

IPL - Exploratory Data Analysis with MySQL

Analysis – for analysis we need data (IPL data)

<https://www.kaggle.com/ramjidoolla/ipl-data-set>

- 🔧 Requisite: - MySQL should be installed on system
- 🔧 Data files
- We have data file and mysql to begin with analysis

1. Create database schema to load the tables

```
CREATE DATABASE ipl;
```

2. Create table to store data

```
CREATE TABLE ipl.matches(  
    id integer, season integer, city varchar(20), date date,  
    team1 varchar(20), team2 varchar(20), toss_winner varchar(20), toss_decision  
    varchar(20),  
    result varchar(50), dl_applied integer, winner varchar(50), win_by_runs  
    integer, win_by_wickets integer,  
    player_of_the_match varchar(50), venue varchar(20), umpire1 varchar(20),  
    umpire2 varchar(20), umpire3 integer  
);
```

```
CREATE TABLE ipl.deliveries (
```

```
    matchid integer, inning integer, batting_team varchar(20), bowling_team  
    varchar(20),
```

```
    overc integer, ball integer, batsman varchar(20), non_striker varchar(20), bowler  
    varchar(20),
```

```
    is_super_over integer, wide_runs integer, bye_runs integer, legbye_runs integer,  
    noball_runs integer,
```

```
    penalty_runs integer, batsman_runs integer, extra_runs integer, total_runs integer,
```

```
    player_dismissed varchar(20), dismissal_kind varchar(50), fielder varchar(20)
```

```
);
```

3. Check the tables structure
4. Now, Load the data in tables

```
SHOW GLOBAL VARIABLES LIKE 'local_infile';  
# -- SET GLOBAL local_infile = 'ON';  
SET GLOBAL local_infile=1;
```

```
load data local infile 'D:/Excel For Analysis/matches.csv'  
into table ipl.matches  
fields terminated by ','  
lines terminated by '\n'  
ignore 1 rows;
```

```
load data local infile 'D:/Excel For Analysis/deliveries.csv'  
into table ipl.deliveries  
fields terminated by ','  
lines terminated by '\n'  
ignore 1 rows;
```

Now, Begin with the queries:

--Shape of Data

Shapes are combination of rows and columns

To know about your table structure (I mean rows and columns) use below query.

```
select count(*) as No_of_row from ipl.matches;
```

	No_of_row
▶	636

```
select count(*) as No_of_columns from information_schema.columns where  
TABLE_NAME='matches';
```

	No_of_columns
▶	18

--Viewing data

```
select * from ipl.matches;
```

```
Select * from ipl.deliveries;
```

--View selected columns

```
select m.season,m.city,m.date,m.team1,m.team2,m.winner,m.win_by_runs from ipl.matches m
```

```
where m.season='2017'
```

```
limit 5;
```

	season	city	date	team1	team2	winner	win_by_runs
▶	2017	Hyderabad	2017-04-05	Sunrisers Hyderabad	Royal Challengers Ba	Sunrisers Hyderabad	35
	2017	Pune	2017-04-06	Mumbai Indians	Rising Pune Supergia	Rising Pune Supergiant	0
	2017	Rajkot	2017-04-07	Gujarat Lions	Kolkata Knight Rider	Kolkata Knight Riders	0
	2017	Indore	2017-04-08	Rising Pune Supergia	Kings XI Punjab	Kings XI Punjab	0
	2017	Bangalore	2017-04-08	Royal Challengers Ba	Delhi Daredevils	Royal Challengers Bangalore	15

--Distinct values

```
select distinct year(m.date) as 'Year of Match' from ipl.matches m
```

```
order by 1;
```

	Year of Match
▶	2008
	2009
	2010
	2011
	2012
	2013
	2014
	2015
	2016

```
select count(distinct player_of_the_match) as 'No of Matches' from ipl.matches m
```

```
order by 1;
```

	No of Matches
▶	202

--Find season winner for each season (season winner is the winner of the last match of each season)

```
select distinctrow m.season, m.winner from ipl.matches m
```

```
order by m.season desc;
```

	season	winner
▶	2017	Delhi Daredevils
	2017	Gujarat Lions
	2017	Kings XI Punjab
	2017	Kolkata Knight Riders
	2017	Mumbai Indians
	2017	Rising Pune Supergiant
	2017	Royal Challengers Bangalore
	2017	Sunrisers Hyderabad
	2016	Delhi Daredevils
	2016	Gujarat Lions

select distinct season, winner from ipl.matches order by season desc;

	season	winner
▶	2017	Delhi Daredevils
	2017	Gujarat Lions
	2017	Kings XI Punjab
	2017	Kolkata Knight Riders
	2017	Mumbai Indians
	2017	Rising Pune Supergiant
	2017	Royal Challengers Bangalore
	2017	Sunrisers Hyderabad
	2016	Delhi Daredevils
	2016	Gujarat Lions

--Find venue of 10 most recently played matches

select DISTINCT m.venue,m.date from ipl.matches m

order by m.date desc

limit 10;

	venue	date
▶	"Rajiv Gandhi Intern	2017-05-21
	M Chinnaswamy Stadiu	2017-05-19
	M Chinnaswamy Stadiu	2017-05-17
	Wankhede Stadium	2017-05-16
	Feroz Shah Kotla	2017-05-14
	Maharashtra Cricket	2017-05-14
	Eden Gardens	2017-05-13
	Green Park	2017-05-13
	Feroz Shah Kotla	2017-05-12
	Wankhede Stadium	2017-05-11

```
# --Case (4,6, single,0)
```

```
select distinct batsman,bowler,ball,case when total_runs=1 then 'Single'
      when total_runs=4 then 'Boundry'
      when total_runs=6 then 'Six'
      else 'Duck'
      end as 'Run in words' from ipl.deliveries ;
```

	batsman	bowler	ball	Run in words
	DA Warner	TS Mills	1	Duck
	DA Warner	TS Mills	2	Duck
	DA Warner	TS Mills	3	Boundry
	DA Warner	TS Mills	4	Duck
	DA Warner	TS Mills	5	Duck
	S Dhawan	TS Mills	6	Duck
	S Dhawan	TS Mills	7	Single
	S Dhawan	A Choudhary	1	Single
▶	DA Warner	A Choudhary	2	Boundry
	DA Warner	A Choudhary	3	Single
	DA Warner	A Choudhary	4	Six
	DA Warner	A Choudhary	5	Duck
	MC Henriques	A Choudhary	6	Duck

```
# --Data Aggregation
```

```
select winner,win_by_wickets,max(win_by_runs) from ipl.matches
#where winner='Mumbai Indians'
group by winner
order by 3 desc;
```

	winner	win_by_wickets	max(win_by_runs)
▶	Mumbai Indians	4	146
	Royal <small>Royal Challengers Bangalore</small>		144
	Kolkata Knight Riders	10	140
	Kings XI Punjab	6	111
	Rajasthan Royals	6	105
	Delhi Daredevils	0	97
	Chennai Super Kings	0	97
	Sunrisers Hyderabad	0	85
	Deccan Chargers	10	82
	Rising Pune Supergiant	7	61

```
# --How many extra runs have been conceded in ipl
```

```
select distinct bowler,sum(extra_runs) from ipl.deliveries
group by bowler
having sum(extra_runs)>0;
```

	bowler	sum(extra_runs)
▶	TS Mills	13
	A Choudhary	7
	YS Chahal	68
	S Aravind	50
	SR Watson	162
	STR Binny	19
	A Nehra	115
	B Kumar	166
	BCJ Cutting	4
	Rashid Khan	16
	DJ Hooda	12
	MC Henriques	43

--On an average, teams won by how many runs in ipl

```
select winner,avg(win_by_runs) from ipl.matches
group by winner
having avg(win_by_runs)>0
order by 2 desc;
```

	winner	avg(win_by_runs)
▶	Chennai Super Kings	20.0886
	Mumbai Indians	16.6522
	Royal Challengers Bangalore	16.3836
	Deccan Chargers	14.5172
	Rajasthan Royals	13.0317
	Kings XI Punjab	12.3143
	Rising Pune Supergiant	12.3000
	Kolkata Knight Riders	11.6494
	Pune Warriors	11.5833
	Sunrisers Hyderabad	11.3333
	Rising Pune Supergiants	10.6000

--How many extra runs were conceded in ipl by SK Warne

```
select bowler,sum(extra_runs) from ipl.deliveries
where bowler='SK Warne'
group by bowler;
```

--How many boundaries (4s or 6s) have been hit in ipl

```
select m.winner, d.total_runs,count(d.total_runs) from ipl.deliveries d
inner join ipl.matches m on m.id=d.matchid
where d.total_runs in (4,6)
and m.winner='Mumbai Indians'
```

group by m.winner, d.total_runs;

	winner	total_runs	count(d.total_runs)
▶	Mumbai Indians	4	2507
	Mumbai Indians	6	980

--How many balls did SK Warne bowl to batsman SR Tendulkar

```
select batsman,bowler, count(ball) from ipl.deliveries
where bowler='SK Warne' and batsman='SR Tendulkar'
group by batsman,bowler;
```

	batsman	bowler	count(ball)
▶	SR Tendulkar	SK Warne	39

--How many matches were played in the month of April

```
select count(*) from ipl.matches
where month(date)='4';
select count(*) from ipl.matches
where extract(month from date)=4;
```

	count(*)
▶	297

--How many matches were played in the March and June

```
select count(*) from ipl.matches
where month(date) in ('3','6');
```

	count(*)
▶	31

--Total number of wickets taken in ipl (count not null values)

```
select count(player_dismissed) as 'Wicket' from ipl.deliveries
where player_dismissed <>'';
```

	Wicket
▶	7438


```
select dismissal_kind,player_dismissed, count(*) from ipl.deliveries
where player_dismissed <>""
group by dismissal_kind,player_dismissed
order by 3 desc;
```

	dismissal_kind	player_dismissed	count(*)
▶	caught	SK Raina	94
	caught	RV Uthappa	87
	caught	RG Sharma	83
	caught	G Gambhir	77
	caught	V Kohli	76
	caught	Yuvraj Singh	76
	caught	KD Karthik	71
	caught	V Sehwag	70
	caught	YK Pathan	68
	caught	MS Dhoni	65

--Pattern Match (Like operators % _)

```
select Distinct player_of_the_match from ipl.matches where player_of_the_match
like '%M%';
```

```
select Distinct player_of_the_match from ipl.matches where player_of_the_match
like 'JJ %';
```

```
select distinct player_of_the_match from ipl.matches where player_of_the_match
like 'K_ P%';
```

	player_of_the_match
▶	KA Pollard
	KH Pandya
	KP Pietersen

--How many teams have word royal in it (could be anywhere in the team name, any case)

```
SELECT distinct team1 FROM ipl.matches where lower(team1) like lower('%Royal%');
```

	team1
▶	Royal Challengers Ba
	Rajasthan Royals

--Group by - Maximum runs by which any team won a match per season

```
select season,max(win_by_runs) from ipl.matches
```

```
group by season
```


order by 1;

	season	max(win_by_runs)
▶	2008	140
	2009	92
	2010	98
	2011	111
	2012	86
	2013	130
	2014	93
	2015	138
	2016	144
	2017	146

--Create score card for each match Id

select batting_team,batsman,sum(batsman_runs) from ipl.deliveries

group by batting_team,batsman

order by 3 desc;

	batting_team	batsman	sum(batsman_runs)
▶	Royal Challengers Ba	V Kohli	4423
	Chennai Super Kings	SK Raina	3707
	Royal Challengers Ba	CH Gayle	3175
	Mumbai Indians	RG Sharma	3037
	Kolkata Knight Rider	G Gambhir	3035
	Chennai Super Kings	MS Dhoni	2986
	Royal Challengers Ba	AB de Villiers	2815
	Sunrisers Hyderabad	DA Warner	2579
	Kings XI Punjab	SE Marsh	2489
	Mumbai Indians	AT Rayudu	2416
	Rajasthan Royals	SR Watson	2378

--Top 10 players with max boundaries (4 or 6)

select DISTINCT batsman,count(total_runs) from ipl.deliveries

where total_runs in (4,6)

group by batsman

order by 2 desc

limit 10;

--Top 20 bowlers who conceded highest extra runs

select bowler,sum(extra_runs) as 'highest extra runs' from ipl.deliveries

group by bowler

order by 2 desc

limit 20;

	bowler	highest extra runs
	SL Malinga	268
▶	P Kumar	236
	UT Yadav	186
	RP Singh	181
	Z Khan	169
	I Sharma	167
	B Kumar	166
	DW Steyn	166
	SR Watson	162
	Harbhajan Singh	158
	DJ Bravo	157

--Top 10 wicket takers

select bowler,count(player_dismissed) as NoWicket_Taken,dismissal_kind from
ipl.deliveries

where dismissal_kind <>""

group by bowler

order by NoWicket_Taken desc

limit 10;

	bowler	NoWicket_Taken	dismissal_kind
▶	SL Malinga	170	caught
	A Mishra	142	caught
	DJ Bravo	137	caught
	Harbhajan Singh	136	caught
	PP Chawla	133	caught
	R Vinay Kumar	125	caught
	A Nehra	121	caught
	Z Khan	119	caught
	B Kumar	117	caught
	R Ashwin	110	caught and bowled

--Name and number of wickets by bowlers who have taken more than or equal to
100 wickets in ipl

select bowler,count(player_dismissed) as NoWicket_Taken,dismissal_kind from
ipl.deliveries

where dismissal_kind <>""

group by bowler

having count(player_dismissed) >=100

order by NoWicket_Taken desc

limit 10;

	bowler	NoWicket_Taken	dismissal_kind
▶	SL Malinga	170	caught
	A Mishra	142	caught
	DJ Bravo	137	caught
	Harbhajan Singh	136	caught
	PP Chawla	133	caught
	R Vinay Kumar	125	caught
	A Nehra	121	caught
	Z Khan	119	caught
	B Kumar	117	caught
	R Ashwin	110	caught and bowled

Top 2 player_of_the_match for each season

select season, player_of_the_match, CNT from

(

select row_number() over (partition by season) as

rn,season,player_of_the_match,Cnt

from (

select season,player_of_the_match, count(player_of_the_match)
as Cnt

from ipl.matches

group by season,player_of_the_match

order by 1 asc,3 desc

) rw

) Temp

where Temp.rn<3;

	season	player_of_the_match	CNT
▶	2008	SE Marsh	5
	2008	SR Watson	4
	2009	YK Pathan	3
	2009	SR Tendulkar	2
	2010	SR Tendulkar	4
	2010	JH Kallis	3
	2011	CH Gayle	6
	2011	MEK Hussey	3
	2012	CH Gayle	5
	2012	AB de Villiers	3
	2013	MEK Hussey	5
	2013	A Mishra	4
	2014	GJ Maxwell	4

--Window Functions - (CTE) -- Combine column date from matches with table deliveries to get data by year

*what is CTE ?

A common table expression (CTE) is a named temporary result set that exists within the scope of a single statement and that can be referred to later within that statement, possibly multiple times.

with

t1 as (select id,season,date,city,team1,team2,winner from ipl.matches),

t2 as (select matchid,batting_team,bowling_team from ipl.deliveries)

select distinct

t1.season,t1.date,t1.city,t1.team1,t1.team2,t2.batting_team,t2.bowling_team,
t1.winner

from t1 inner join t2 on t1.id=t2.matchid;

	season	date	city	team1	team2	batting_team	bowling_team	winner
▶	2017	2017-04-05	Hyderabad	Sunrisers Hyderabad	Royal Challengers Ba	Sunrisers Hyderabad	Royal Challengers Ba	Sunrisers Hyderabad
	2017	2017-04-05	Hyderabad	Sunrisers Hyderabad	Royal Challengers Ba	Royal Challengers Ba	Sunrisers Hyderabad	Sunrisers Hyderabad
	2017	2017-04-06	Pune	Mumbai Indians	Rising Pune Supergia	Mumbai Indians	Rising Pune Supergia	Rising Pune Supergiant
	2017	2017-04-06	Pune	Mumbai Indians	Rising Pune Supergia	Rising Pune Supergia	Mumbai Indians	Rising Pune Supergiant
	2017	2017-04-07	Rajkot	Gujarat Lions	Kolkata Knight Rider	Gujarat Lions	Kolkata Knight Rider	Kolkata Knight Riders
	2017	2017-04-07	Rajkot	Gujarat Lions	Kolkata Knight Rider	Kolkata Knight Rider	Gujarat Lions	Kolkata Knight Riders
	2017	2017-04-08	Indore	Rising Pune Supergia	Kings XI Punjab	Rising Pune Supergia	Kings XI Punjab	Kings XI Punjab
	2017	2017-04-08	Indore	Rising Pune Supergia	Kings XI Punjab	Kings XI Punjab	Rising Pune Supergia	Kings XI Punjab