CS 6350

BIG DATA MANAGEMENT AND ANALYTICS

Project Report on

IPL DATA SET ANALYSIS

Team Members :

Revanth Nyalakonda	***rxn162130***
Bharath Reddy Loka	***bxl160630***
SaiKiran Taduri	***sxt161730***
Vineet Varma N.	***vxn161130***

INDEX

1.ABSTRACT:

Cricket is a game that contains a considerable measure of factual information. There is information about batting records, knocking down some pins records, singular player records, scorecard of various matches played, and so forth. This information can be put to appropriate use to foresee the aftereffects of recreations thus this issue has turned into an intriguing issue in this day and age. The greater part of watchers these days attempt to do some kind of expectation at some phase of the competitions to see which group will in the long run win the forthcoming matches and consequently the competition. This report goes for taking care of the issue of anticipating the consequences of amusements by recognizing the imperative characteristics from the informational collection and utilizing the information mining calculations. I have restricted my zone of study to the local Twenty 20 competition which is held in India consistently amid the mid year i.e. the Indian Premier League. The past work that I read and utilized as a source of perspective were either foreseeing diversion comes about for all games as a rule or games like Basketball, Soccer, and so on. My report portrays in detail the distinctive quality determination methods and also the model building calculations used to take care of this issue of result expectation in cricket. I have additionally utilized RMSE as the assessment criteria to assess how well the forecast performs. Some future work is additionally recommended in this report on the off chance that some other understudy later on is occupied with proceeding with the investigation of this issue and enhancing my outcomes.

2.INTRODUCTION:

Cricket is being played in numerous nations all around the globe. There are a considerable measure of residential and worldwide competitions being held in numerous nations which play cricket. Cricket is a diversion played between two groups including 11 players in each group. The outcome is either a win, misfortune or a tie. Be that as it may, in some cases because of awful climate conditions the amusement is additionally washed out as Cricket is a diversion which can't be played in rain. Additionally, this diversion is likewise to a great degree capricious on the grounds that at each phase of the amusement the energy movements to one of the groups between the two. A ton of times the outcome gets this show on the road chosen the last bundle of the match where the amusement gets truly close. Considering all these unusual situations of this erratic amusement, there is an enormous enthusiasm among the onlookers to do some expectation either toward the begin of the diversion or amid the diversion. Numerous onlookers additionally play wagering recreations to win cash. In this way, remembering every one of these potential outcomes, this report goes for concentrate the issue of anticipating the diversion comes about before the amusement has begun in light of the insights and information accessible from the informational collection. The review utilizes the Indian Premier League informational collection of each of the 8 seasons played till now i.e. from 2008 to 2016.

There are distinctive approaches to do the forecast. The expectation should be possible contemplating the player's execution and additionally the group execution. There are numerous unusual things that occur in a cricket diversion like matches being washed out because of rain, a key player getting harmed before the amusement, players changing their groups, and so on. Some of the time a key player additionally gets inured amid the diversion and consequently is not ready to take additionally part in the amusement. Every one of these variables do influence the expectation to some degree. The report talks about a philosophy that I took after for the amusement result expectation. The procedure comprises of first the quality choice calculations which trim down the rundown of ascribes to just essential ones and after that the information mining calculations which can be connected on those properties. The amusement forecast issue that I am examining does not think about the player's execution but rather it takes into thought the group's past execution at an abnormal state degree alongside alternate variables like hurl champ, hurl choice, home support, and so on.We're lovingly calling this "machine learning magic," yet it's not new. This is one of the quickest approaches to assemble reasonable instinct around machine learning. The objective is to build out of the box models and apply them to various datasets. This project is built behind these reasons:
Primarily, you'll assemble instinct for model-to-issue fit. Which models are strong to missing information. Which models handle clear cut components well.
The next important factor in this project will show you the important ability of prototyping models rapidly. In this present reality, it's regularly hard to know which model will perform best without just attempting them.

This report introduces a Gradient Boosting Predictor display for foreseeing the results of the matches in IPL 9 being played in IPL. The model has a few elements that are gotten from the informational collection segments which depend on diverse contemplations developing out of a more profound examination of T20 cricket. The models are made using Data Analytics systems from machine learning space. The precision measure is furthermore used for the Prediction demonstrate. Additionally change in the model can be attempted by using a greater test and preparing information than the one that has been utilized as a part of this work.

3.PROBLEM STATEMENT:

In this project we are trying to build a model that predicts the out come of the match by using the related features that are extracted from the dataset. It is an attempt to predict the win probability of the teams in a given match at the end of each over and to look at the important factors affecting the match output. This model objective is to predict the win probability of a team at the end of each over for the finals of IPL season 2016. Training and test data were given accordingly like the training data goes with all the matches that are played during the 2016 season and test data corresponds to the final match which is used for predicting the results.

4.RELATED WORK:

There has been a considerable measure of related review to this issue in different distinctive games. The papers I have utilized as references are altogether related work that had been done on this issue. The paper by Trawinski [1] depicted the forecast of results utilizing a fluffy arrangement framework. This paper was anticipating the outcomes for ball games. The approach that I utilized as a part of the venture does not consider the outer elements like player harm, exhaustion or the triumphant streak. My informational index contains more information about the matches and the occasions occurring in the match like hurl and player of match as opposed to the information about outer variables.

4.1 METHODLOGY: I have taken after the accompanying strategy over the span of my venture. The system comprises of 5 distinct stages as appeared in Figure 1 i.e. Informational collection Generation, Data Cleaning, Attribute Selection, Model Building and Analysis of Results.

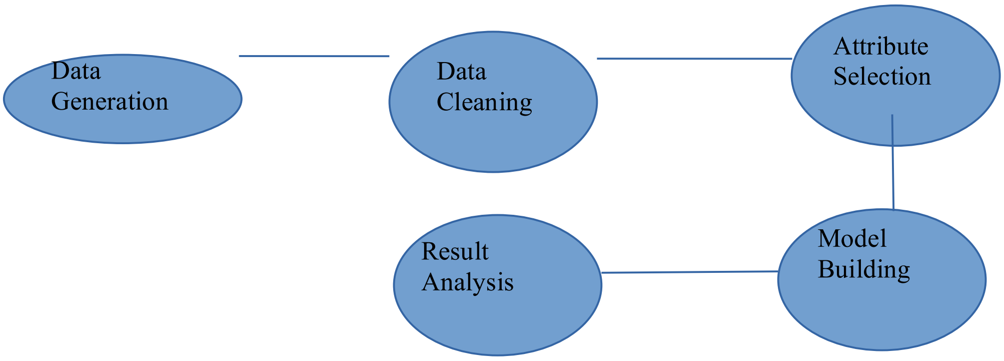


Figure 1: Project Methodology

5.DATASET DESCRIPTION:

5.1 DATASET DETAILS AND GENERATION:

The data was collected from the https://www.kaggle.com/manasgarg/ipl [2] website. The site has information about every one of the 8 seasons (from 2008 to 2016) of residential competitions held in India i.e. the Indian Premiere League. It is a 20-20 arrangement of competition. It implies that each group bats or bowls for most extreme 20 overs each and the consequence of the diversion is chosen toward the finish of the aggregate 40 overs played. The informational index downloaded from this site was of two arrangements; one was the ball by ball detail of each match in the .csv design. The .csv documents had information around 577 diverse matches. The other file has many large number of instances around 1,36,598.

We are Planning to use the data set for some predictions and data analysis with deep exploration into the data of the Indian Premier League. We use some classifier models for result predictions. Primary data exploration is being to done to play with data and get familiar with it.

We have 2 data files

1.Deliveries.csv- 136598 instances

2.Macthes.csv- 577 instances

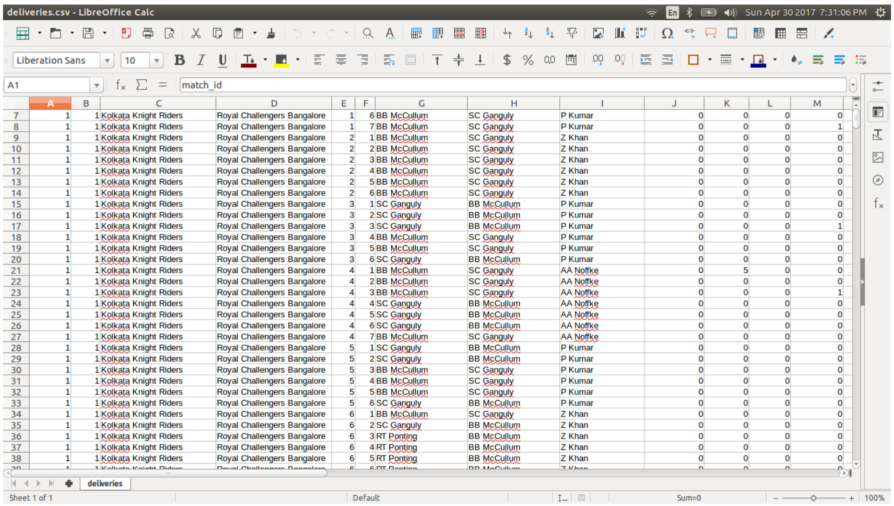
5.2 DATA DISTRIBUTION:

1. Deliveries.csv : ("Match_Id", "Inning", "Batting_Team", "Bowling_team", "Over", "Ball", "Batsmen", "Non- Striker", "Bowler", "is_super_over", "wide_runs", "bye_runs", "leg-bye_runs", "penalty_runs", "batsmen_runs", "extra_runs", "total_runs", "player_dismissed", "Dismissal_kind", "Fielder")

2. Matches.csv : ("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")

5.3 DATASET SAMPLE SCREEN SHOTS:

***Deliveries.csv:**



deliveries.csv - LibreOffice Calc

Sun Apr 30 2017 7:31:06 PM

Liberation Sans 10

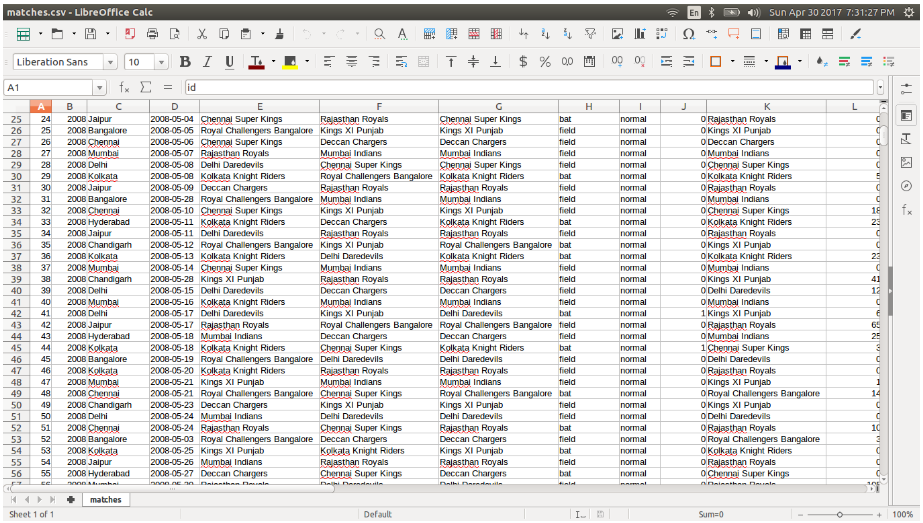
A1 match_id

	A	B	C	D	E	F	G	H	I	J	K	L	M
7	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	6	BB McCullum	SC Ganguly	P Kumar	0	0	0	0
8	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	1	7	BB McCullum	SC Ganguly	P Kumar	0	0	0	1
9	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	2	1	BB McCullum	SC Ganguly	Z Khan	0	0	0	0
10	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	2	2	BB McCullum	SC Ganguly	Z Khan	0	0	0	0
11	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	2	3	BB McCullum	SC Ganguly	Z Khan	0	0	0	0
12	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	2	4	BB McCullum	SC Ganguly	Z Khan	0	0	0	0
13	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	2	5	BB McCullum	SC Ganguly	Z Khan	0	0	0	0
14	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	2	6	BB McCullum	SC Ganguly	Z Khan	0	0	0	0
15	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	3	1	SC Ganguly	BB McCullum	P Kumar	0	0	0	0
16	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	3	2	SC Ganguly	BB McCullum	P Kumar	0	0	0	0
17	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	3	3	SC Ganguly	BB McCullum	P Kumar	0	0	0	1
18	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	3	4	BB McCullum	SC Ganguly	P Kumar	0	0	0	0
19	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	3	5	BB McCullum	SC Ganguly	P Kumar	0	0	0	0
20	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	3	6	SC Ganguly	BB McCullum	P Kumar	0	0	0	0
21	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	4	1	BB McCullum	SC Ganguly	AA Noffke	0	5	0	0
22	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	4	2	BB McCullum	SC Ganguly	AA Noffke	0	0	0	0
23	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	4	3	BB McCullum	SC Ganguly	AA Noffke	0	0	0	1
24	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	4	4	SC Ganguly	BB McCullum	AA Noffke	0	0	0	0
25	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	4	5	SC Ganguly	BB McCullum	AA Noffke	0	0	0	0
26	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	4	6	SC Ganguly	BB McCullum	AA Noffke	0	0	0	0
27	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	4	7	BB McCullum	SC Ganguly	AA Noffke	0	0	0	0
28	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	5	1	SC Ganguly	BB McCullum	P Kumar	0	0	0	0
29	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	5	2	SC Ganguly	BB McCullum	P Kumar	0	0	0	0
30	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	5	3	BB McCullum	SC Ganguly	P Kumar	0	0	0	0
31	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	5	4	BB McCullum	SC Ganguly	P Kumar	0	0	0	0
32	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	5	5	BB McCullum	SC Ganguly	P Kumar	0	0	0	0
33	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	5	6	SC Ganguly	BB McCullum	P Kumar	0	0	0	0
34	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	6	1	BB McCullum	SC Ganguly	Z Khan	0	0	0	0
35	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	6	2	SC Ganguly	BB McCullum	Z Khan	0	0	0	0
36	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	6	3	RT Ponting	BB McCullum	Z Khan	0	0	0	0
37	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	6	4	RT Ponting	BB McCullum	Z Khan	0	0	0	0
38	1	1	Kolkata Knight Riders	Royal Challengers Bangalore	6	5	RT Ponting	BB McCullum	Z Khan	0	0	0	0

deliveries

Sheet 1 of 1 Default Sum=0 100%

***Matches.csv:**



matches.csv - LibreOffice Calc

Sun Apr 30 2017 7:31:27 PM

Liberation Sans 10

A1 id

	A	B	C	D	E	F	G	H	I	J	K	L
25	24	2008	Jaipur	2008-05-04	Chennai Super Kings	Rajasthan Royals	Chennai Super Kings	bat	normal	0	Rajasthan Royals	0
26	25	2008	Bangalore	2008-05-05	Royal Challengers Bangalore	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	0
27	26	2008	Chennai	2008-05-06	Chennai Super Kings	Deccan Chargers	Deccan Chargers	field	normal	0	Deccan Chargers	0
28	27	2008	Mumbai	2008-05-07	Rajasthan Royals	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0
29	28	2008	Delhi	2008-05-08	Delhi Daredevils	Chennai Super Kings	Chennai Super Kings	field	normal	0	Chennai Super Kings	0
30	29	2008	Kolkata	2008-05-08	Kolkata Knight Riders	Royal Challengers Bangalore	Kolkata Knight Riders	bat	normal	0	Kolkata Knight Riders	5
31	30	2008	Jaipur	2008-05-09	Deccan Chargers	Rajasthan Royals	Rajasthan Royals	field	normal	0	Rajasthan Royals	0
32	31	2008	Bangalore	2008-05-28	Royal Challengers Bangalore	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0
33	32	2008	Chennai	2008-05-10	Chennai Super Kings	Kings XI Punjab	Kings XI Punjab	field	normal	0	Chennai Super Kings	18
34	33	2008	Hyderabad	2008-05-11	Kolkata Knight Riders	Deccan Chargers	Kolkata Knight Riders	bat	normal	0	Kolkata Knight Riders	23
35	34	2008	Jaipur	2008-05-11	Delhi Daredevils	Rajasthan Royals	Rajasthan Royals	field	normal	0	Rajasthan Royals	0
36	35	2008	Chandigarh	2008-05-12	Royal Challengers Bangalore	Kings XI Punjab	Royal Challengers Bangalore	bat	normal	0	Kings XI Punjab	0
37	36	2008	Kolkata	2008-05-13	Kolkata Knight Riders	Delhi Daredevils	Kolkata Knight Riders	bat	normal	0	Kolkata Knight Riders	23
38	37	2008	Mumbai	2008-05-14	Chennai Super Kings	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0
39	38	2008	Chandigarh	2008-05-28	Kings XI Punjab	Rajasthan Royals	Rajasthan Royals	field	normal	0	Kings XI Punjab	41
40	39	2008	Delhi	2008-05-15	Delhi Daredevils	Deccan Chargers	Deccan Chargers	field	normal	0	Delhi Daredevils	12
41	40	2008	Mumbai	2008-05-16	Kolkata Knight Riders	Mumbai Indians	Mumbai Indians	field	normal	0	Mumbai Indians	0
42	41	2008	Delhi	2008-05-17	Delhi Daredevils	Kings XI Punjab	Delhi Daredevils	bat	normal	1	Kings XI Punjab	6
43	42	2008	Jaipur	2008-05-17	Rajasthan Royals	Royal Challengers Bangalore	Royal Challengers Bangalore	field	normal	0	Rajasthan Royals	65
44	43	2008	Hyderabad	2008-05-18	Mumbai Indians	Deccan Chargers	Deccan Chargers	field	normal	0	Mumbai Indians	25
45	44	2008	Kolkata	2008-05-18	Kolkata Knight Riders	Chennai Super Kings	Kolkata Knight Riders	bat	normal	1	Chennai Super Kings	3
46	45	2008	Bangalore	2008-05-19	Royal Challengers Bangalore	Delhi Daredevils	Delhi Daredevils	field	normal	0	Delhi Daredevils	0
47	46	2008	Kolkata	2008-05-20	Kolkata Knight Riders	Rajasthan Royals	Rajasthan Royals	field	normal	0	Rajasthan Royals	0
48	47	2008	Mumbai	2008-05-21	Kings XI Punjab	Mumbai Indians	Mumbai Indians	field	normal	0	Kings XI Punjab	1
49	48	2008	Chennai	2008-05-21	Royal Challengers Bangalore	Chennai Super Kings	Royal Challengers Bangalore	bat	normal	0	Royal Challengers Bangalore	14
50	49	2008	Chandigarh	2008-05-23	Deccan Chargers	Kings XI Punjab	Kings XI Punjab	field	normal	0	Kings XI Punjab	0
51	50	2008	Delhi	2008-05-24	Mumbai Indians	Delhi Daredevils	Delhi Daredevils	field	normal	0	Delhi Daredevils	0
52	51	2008	Chennai	2008-05-24	Rajasthan Royals	Chennai Super Kings	Rajasthan Royals	bat	normal	0	Rajasthan Royals	10
53	52	2008	Bangalore	2008-05-03	Royal Challengers Bangalore	Deccan Chargers	Deccan Chargers	field	normal	0	Royal Challengers Bangalore	3
54	53	2008	Kolkata	2008-05-25	Kings XI Punjab	Kolkata Knight Riders	Kings XI Punjab	bat	normal	0	Kolkata Knight Riders	0
55	54	2008	Jaipur	2008-05-26	Mumbai Indians	Rajasthan Royals	Rajasthan Royals	field	normal	0	Rajasthan Royals	0
56	55	2008	Hyderabad	2008-05-27	Deccan Chargers	Chennai Super Kings	Deccan Chargers	bat	normal	0	Chennai Super Kings	0

matches

Sheet 1 of 1 Default Sum=0 100%

5.4 DATA CLEANING AND PREPROCESSING:

The data obtained from Kaggle repository was at that point cleaned. In this way, I didn't need to do any kind of cleaning on the information. Be that as it may, I needed to handle the missing qualities information and the information for the matches which were washed out because of rain. There were 7 coordinates whose information were absent in the .csv records.

In this analysis, we are going to look at the matches played only during the latest season 2016. So let us subset the dataset to get only these rows. Also some matches affected by rain and hence Duckworth-Lewis method are used for these matches and so using these matches for our training model might cause some error in our training and so let us neglect those matches as well.

5.5 CODE FOR THE ABOVE PURPOSE:

**Let us take only the matches played in 2016 for this analysis*

```
match_df = match_df.ix[match_df.season==2016,:]

match_df = match_df.ix[match_df.dl_applied == 0,:]

match_df.head() /* to show the resulting data set */
```

5.6 EXTRACTING FEATURES: The informational collection acquired in the wake of taking care of the missing qualities must be sifted with the assistance of the characteristic determination calculation. Since, there were 21 credits it was important to distinguish all the critical qualities which would be helpful for the model building undertakings. The paper by Haghighat [3] clarified the quality end handle. Here, initial an informational index with some number of characteristics is chosen and afterward each trait is disposed of one by one from the arrangement of the properties. A property is totally wiped out if the exactness enhances after its expulsion or else the characteristic is kept in the informational collection. Along these lines, toward the finish of this disposal procedure we get an arrangement of qualities utilizing which we get the most noteworthy exactness of forecast for the grouping calculations. We have separated a few elements for building our model to anticipate the outcomes a portion of the elements that we extricated from the information are as per the following

1.Wickets taken in the last over
2.Total score of the innings
3.Total wickets
4.Target that the team is chasing down
5.Remaining target
6.Run rate
7.Required run rate
8.Difference between run rate and required run rate
9.Binary variables on whether the team for which we are predicting is batting team or bowling team
10.Runs scored in the last Over

5.7CODE FOR EXTRACTING FEATURES:

```
# runs and wickets per over #
score_df = pd.merge(score_df, match_df[['id','season', 'winner', 'result', 'dl_applied', 'team1', 'team2']], left_on='match_id',
right_on='id')
score_df.player_dismissed.fillna(0, inplace=True)
score_df['player_dismissed'].ix[score_df['player_dismissed'] != 0] = 1
train_df = score_df.groupby(['match_id', 'inning', 'over', 'team1', 'team2', 'batting_team', 'winner'])[['total_runs',
'player_dismissed']].agg(['sum']).reset_index()
train_df.columns = train_df.columns.get_level_values(0)

# innings score and wickets #
train_df['innings_wickets'] = train_df.groupby(['match_id', 'inning'])['player_dismissed'].cumsum()
train_df['innings_score'] = train_df.groupby(['match_id', 'inning'])['total_runs'].cumsum()
train_df.head()

# Get the target column #
temp_df = train_df.groupby(['match_id', 'inning'])['total_runs'].sum().reset_index()
temp_df = temp_df.ix[temp_df['inning']==1,:]
temp_df['inning'] = 2
temp_df.columns = ['match_id', 'inning', 'score_target']
train_df = train_df.merge(temp_df, how='left', on = ['match_id', 'inning'])
train_df['score_target'].fillna(-1, inplace=True)

# get the remaining target #
def get_remaining_target(row):
   if row['score_target'] == -1.:
      return -1
   else:
```

```
train_df['remaining_target'] = train_df.apply(lambda row: get_remaining_target(row),axis=1)

# get the run rate #
train_df['run_rate'] = train_df['innings_score'] / train_df['over']

# get the remaining run rate #
def get_required_rr(row):
   if row['remaining_target'] == -1:
      return -1.
   elif row['over'] == 20:
      return 99
   else:
      return row['remaining_target'] / (20-row['over'])

train_df['required_run_rate'] = train_df.apply(lambda row: get_required_rr(row), axis=1)

def get_rr_diff(row):
   if row['inning'] == 1:
      return -1
   else:
      return row['run_rate'] - row['required_run_rate']

train_df['runrate_diff'] = train_df.apply(lambda row: get_rr_diff(row), axis=1)
train_df['is_batting_team'] = (train_df['team1'] == train_df['batting_team']).astype('int')
train_df['target'] = (train_df['team1'] == train_df['winner']).astype('int')

train_df.head()
```

6.PROPOSED SOLUTION MODEL:

"Gradient Boosting"[4]: We are creating this model using MLLib Library and classifier algorithms to predict the team winning percentage by the end of the season or the match. Training data and the testing data depends on the season and match we are predicting so it's not constant but for the "gradient boosting" we give all the matches that are played in the whole season to predict the winner of the final match.

A feeble speculation or weak learner is characterized as one whose execution is in any event somewhat superior to arbitrary possibility. Theory boosting was separating perceptions, leaving those perceptions that the feeble learner can deal with and concentrating on growing new powerless figures out how to deal with the staying troublesome perceptions. Forecasts are made by dominant part vote of the feeble learners' expectations, weighted by their individual precision. The best type of the AdaBoost calculation was for paired grouping issues and was called AdaBoost.M1. This class of calculations were depicted as a phase shrewd added substance display. This is on the grounds that one new feeble learner is included at once and existing frail learners in the model are solidified and left unaltered. Choice trees are utilized as the frail learner in angle boosting. Particularly relapse tress are utilized that yield genuine qualities for parts and whose yield can be included, permitting ensuing models yields to be included and "remedy the residuals in the predictions.Trees are developed in an eager way, picking the best split focuses in light of immaculateness scores to limit the misfortune.

XGBoost is a calculation that has as of late been commanding connected machine learning and Kaggle rivalries for organized or unthinkable information. XGBoost is an execution of inclination helped choice tress intended for speed and execution. XGBoost is utilized for directed learning issues, where we utilize the preparation information with numerous elements (Xi) to anticipate an objective variable (Yi)

6.1 Coding language and preliminary results:

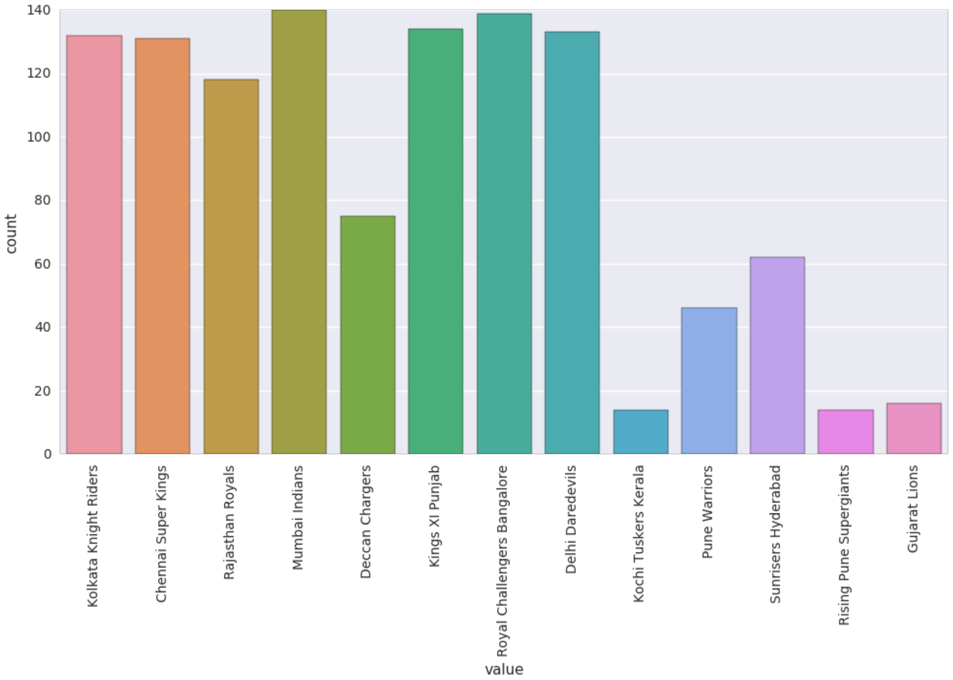
***Python**

We started using python for data exploration and we are just exploring the data set so the preliminary results are retrieved from the data before we started to build the model and henceforth the results that were obtained were showcased completely with the code snippets.

6.2 Number of matches played by each team:

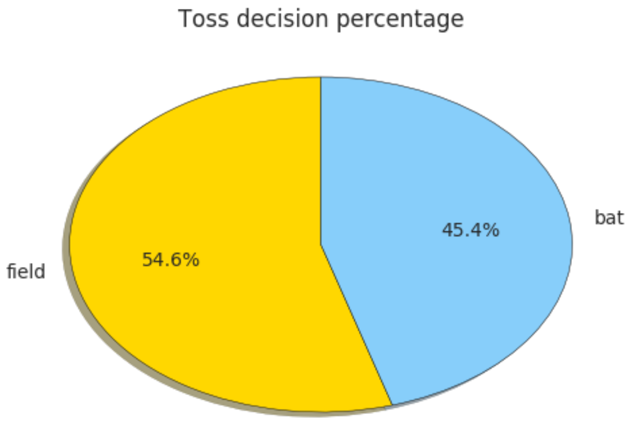
```
temp_df = pd.melt(match_df, id_vars=['id','season'], value_vars=['team1', 'team2'])
```

```
sns.countplot(x='value', data=temp_df)
plt.xticks(rotation='vertical')
display(plt.show())
```



6.3 TOSS DECISIONS BY EACH TEAM:

```
temp_series = match_df.toss_decision.value_counts()
labels = (np.array(temp_series.index))
sizes = (np.array((temp_series / temp_series.sum())*100))
colors = ['gold', 'lightskyblue']
plt.pie(sizes, labels=labels, colors=colors,
        autopct='%1.1f%%', shadow=True, startangle=90)
plt.title("Toss decision percentage")
display(plt.show())
```



6.4 PERFORMANCE METRIC:

The performance measure for this model is RMSE, root mean squared error as defined below

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

7.STEPS TO CREATE THE MODEL AND PSEUDO CODE FOR GRADIENT BOOSTING ALGORITHM:

7.1 STEPS TO CREATE THE MODEL:

- ✓ Import the necessary modules
- ✓ Load the dataset and look at the top few rows
- ✓ Apply the preprocessing techniques
- ✓ Extracting features for the model
- ✓ Create custom function using XGBoost package
- ✓ Create the feature map
- ✓ Create the function for labeling
- ✓ Give the important features more weight
- ✓ Output the results

Next Steps:

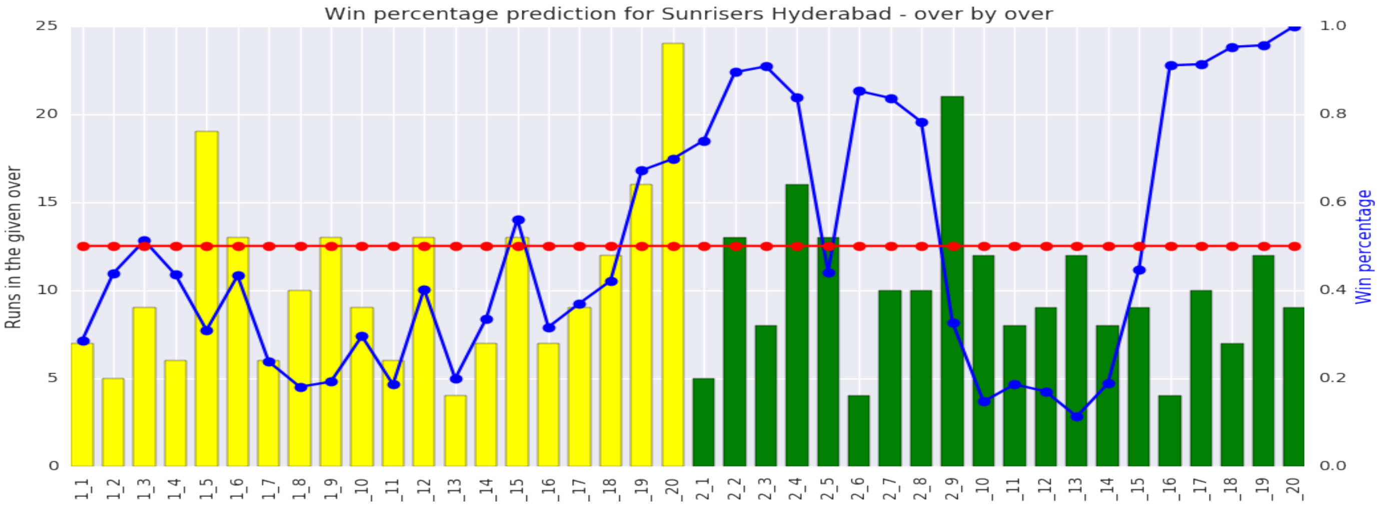
7.2 PSEUDO CODE FOR THE ALGORITHM:

- ✓ initialize list of weak learners to a singleton list with simple prior
- ✓ for each round in 1 to numRounds:
- ✓ re weight examples (x,y) to (x1,y1) by up weighting examples that the existing forest poorly predicts
- ✓ estimate the new weak classifier on weighted examples
- ✓ compute weight of the new weak classifier
- ✓ add the pair to the forest
- ✓ return forest

8.RESULTS EXPLORATION:

The prediction graph shows the result of the model that we predict:

As we can see, team batting first has scored lot of runs in the last two overs (16 and 24), which clearly gave them an edge. Also the scoring rate of team batting second was low in the first 8 overs which made the win probability to hover over 0.5. Then 9th over changed the dynamics since second batting team scored 21 runs in that over clearly giving them an edge. Wickets that went away in the 13th to 15th overs helped first batting team increase the win percentage. In 16th over first batting team conceded only 4 runs which shifted the game in their favor.



8.1 ROOT MEAN SQUARED ERROR:

```
avg= reduce(lambda x, y: x + y, preds) / len(preds)

lz=[avg] * 39

from sklearn.metrics import mean_squared_error

RMSE = mean_squared_error(lz, preds)**0.5

print RMSE
```

RESULT: 0.274287642129 Which means the model is predicting with 74% accuracy which is good.

9.CONCLUSION:

This Dataset examination has given us few highlights and realities about the issue we have within reach. Above all else, it is basic to recall that we don't have qualities for the data which are being alluded to in the whole Dataset. With the association of the qualities and the portrayal the method for taking care of would have been distinctive. As this is out of our extension and as this hasn't been given too, the approach was truly straight forward. The dataset gives the clear idea without much of missing values and outliers hence the preprocessing has also become very easier. The model we have used is generally used in many kaggle competitions as it is based on many weak classifiers to build as strong one. The RMSE is the measure of how the model performs accurately given the data with the extracted features. The accuracy of 74 shows that more can be done about the features extraction and many better features can be used to get more accuracy.

10.REFERENCES:

1.Trawinski, Krzysztof. "A fuzzy classification system for prediction of the results of the basketball games." Fuzzy Systems (FUZZ), 2010 IEEE International Conference on. IEEE, 2010.

2. **https://www.kaggle.com/manasgarg/ipl (Kaggle Repository)**

3.Haghighat, Maral, Hamid Rastegari, and Nasim Nourafza. "A review of data mining techniques for result prediction in sports." Advances in Computer Science: an International Journal 2.5 (2013): 7-12.

4.http://www.ccs.neu.edu/home/vip/teach/MLcourse/4_boosting/slides/gradient_boosting.pdf