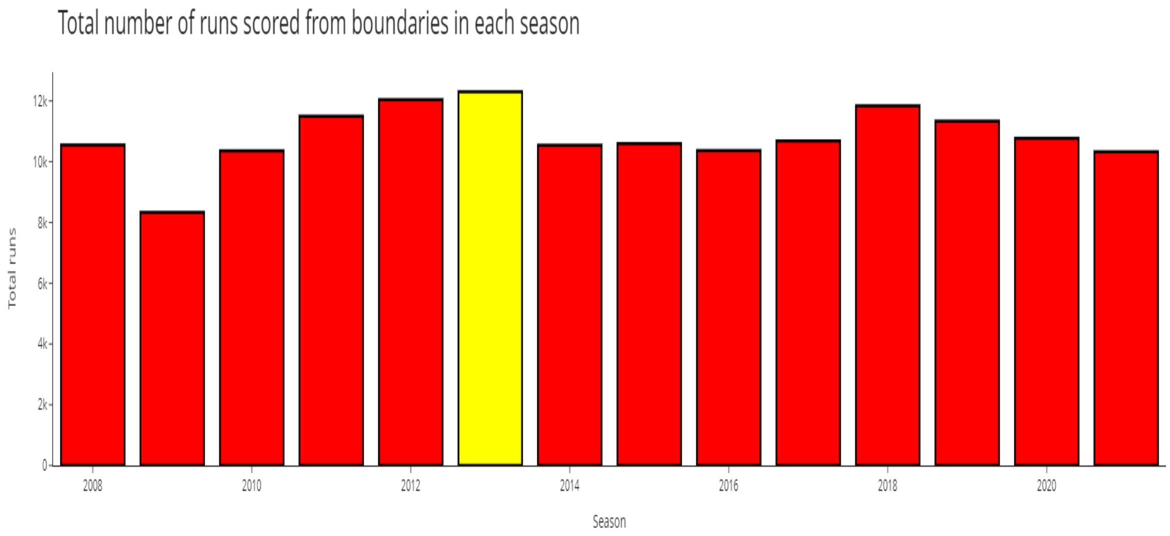
```
deliveries_data.isnull().sum()
```

✓ 0.3s

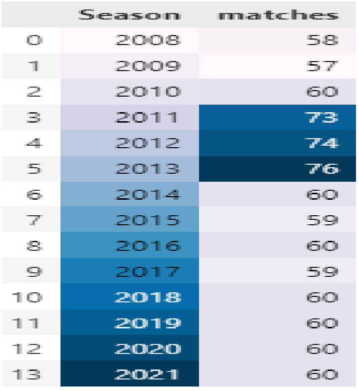
```
ID                       0
innings                  0
overs                    0
ballnumber               0
batter                   0
bowler                   0
non-striker              0
extra_type          197043
batsman_run              0
extras_run               0
total_run                0
non_boundary             0
isWicketDelivery         0
player_out          197803
kind                197803
fielders_involved   200758
BattingTeam              0
dtype: int64
```

3.1 SEASON-WISE ANALYSIS

TOTAL NO OF RUNS SCORED FROM BOUNDARIES IN EACH SEASON

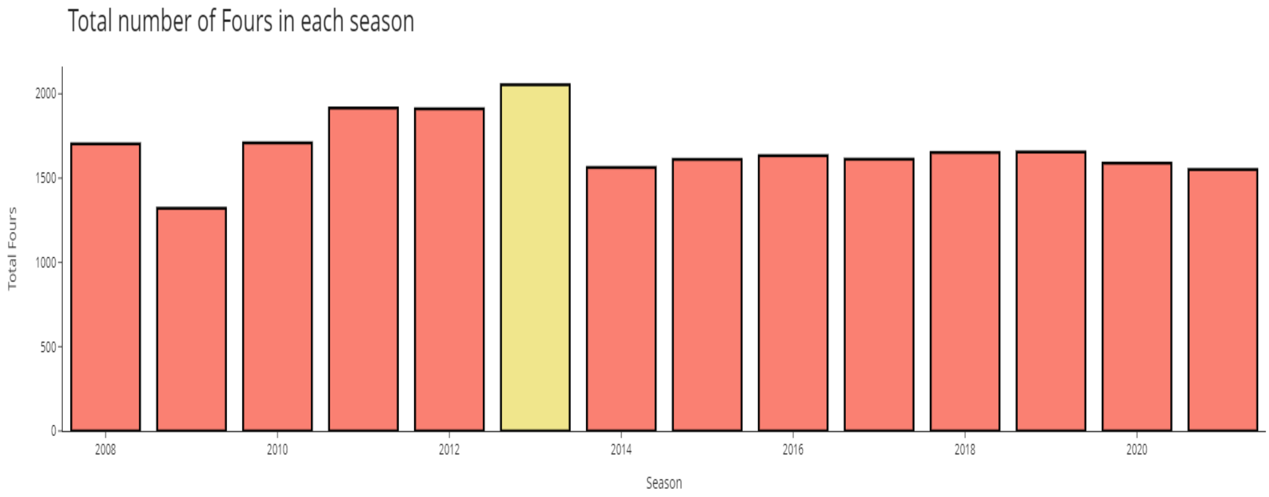

Total number of runs scored from boundaries in each season

NO OF MATCHES PLAYED IN EACH SEASON

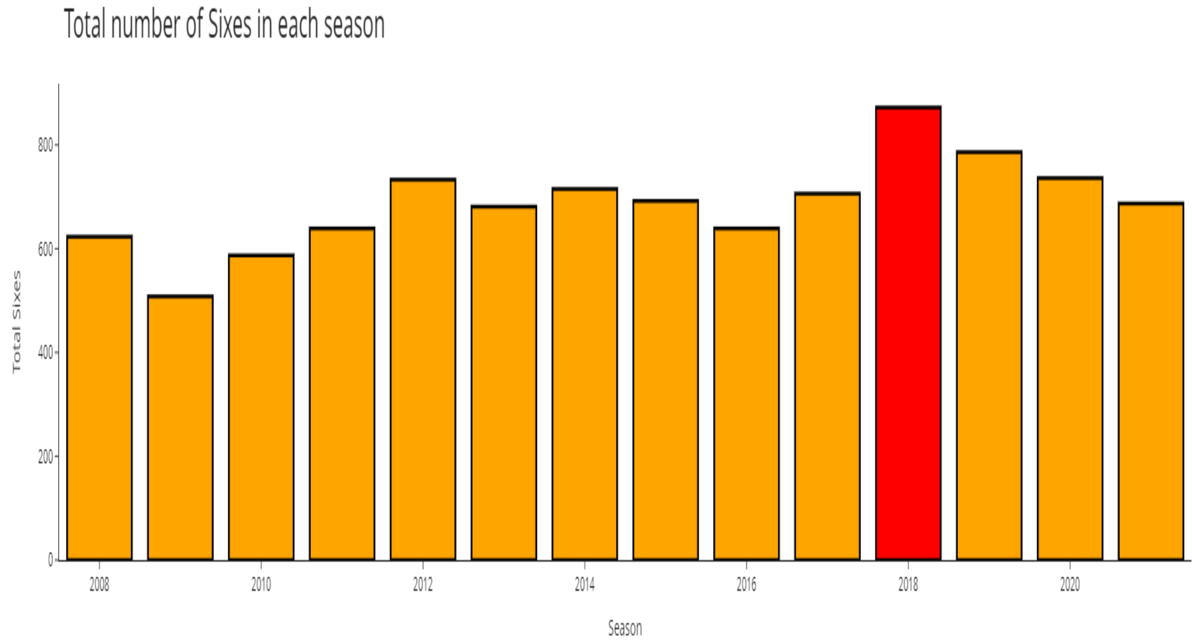


	Season	matches
0	2008	58
1	2009	57
2	2010	60
3	2011	73
4	2012	74
5	2013	76
6	2014	60
7	2015	59
8	2016	60
9	2017	59
10	2018	60
11	2019	60
12	2020	60
13	2021	60

TOTAL NO OF FOURS IN EACH SEASON


Total number of Fours in each season

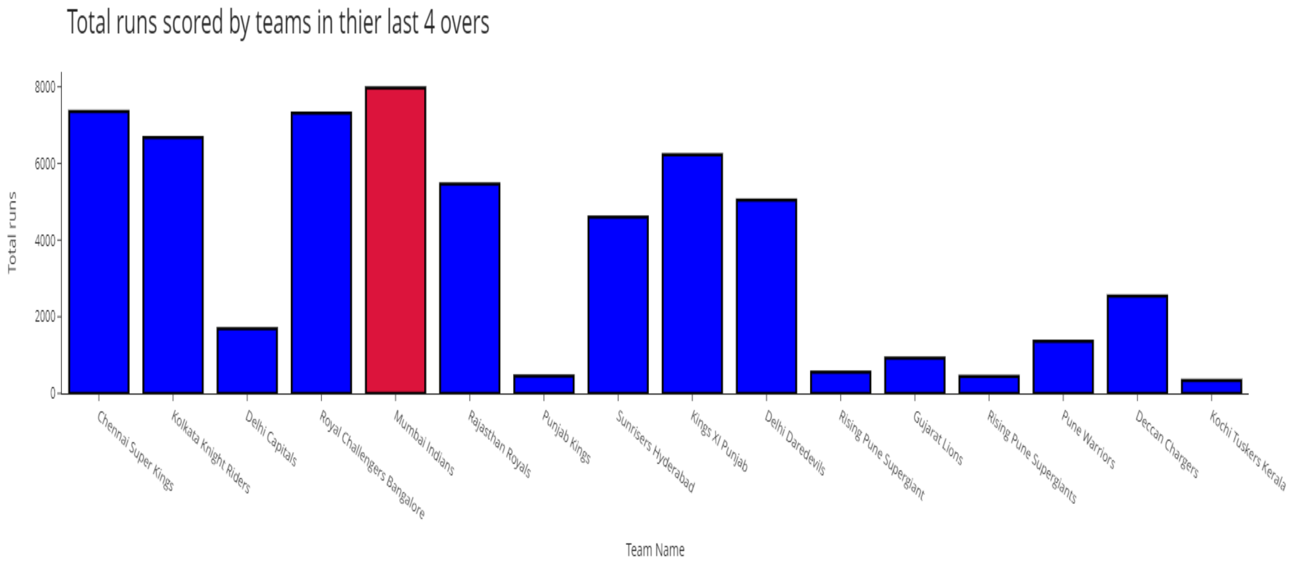
TOTAL NO OF SIXES IN EACH SEASON



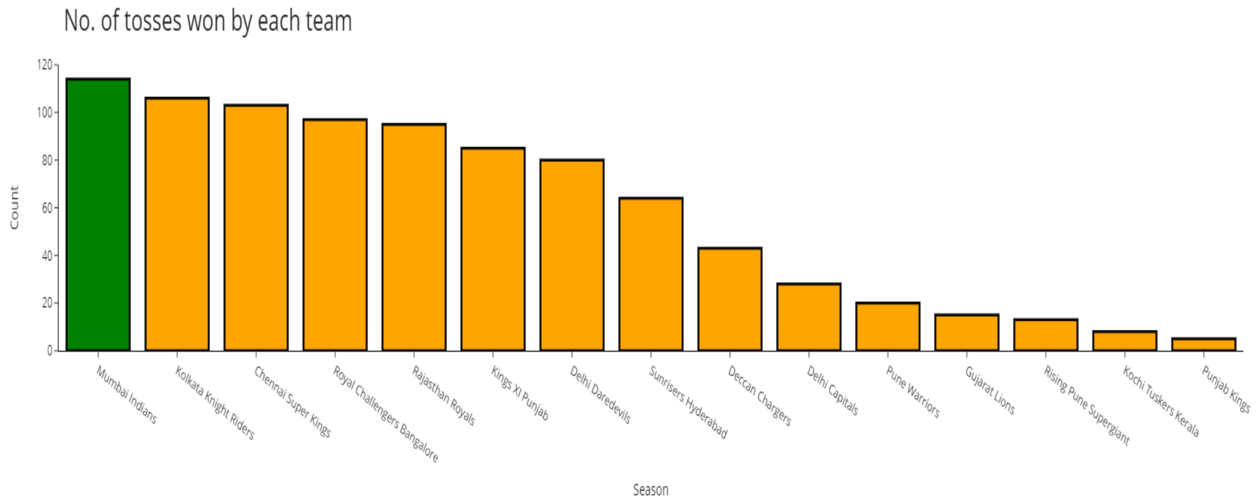
The dataset from kaggle is used to perform EDA. The null values are removed using specific commands in python . To perform season-wise analysis, total number of boundaries in each season, number of matches played etc were explored . In the year 2013 most number of matches have been played by teams. Also this year more number of boundaries were scored by players. The year 2018 is known for having more number of sixes from the players.

3.2 TEAM-WISE ANALYSIS

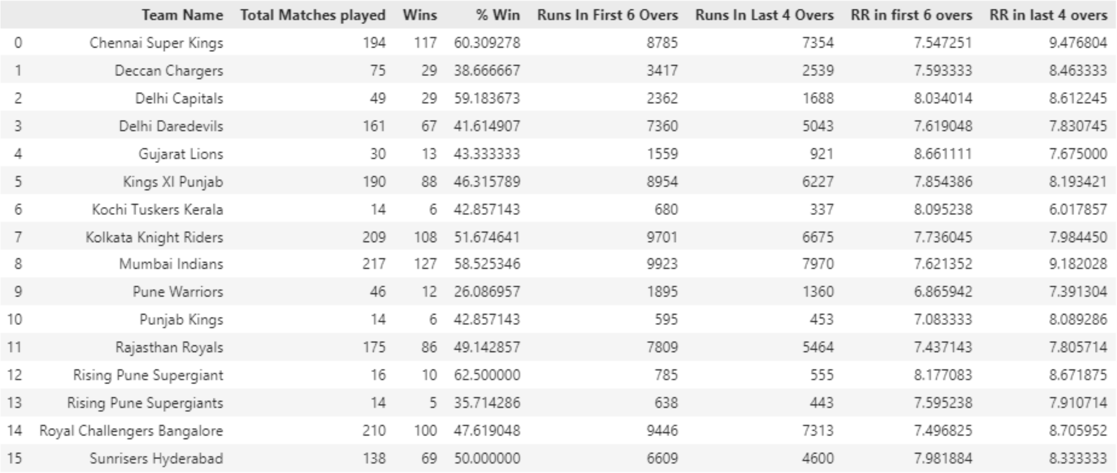
TOTAL RUNS SCORED BY TEAMS IN THEIR LAST 4 OVERS



NO OF TOSSES WON BY EACH TEAM

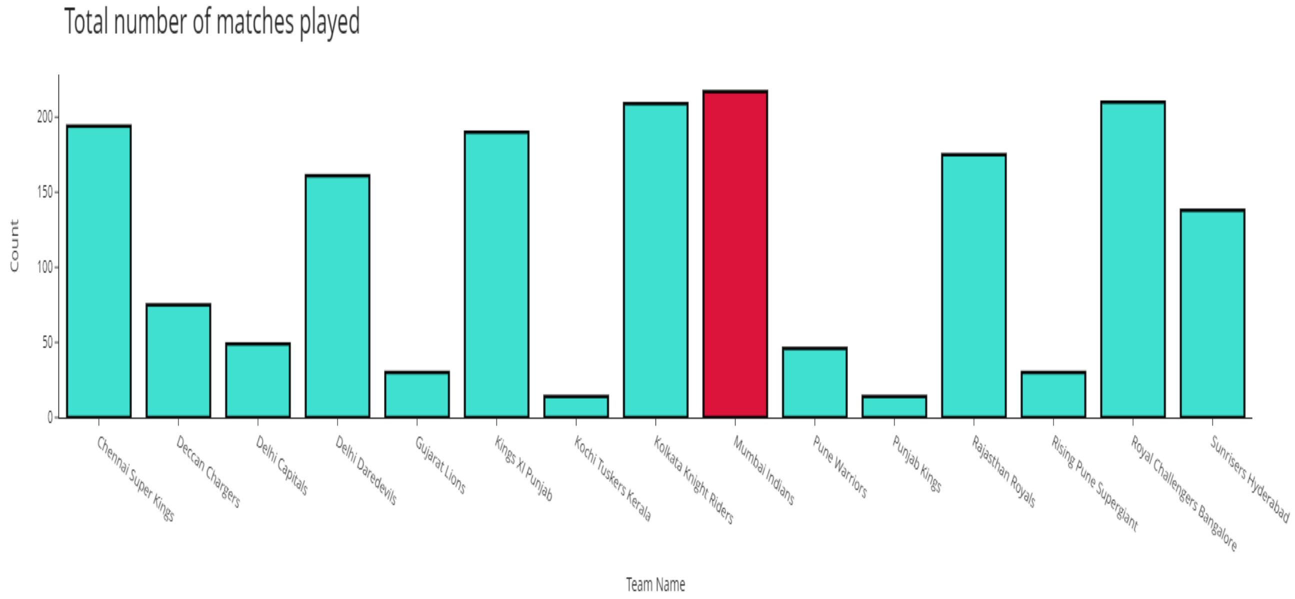


RUN RATE IN FIRST 6 OVERS

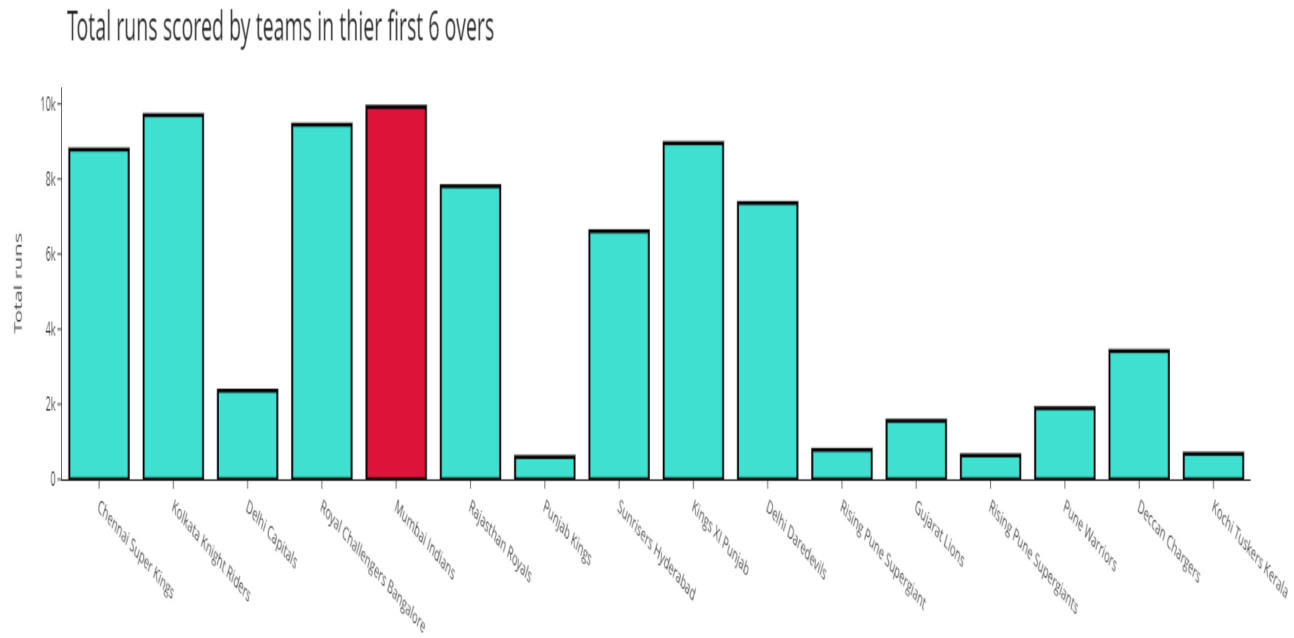


	Team Name	Total Matches played	Wins	% Win	Runs In First 6 Overs	Runs In Last 4 Overs	RR in first 6 overs	RR in last 4 overs
0	Chennai Super Kings	194	117	60.309278	8785	7354	7.547251	9.476804
1	Deccan Chargers	75	29	38.666667	3417	2539	7.593333	8.463333
2	Delhi Capitals	49	29	59.183673	2362	1688	8.034014	8.612245
3	Delhi Daredevils	161	67	41.614907	7360	5043	7.619048	7.830745
4	Gujarat Lions	30	13	43.333333	1559	921	8.661111	7.675000
5	Kings XI Punjab	190	88	46.315789	8954	6227	7.854386	8.193421
6	Kochi Tuskers Kerala	14	6	42.857143	680	337	8.095238	6.017857
7	Kolkata Knight Riders	209	108	51.674641	9701	6675	7.736045	7.984450
8	Mumbai Indians	217	127	58.525346	9923	7970	7.621352	9.182028
9	Pune Warriors	46	12	26.086957	1895	1360	6.865942	7.391304
10	Punjab Kings	14	6	42.857143	595	453	7.083333	8.089286
11	Rajasthan Royals	175	86	49.142857	7809	5464	7.437143	7.805714
12	Rising Pune Supergiant	16	10	62.500000	785	555	8.177083	8.671875
13	Rising Pune Supergiants	14	5	35.714286	638	443	7.595238	7.910714
14	Royal Challengers Bangalore	210	100	47.619048	9446	7313	7.496825	8.705952
15	Sunrisers Hyderabad	138	69	50.000000	6609	4600	7.981884	8.333333

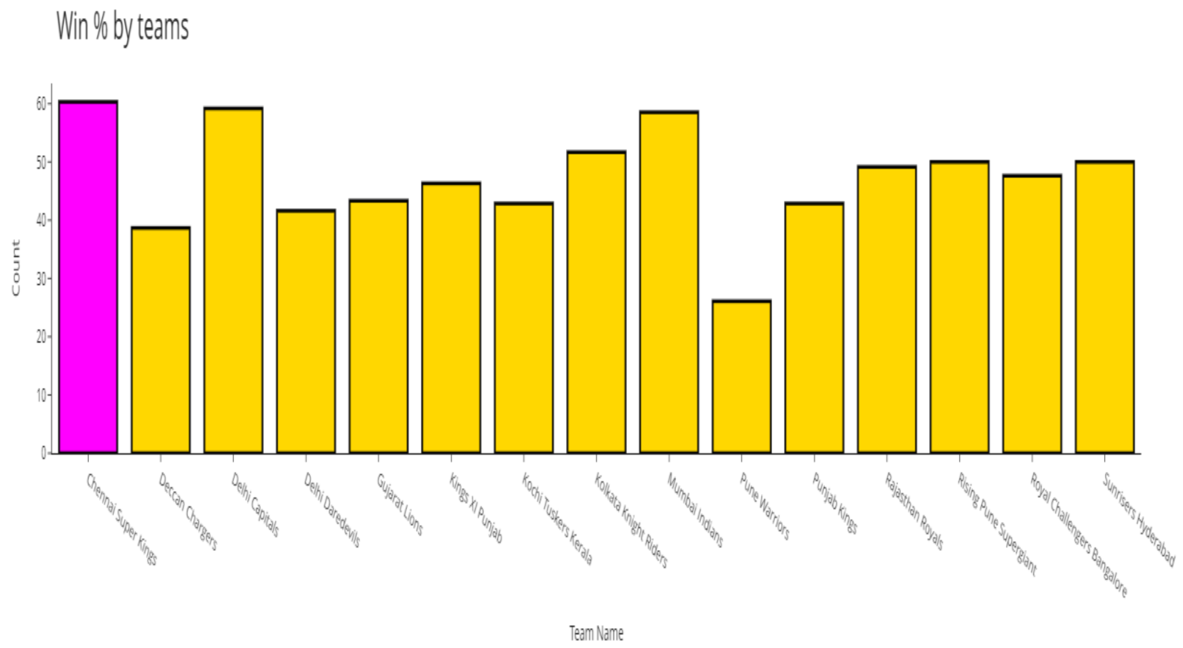
TOTAL NUMBER OF MATCHES PLAYED BY EACH TEAM



TOTAL RUNS SCORED BY TEAMS IN FIRST 6 OVERS



WIN % BY TEAMS

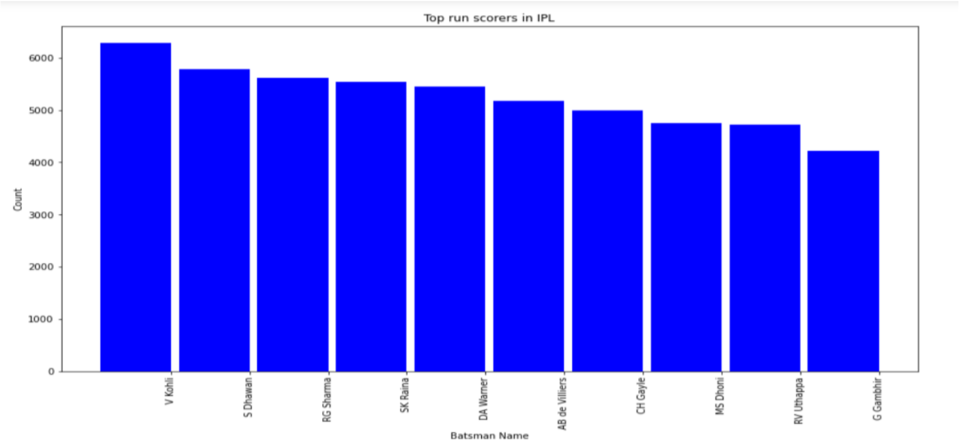


To perform team-wise analysis the total number of runs scored by each team in their last 4 overs were specially considered. Mumbai Indians scored nearly 8000 runs in their last 4 overs of the match till date and they were lucky enough to win more number of tosses. Likewise the run rate and total runs of each team for first six overs were also gathered. Mumbai Indians played most number of matches. The win percentage of CSK team is considerably high.

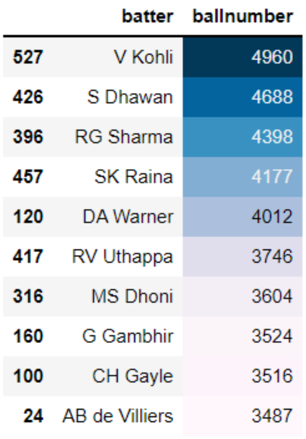
3.3PLAYER ANALYSIS

BATTER ANALYSIS

LEADING RUN SCORERS

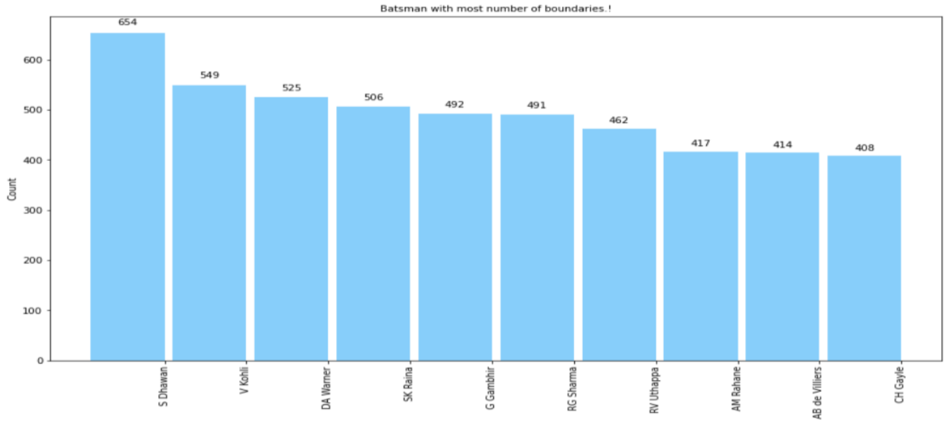


BATSMAN WHO HAD PLAYED MORE NUMBER OF BALLS

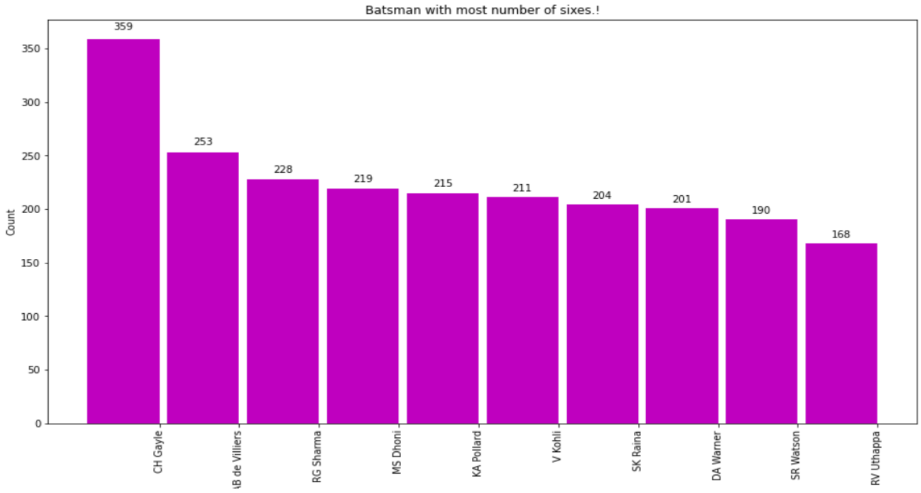


	batter	ballnumber
527	V Kohli	4960
426	S Dhawan	4688
396	RG Sharma	4398
457	SK Raina	4177
120	DA Warner	4012
417	RV Uthappa	3746
316	MS Dhoni	3604
160	G Gambhir	3524
100	CH Gayle	3516
24	AB de Villiers	3487

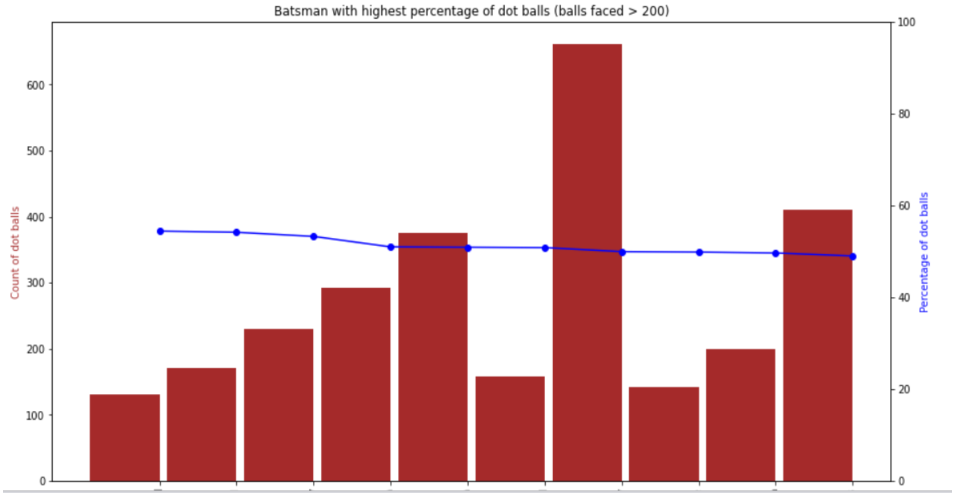
PLAYER WHO HAD HIT MOST NUMBER OF 4s



PLAYER WHO HAD HIT MOST NUMBER OF 6s

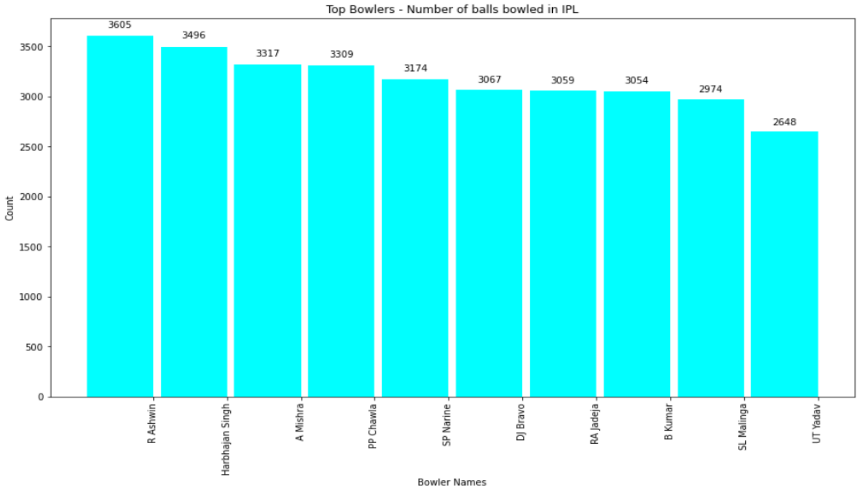


BATSMAN WITH HIGHEST PERCENTAGE OF DOT BALLS

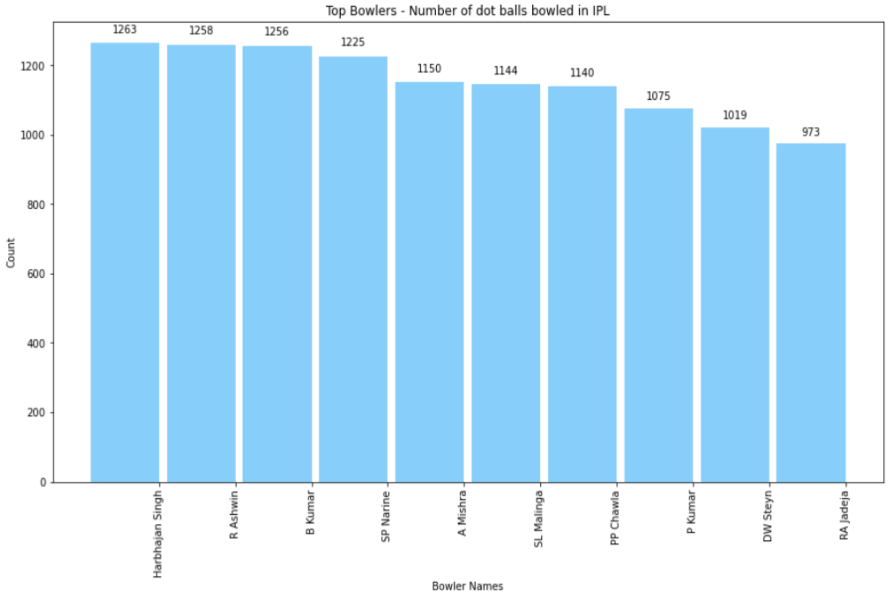


BOWLWER ANALYSIS

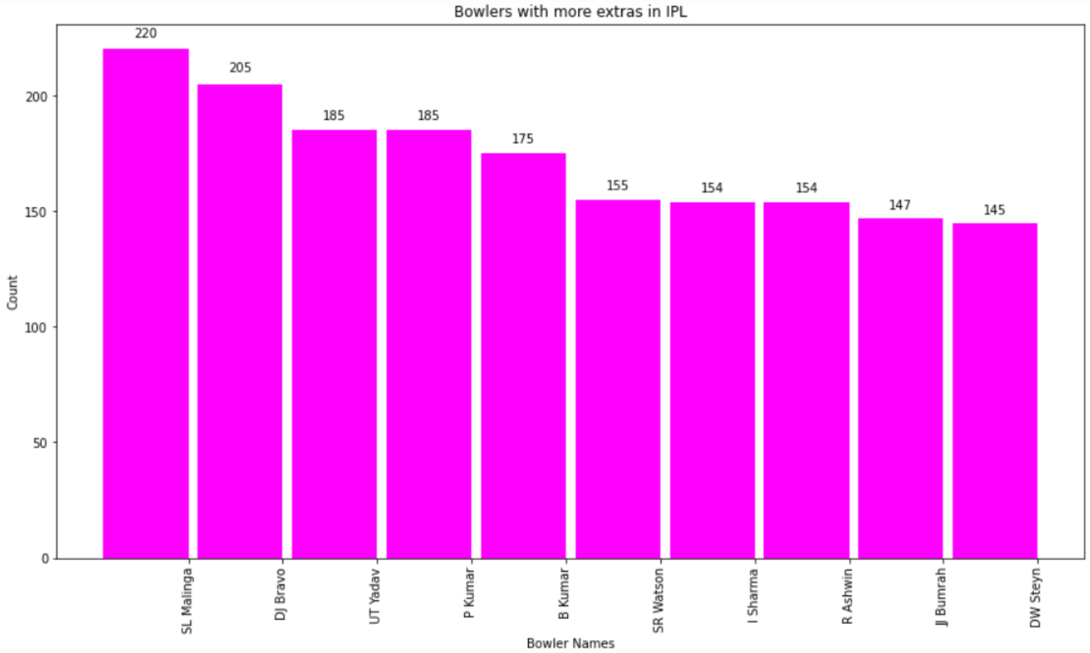
BOWLWERS WHO HAD BOWLED MOST NUMBER OF BALLS IN IPL



BOWLWERS WITH MORE NUMBER OF DOT BALLS



BOWLERS WITH MORE NUMBER OF EXTRAS

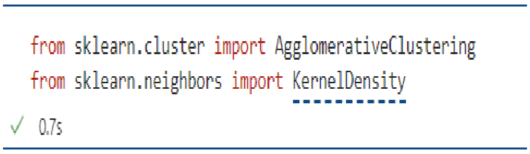


Virat kohli had scored 4960 runs in IPL and is ranked top among the players who got more number of balls. S Dhawan scored more number of boundaries and G H Gayle scored more n umber of sixes. In top bowlers list Ashwin bowled more times. Harahan Singh bowled more dot balls and S Malinga bowled more extras.

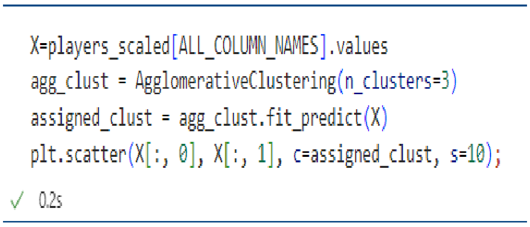
4. CLUSTERING ALGORITHMS

4.1 Hierarchical clustering

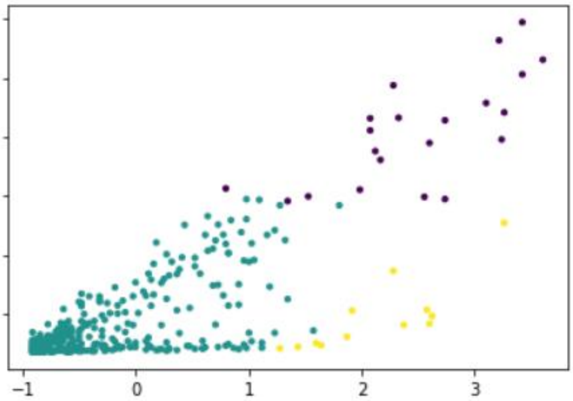
Hierarchical clustering is an alterative to the partitional methods. It groups the similar objects together. It has a tree structure arranged in sequential order. At first all the groups will be combined, then it splits and forms respective groups. They are two types of clustering: Agglomerative and Divise approach. In agglomerative single clusters tend to form similar combined groups. In Divise approach the combined cluters split to form their own different clusters. There are different cluster distance measures such as single linkage, complete linkage and Average linkage. Average linkage takes the average of the diatance two datapoints.

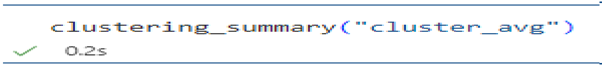


```
from sklearn.cluster import AgglomerativeClustering
from sklearn.neighbors import KernelDensity
✓ 0.7s
```

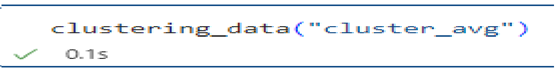


```
X=players_scaled[ALL_COLUMN_NAMES].values
agg_clust = AgglomerativeClustering(n_clusters=3)
assigned_clust = agg_clust.fit_predict(X)
plt.scatter(X[:, 0], X[:, 1], c=assigned_clust, s=10);
✓ 0.2s
```

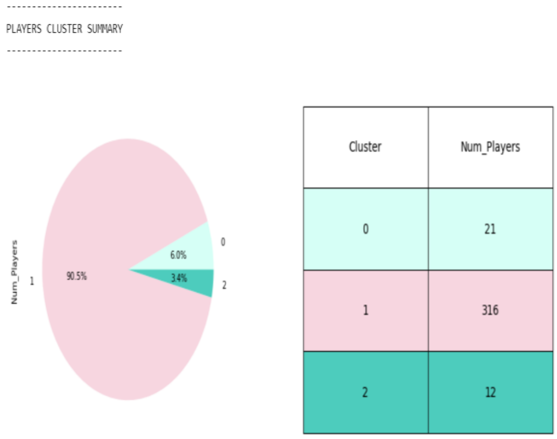




```
clustering_summary("cluster_avg")
✓ 0.2s
```

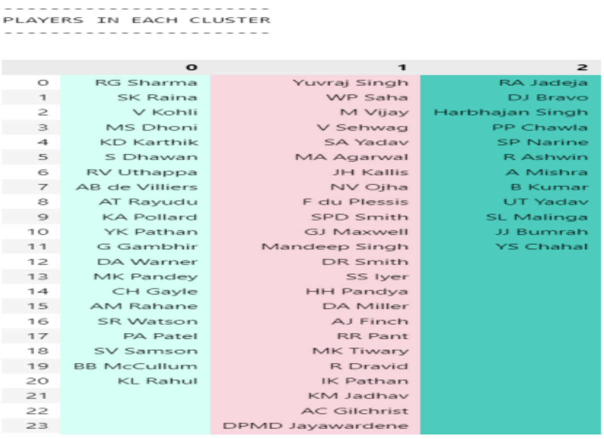


```
clustering_data("cluster_avg")
✓ 0.1s
```



```
-----------------------
PLAYERS CLUSTER SUMMARY
-----------------------
```

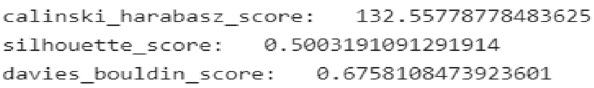
Cluster	Num_Players
0	21
1	316
2	12



```
-----------------------
PLAYERS IN EACH CLUSTER
-----------------------
```

	0	1	2
0	RG Sharma	Yuvraj Singh	RA Jadeja
1	SK Raina	WP Saha	DJ Bravo
2	V Kohli	M Vijay	Harbhajan Singh
3	MS Dhoni	V Sehwag	PP Chawla
4	KD Karthik	SA Yadav	SP Narine
5	S Dhawan	MA Agarwal	R Ashwin
6	RV Uthappa	JH Kallis	A Mishra
7	AB de Villiers	NV Ojha	B Kumar
8	AT Rayudu	F du Plessis	UT Yadav
9	KA Pollard	SPD Smith	SL Malinga
10	YK Pathan	GJ Maxwell	JJ Bumrah
11	G Gambhir	Mandeep Singh	YS Chahal
12	DA Warner	DR Smith	
13	MK Pandey	SS Iyer	
14	CH Gayle	HH Pandya	
15	AM Rahane	DA Miller	
16	SR Watson	AJ Finch	
17	PA Patel	RR Pant	
18	SV Samson	MK Tiwary	
19	BB McCullum	R Dravid	
20	KL Rahul	IK Pathan	
21		KM Jadhav	
22		AC Gilchrist	
23		DPMD Jayawardene	

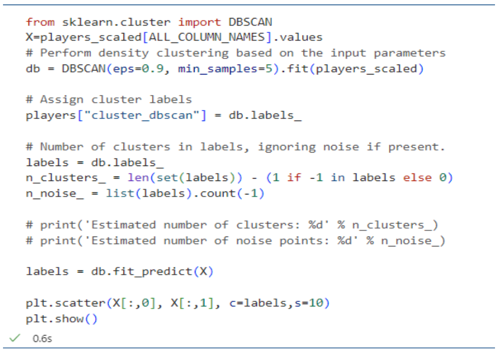
PERFORMANE EVALUATION – HIERARCHICAL CLUSTERING – AGNES



```
calinski_harabasz_score:    132.55778778483625
silhouette_score:    0.5003191091291914
davies_bouldin_score:    0.6758108473923601
```

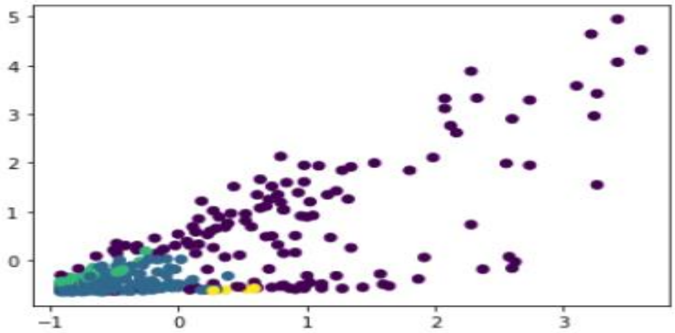
4.2 DBSCAN CLUSTERING

DBSCAN clustering : DBSCAN (Density Based Spatial Clustering of Applications with Noise) falls under the category of density based clustering. In heirarchical and patitional clustering, clusters will be formed depending on the number of K values. But density clustering is formed based on the point and area. It is effective for non-linear or arbitrary shapes. Noise can be easily identified and removed. It finds and combines neighbourhood values to form clusters and equally separate them. Based on the number of points, number of clusters are formed. There is no need for predefining the number of clusters. It requires two parameters based on which clusters are formed. They are Epsilon and minPoints . Epsilon is the radius or the distance between the points from which a circle is created. MinPoints correspond the points inside each circle. Data points are divided as core point, border point and noise point to perform clustering based on the parameters.



```
from sklearn.cluster import DBSCAN
X=players_scaled[ALL_COLUMN_NAMES].values
# Perform density clustering based on the input parameters
db = DBSCAN(eps=0.9, min_samples=5).fit(players_scaled)

# Assign cluster labels
players["cluster_dbscan"] = db.labels_

# Number of clusters in labels, ignoring noise if present.
labels = db.labels_
n_clusters_ = len(set(labels)) - (1 if -1 in labels else 0)
n_noise_ = list(labels).count(-1)

# print('Estimated number of clusters: %d' % n_clusters_)
# print('Estimated number of noise points: %d' % n_noise_)

labels = db.fit_predict(X)

plt.scatter(X[:,0], X[:,1], c=labels,s=10)
plt.show()
```

✓ 0.6s



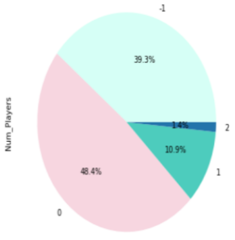
```
clustering_summary("cluster_dbscan")
```

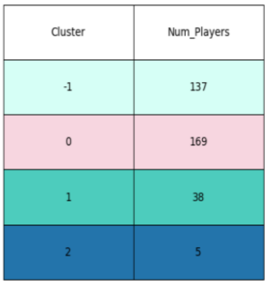
✓ 0.3s

```
clustering_data("cluster_dbscan")
```

✓ 0.2s

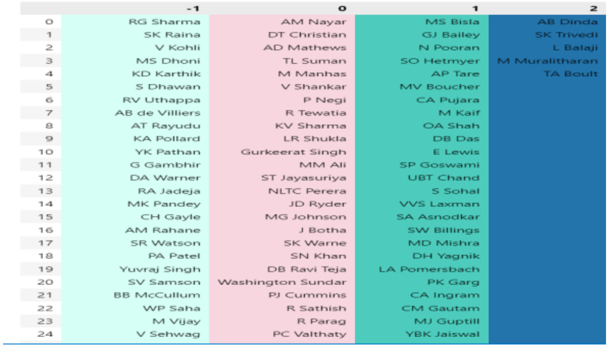
```
----------------------
PLAYERS CLUSTER SUMMARY
----------------------
```





Cluster	Num_Players
-1	137
0	169
1	38
2	5

```
------------------------
PLAYERS IN EACH CLUSTER
------------------------
```



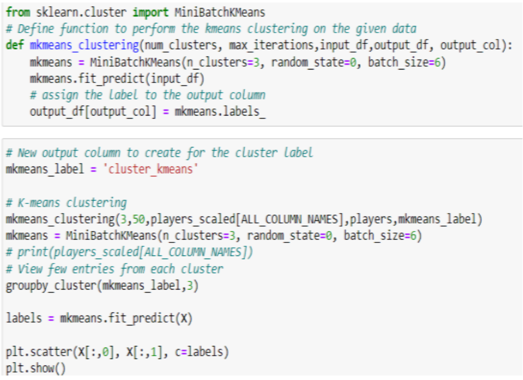
	-1	0	1	2
0	RG Sharma	AM Nayar	MS Bisla	AB Dinda
1	SK Raina	DT Christian	GJ Bailey	SK Trivedi
2	V Kohli	AD Mathews	N Pooran	L Balaji
3	MS Dhoni	TL Suman	SO Hetmyer	M Muralitharan
4	KD Karthik	M Manhas	AP Tare	TA Boult
5	S Dhawan	V Shankar	MV Boucher	
6	RV Uthappa	P Negi	CA Pujara	
7	AB de Villiers	R Tewatia	M Kaif	
8	AT Rayudu	KV Sharma	OA Shah	
9	KA Pollard	LR Shukla	DB Das	
10	YK Pathan	Gurkeerat Singh	E Lewis	
11	G Gambhir	MM Ali	SP Goswami	
12	DA Warner	ST Jayasuriya	UBT Chand	
13	RA Jadeja	NLTC Perera	S Sohal	
14	MK Pandey	JD Ryder	VVS Laxman	
15	CH Gayle	MG Johnson	SA Asnodkar	
16	AM Rahane	J Botha	SW Billings	
17	SR Watson	SK Warne	MD Mishra	
18	PA Patel	SN Khan	DH Yagnik	
19	Yuvraj Singh	DB Ravi Teja	LA Pomersbach	
20	SV Samson	Washington Sundar	PK Garg	
21	BB McCullum	PJ Cummins	CA Ingram	
22	WP Saha	R Sathish	CM Gautam	
23	M Vijay	R Parag	MJ Guptill	
24	V Sehwag	PC Valthaty	YBK Jaiswal	

PERFORMANE EVALUATION – DBSCAN CLUSTERING

```
Silhouette Coefficient: 0.076
Calinski-Harabasz Index: 50.014
Davies-Bouldin Index: 1.304
```

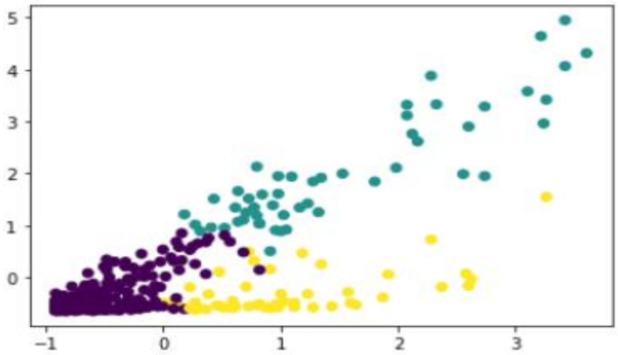
4.3 MINI BATCH K MEANS CLUSTERING

A version of the classic K-means clustering technique is the Mini-batch K-means algorithm. It saves data in small, arbitrary, specified batches in memory, then gathers and uses a random sample of the data to update the clusters with each iteration. There’s no need to keep the entire dataset in memory. The distance between the mini-batch and the k centroids must be determined at each iteration. The user must store k centroids and a chunk of data in memory for each iteration. Because it does not loop over the complete dataset, it sometimes outperforms the usual K-means algorithm when working on large datasets. The key benefit of adopting the Mini-batch K-means algorithm is that it lowers the computing cost of cluster detection.



```python
from sklearn.cluster import MiniBatchKMeans
# Define function to perform the kmeans clustering on the given data
def mkmeans_clustering(num_clusters, max_iterations,input_df,output_df, output_col):
    mkmeans = MiniBatchKMeans(n_clusters=3, random_state=0, batch_size=6)
    mkmeans.fit_predict(input_df)
    # assign the label to the output column
    output_df[output_col] = mkmeans.labels_
```

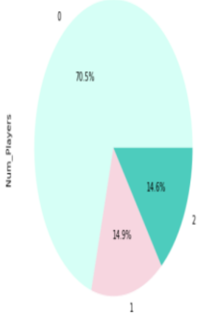
```python
# New output column to create for the cluster label
mkmeans_label = 'cluster_kmeans'

# K-means clustering
mkmeans_clustering(3,50,players_scaled[ALL_COLUMN_NAMES],players,mkmeans_label)
mkmeans = MiniBatchKMeans(n_clusters=3, random_state=0, batch_size=6)
# print(players_scaled[ALL_COLUMN_NAMES])
# View few entries from each cluster
groupby_cluster(mkmeans_label,3)

labels = mkmeans.fit_predict(X)

plt.scatter(X[:,0], X[:,1], c=labels)
plt.show()
```

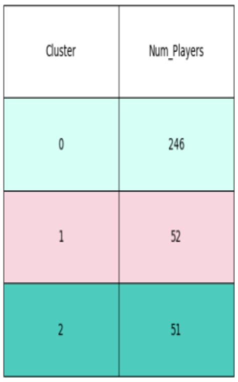


```
clustering_summary(mkmeans_label)
```

```
clustering_data(mkmeans_label)
```

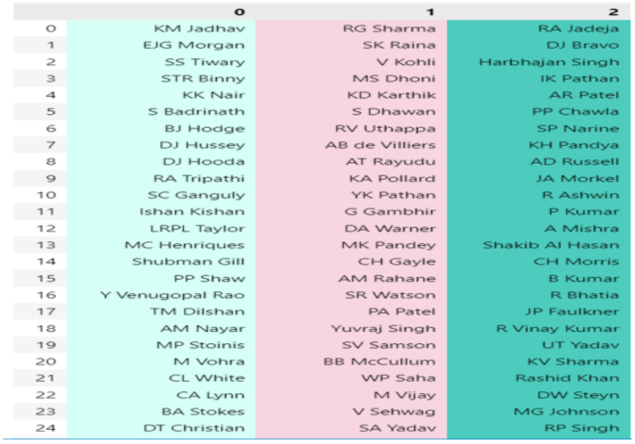
```
........................
PLAYERS CLUSTER SUMMARY
........................
```





Cluster	Num_Players
0	246
1	52
2	51

```
-----------------------
PLAYERS IN EACH CLUSTER
-----------------------
```



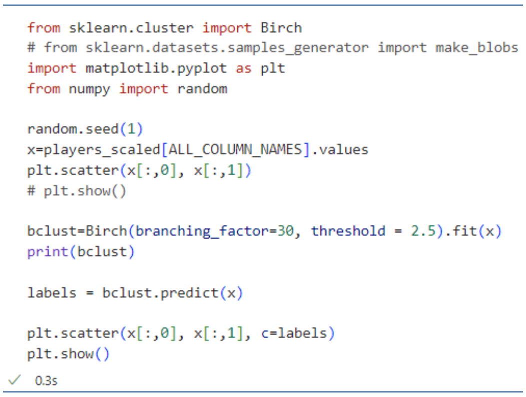
	0	1	2
0	KM Jadhav	RG Sharma	RA Jadeja
1	EJG Morgan	SK Raina	DJ Bravo
2	SS Tiwary	V Kohli	Harbhajan Singh
3	STR Binny	MS Dhoni	IK Pathan
4	KK Nair	KD Karthik	AR Patel
5	S Badrinath	S Dhawan	PP Chawla
6	BJ Hodge	RV Uthappa	SP Narine
7	DJ Hussey	AB de Villiers	KH Pandya
8	DJ Hooda	AT Rayudu	AD Russell
9	RA Tripathi	KA Pollard	JA Morkel
10	SC Ganguly	YK Pathan	R Ashwin
11	Ishan Kishan	G Gambhir	P Kumar
12	LRPL Taylor	DA Warner	A Mishra
13	MC Henriques	MK Pandey	Shakib Al Hasan
14	Shubman Gill	CH Gayle	CH Morris
15	PP Shaw	AM Rahane	B Kumar
16	Y Venugopal Rao	SR Watson	R Bhatia
17	TM Dilshan	PA Patel	JP Faulkner
18	AM Nayar	Yuvraj Singh	R Vinay Kumar
19	MP Stoinis	SV Samson	UT Yadav
20	M Vohra	BB McCullum	KV Sharma
21	CL White	WP Saha	Rashid Khan
22	CA Lynn	M Vijay	DW Steyn
23	BA Stokes	V Sehwag	MG Johnson
24	DT Christian	SA Yadav	RP Singh

PERFORMANE EVALUATION – MINI BATCH K MEANS CLUSTERING

```
Silhouette Coefficient: 0.438
Calinski-Harabasz Index: 224.560
Davies-Bouldin Index: 0.867
```

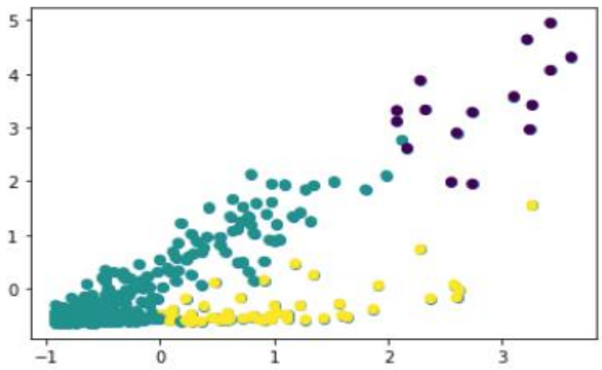
4.4 BIRCH CLUSTERING

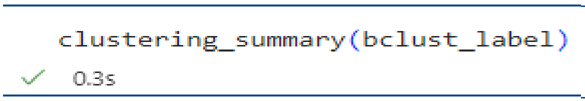
BIRCH clustering: It is developed from heirarchical clustering especially multi-phasse hierachical clustering. Balanced Iterative Reducing and Clustering using Hierarchies(BIRCH). For generating the final cluster some iterative process is taken. There is some kind of threshold to balance the cluster generation problems. The threshold can be reduced correspondingly and heirachical clustering is performed. Effective for clustering using large datasets. The existing algorithms can produce high I/O costs. This is where BIRCH is used since it is dynamically adjusted taking into acount the available storage.



```
from sklearn.cluster import Birch
# from sklearn.datasets.samples_generator import make_blobs
import matplotlib.pyplot as plt
from numpy import random

random.seed(1)
x=players_scaled[ALL_COLUMN_NAMES].values
plt.scatter(x[:,0], x[:,1])
# plt.show()

bclust=Birch(branching_factor=30, threshold = 2.5).fit(x)
print(bclust)

labels = bclust.predict(x)

plt.scatter(x[:,0], x[:,1], c=labels)
plt.show()
```

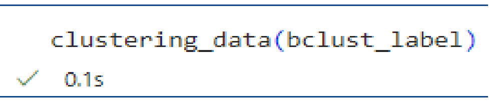
✓ 0.3s





```
clustering_summary(bclust_label)
```

✓ 0.3s

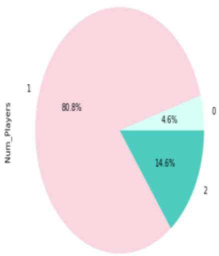


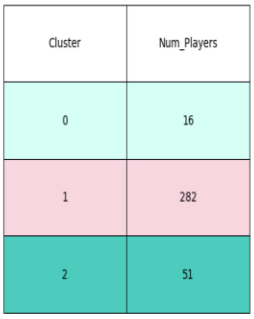
```
clustering_data(bclust_label)
```

✓ 0.1s

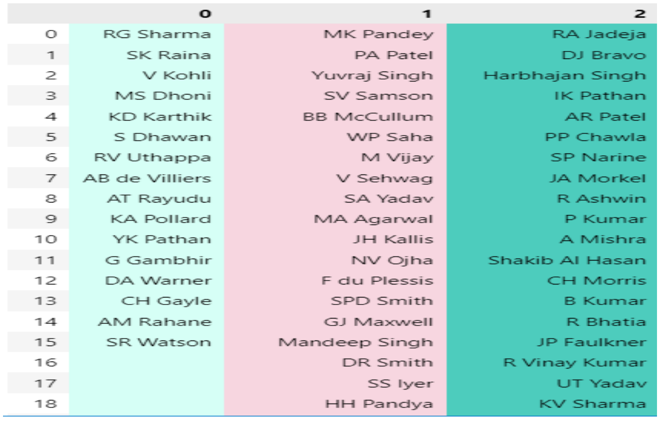
```
------------------------
PLAYERS IN EACH CLUSTER
------------------------
```

```
------------------------
PLAYERS CLUSTER SUMMARY
------------------------
```



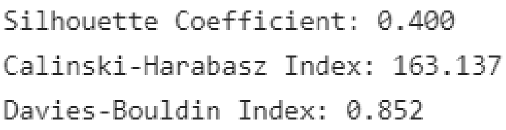


Cluster	Num_Players
0	16
1	282
2	51



	0	1	2
0	RG Sharma	MK Pandey	RA Jadeja
1	SK Raina	PA Patel	DJ Bravo
2	V Kohli	Yuvraj Singh	Harbhajan Singh
3	MS Dhoni	SV Samson	IK Pathan
4	KD Karthik	BB McCullum	AR Patel
5	S Dhawan	WP Saha	PP Chawla
6	RV Uthappa	M Vijay	SP Narine
7	AB de Villiers	V Sehwag	JA Morkel
8	AT Rayudu	SA Yadav	R Ashwin
9	KA Pollard	MA Agarwal	P Kumar
10	YK Pathan	JH Kallis	A Mishra
11	G Gambhir	NV Ojha	Shakib Al Hasan
12	DA Warner	F du Plessis	CH Morris
13	CH Gayle	SPD Smith	B Kumar
14	AM Rahane	GJ Maxwell	R Bhatia
15	SR Watson	Mandeep Singh	JP Faulkner
16		DR Smith	R Vinay Kumar
17		SS Iyer	UT Yadav
18		HH Pandya	KV Sharma

PERFORMANE EVALUATION – BIRCH CLUSTERING



```
Silhouette Coefficient: 0.400
Calinski-Harabasz Index: 163.137
Davies-Bouldin Index: 0.852
```

5. RESULTS OBTAINED

We performed Season-wise, team-wise and player wise analysis on matches and deliveries dataset and visualized the results.We extracted the details(matches played ,strike rate, balls bowled, runs etc.)of all the players through out all the seasons and created a dataset which we have used to perform clustering.The players performance using 4 clustering algorithms are compared. Based on the performance metrics we conclude that hierarchical clustering, AGNES using average linkage produced better clustering results compared to Mini batch k means, Birch and DBSCAN clustering.

6. REFERENCE

- https://www.kaggle.com/

- Sudhamathy, G., & Meenakshi, G. R. (2020). Prediction on IPL Data Using Machine Learning Techniques In R Package. *ICTACT Journal on Soft Computing*, *11*(1), 2199-2204.

- Sinha, A. (2020). Application of Machine Learning in Cricket and Predictive Analytics of IPL 2020.

- Dey, P. K., Chakraborty, G., Ruj, P., & Sarkar, S. (2012). A Data Mining Approach on Cluster Analysis of IPL. *International Journal of Machine Learning and Computing*, *2*(4), 351.

- Singh, Sneha, et al. "Analysis and Prediction of Cricket Match Using Machine Learning." *Research & Review: Machine Learning and Cloud Computing* 1.1 (2022): 30-37.

- Mohapatra, Santanu, et al. "Exploratory Data Analysis on IPL Data." *Contemporary Issues in Communication, Cloud and Big Data Analytics*. Springer, Singapore, 2022. 315-325.

- Lamsal, R., & Choudhary, A. (2018). Predicting Outcome of Indian Premier League (IPL) Matches Using Machine Learning. *arXiv preprint arXiv:1809.09813*.

- Dey, P. K. (2012). Fuzzy Clustering Technique in IPL database. *International Journal of Advanced Computer Science*, *2*(7), 259-262.

- Kapadia, K., Abdel-Jaber, H., Thabtah, F., & Hadi, W. (2020). Sport analytics for cricket game results using machine learning: An experimental study. *Applied Computing and Informatics*.

- Rani, P. J., Kamath, A. V., Menon, A., Dhatwalia, P., Rishabh, D., & Kulkarni, A. (2020, July). Selection of Players and Team for an Indian Premier League Cricket Match Using Ensembles of Classifiers. In *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)* (pp. 1-6). IEEE.

- Kansal, P., Kumar, P., Arya, H., & Methaila, A. (2014, November). Player valuation in Indian premier league auction using data mining technique. In *2014 international conference on contemporary computing and informatics (IC3I)* (pp. 197-203). IEEE.

- Ray, S., & Sengupta, K. (2018). Reflecting Design Considerations: An End-to-End Case Study on Preparing Cricket Data Available on Net Analysis Ready. *IUP Journal of Information Technology, 14*(3).

- Fung, G. (2001). A comprehensive overview of basic clustering algorithms.

- Xu, R., & Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, *16*(3), 645-678.

- Maulik, U., & Bandyopadhyay, S. (2002). Performance evaluation of some clustering algorithms and validity indices. *IEEE Transactions on pattern analysis and machine intelligence*, *24*(12), 1650-1654.

- Ahmad, P. H., & Dang, S. (2015). Performance evaluation of clustering algorithm using different datasets. *International Journal of Advance Research in Computer Science and Management Studies*, *3*(1), 167-173.