Design Document for the Analysis of European Soccer Dataset

Contact Persons:

(TEAM 6)
Himaja Vadaga
Sandeep Kumar Bethi
Bryce Brako

Version Control:

Date	Version	Description	Changed by	Reviewed by
08/19/2016	1.0	First Draft	Bryce Brako	Himaja Vadaga Sandeep Kumar Bethi

1. Preface

This document is created to describe the design, development and deployment procedures taken to perform the analysis of the European Soccer Dataset. It will illustrate the design of the algorithm, the web service and the user interface in the design document.

2. INTRODUCTION

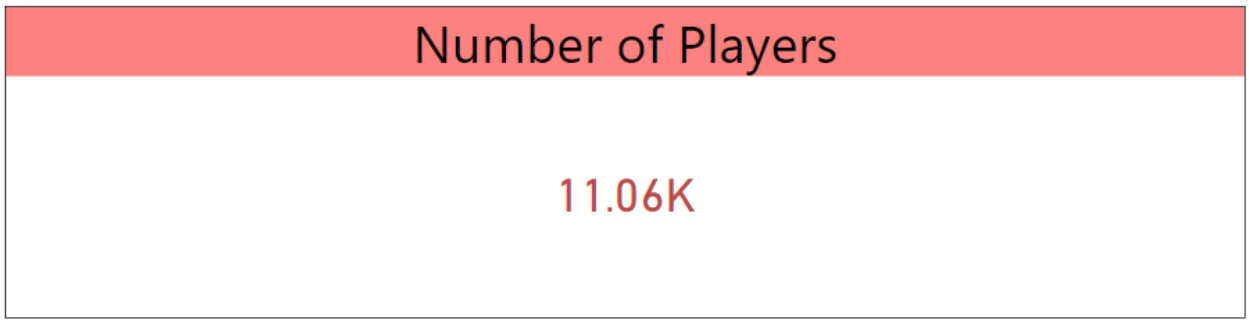
This dataset was taken from Kaggle. It has multiple csv files. The csv files named as Player stats and Match had maximum data needed for the analysis.

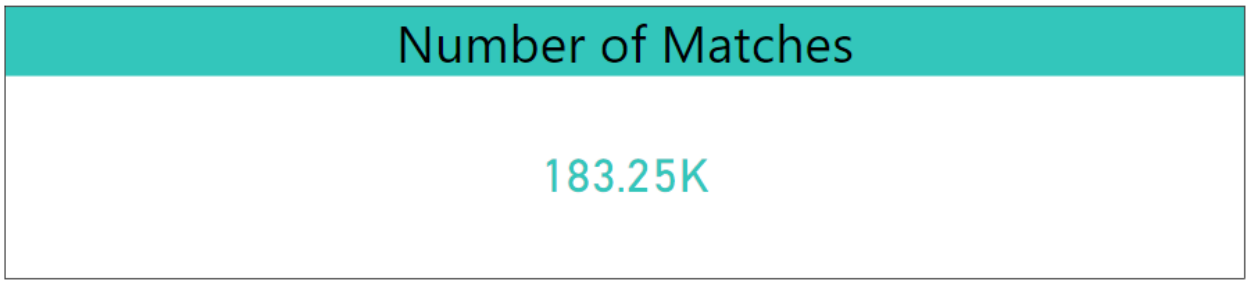
3. About the Dataset

- Till now we have worked with only single csv files during our assignments. But for this project we had a dataset with multiples csv files so our major task was to join these files or tables in an appropriate manner where we could do some useful analysis and build models accordingly.
- The tables were as follows:
    A) Player Stats: This table contained all the stats of different players recorded at different times
    B) Player: This table had data related to player information like the birthday, Player ID
    C) Match: This table had data related to matches from different leagues and different seasons. It contained results of matches from 2008-2009 season to 2015-2016 season. It also had data related to betting odds from different websites
    D) League: This table had data related to different leagues like the league ID's and league names
    E) Country: This table had data related to different countries like to which league they belong and their ID's
    F) Team ID: This table had data related to different teams like the countries they belong to and also the leagues in which they are part of.
- We had majority of the data useful for analysis in Match and player stats table.
- These two tables had missing values and the data type were given wrong for many columns
- There were about 24000 rows and 115 columns in the match table. Among this there was about 3% of missing data.
- Since this dataset is a cross sectional dataset with each record independent of another record. We deleted the missing records which constituted to about 3% of total data.
- The match dataset is used for building the model for classifying the result of the match whether it is a home win or draw or away win
- The player stats dataset is used for building the player valuation model which gives the range of price of a player that one could spend on buying or selling a player
- We made the season, league, stage columns as categorical features to improve the accuracy
- We performed feature engineering by building additional columns such as day, moth and year
- We also performed feature engineering for calculating age column
- We performed joins of two tables player stats and player in order to calculate the age of different players in player stats table
- Age feature was a very important feature in determining the price of the player. This age feature wasn't present readily in the dataset.
- We made use of eeptools package from R in order to build the age feature
- We combined the different tables such as player stats, player by making use of player_api_id feature
- Similarly we made use of Team_ID in joining datasets of match and team tables
- We made use of filter based feature selection module present in Azure ML studio for feature selection
- From the results we got from feature selection we went ahead and built the model.

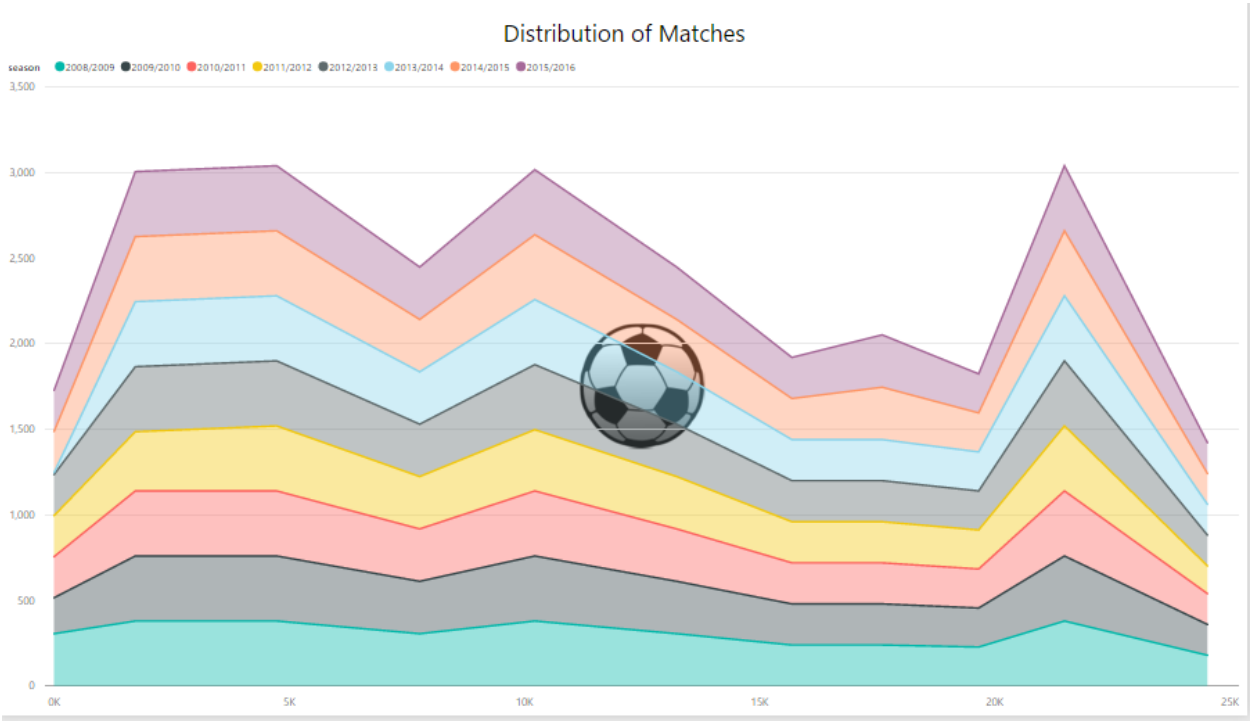
4. Analysis of the Raw Data

Below are the screenshots of visualizations done in Power BI for the initial analysis of the dataset.

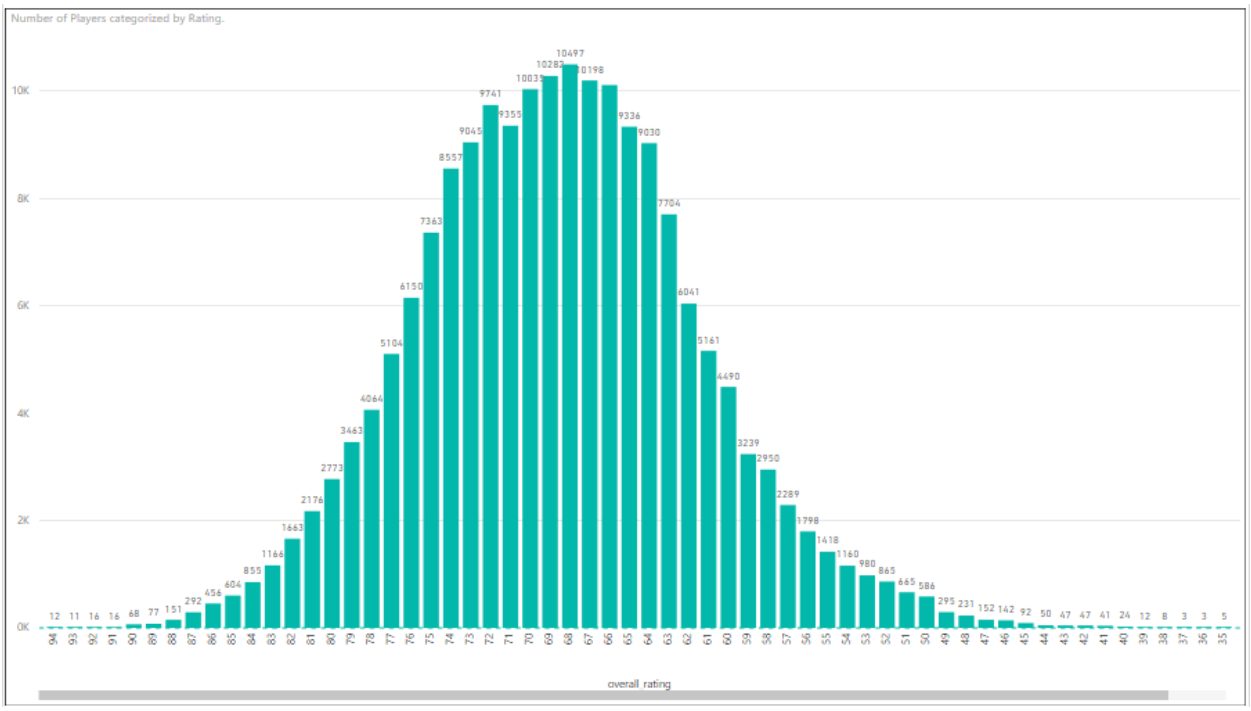




The screenshot above describes the number of players and its statistics we were performing our analysis on and the number of matches played from years 2008 to 2016.



The screenshot above gives the distribution of matches over the years.



The above screenshot gives the distribution of players with respect to their overall rating.

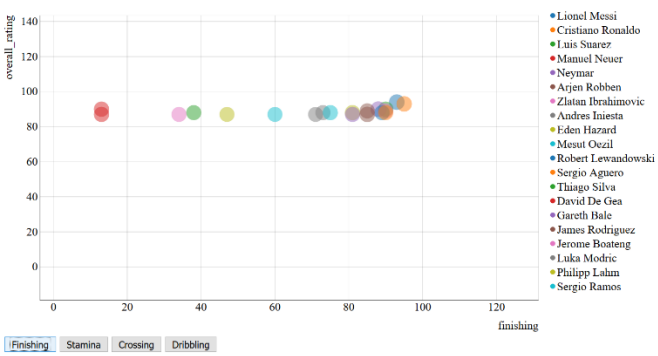
Visualizations have been made using D3.js and R to communicate information clearly and efficiently via statistical graphics, plots and information graphics. Numerical data may be encoded using dots, lines, or bars, to visually communicate a quantitative message. Effective visualization helps users analyze and reason about data and evidence. It makes complex data more accessible, understandable and usable. Users may have particular analytical tasks, such as making comparisons or understanding causality, and the design principle of the graphic (i.e., showing comparisons or showing causality) follows the task.

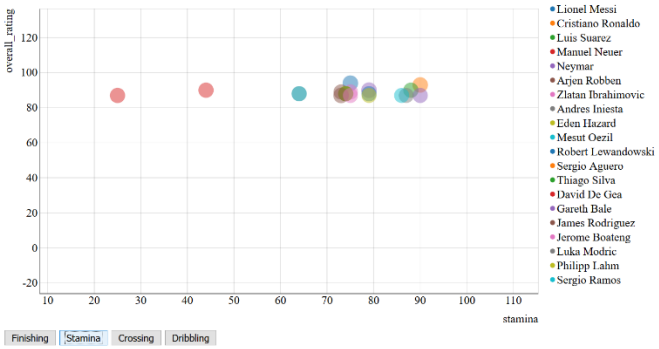
5. D3.JS Visualizations:

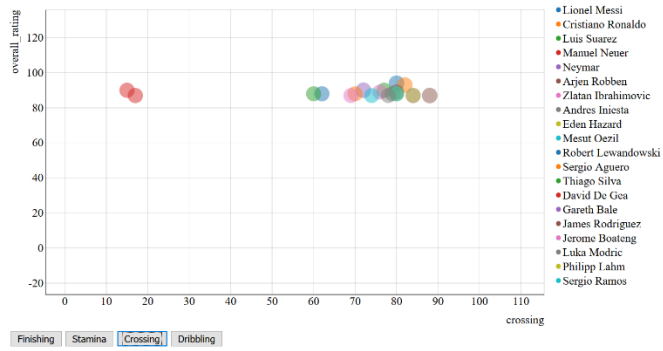
D3.js helps you bring data to life using HTML, SVG, and CSS. D3's emphasis on web standards gives you the full capabilities of modern browsers without tying yourself to a proprietary framework, combining powerful visualization components and a data-driven approach to DOM manipulation.

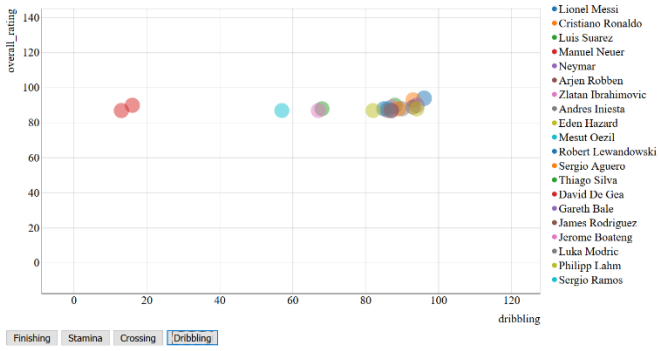
Two visualizations:

1. **Zoomable Scatterplot**









Zoomable scatterplot lets you to look at the point closely to find the relation with other points in the plot. Here, you can scroll in and out to see the difference and compare the points on the plot.

Dataset:

The dataset is refined dataset by merging two table player and player_stats. Selecting top20 players based on the overall ratings keeping the date of the record in consideration.

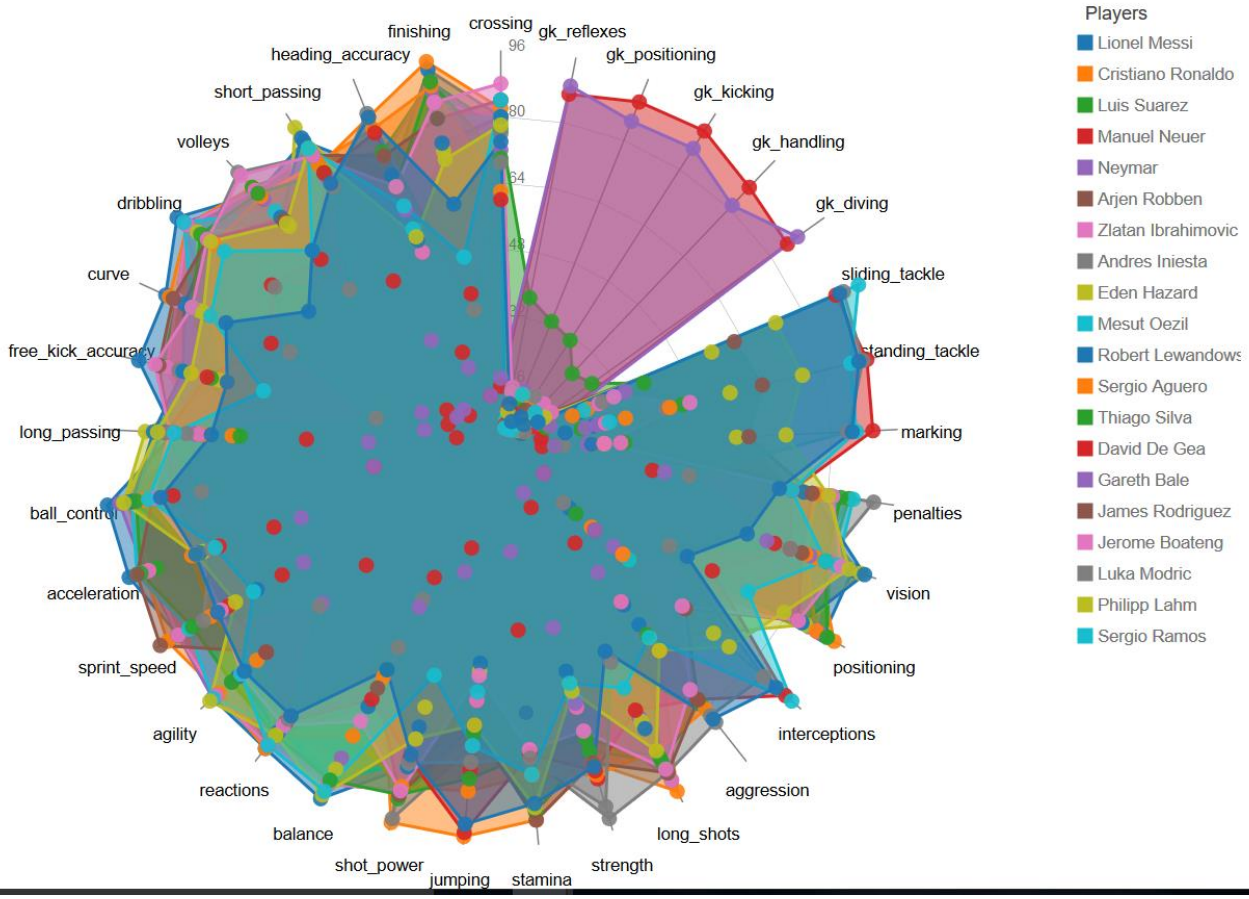
Explanation:

- Visualizing skillset of top 20 players over graph where y-axis is the overall_rating and x-axis changes with the button clicks.
- It features gk_reflexes, finishing, stamina, crossing and dribbling.
- With the clicks it depicts the player's performance with respect to each other and with the specific skill.

Business Value:

- Higher the value of gk_reflexes means better the player has the chances of being a good goal keeper
- Higher the value of finishing means better the player has the chances of being a good striker
- Higher the values of stamina, crossing and dribbling helps the manager to choose the player wisely for a better team formation

2. Radar Chart



To see a complete profile of individual players I create a radar chart. This is a great way to plot multiple variables. Data first had to be transformed to work. This makes them useful for seeing which variables have similar values or if there are any outliers amongst each variable. Radar Charts are also useful for seeing which variables are scoring high or low within a dataset, making them ideal for displaying performance

Dataset:

The dataset is refined dataset by merging two table player and player_stats. Selecting top20 players based on the overall ratings keeping the date of the record in consideration.

Explanation:

- Each variable is provided an axis that starts from the center.
- Multivariate data can be displayed and compared with low/high values.
- Helps to validate the performance of each player separately.
- All axes are arranged radially, with equal distances between each other, while maintaining the same scale between all axes.

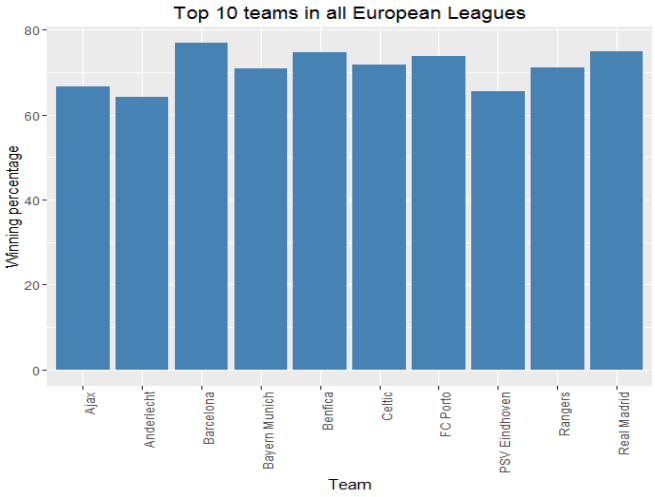
Business Value:

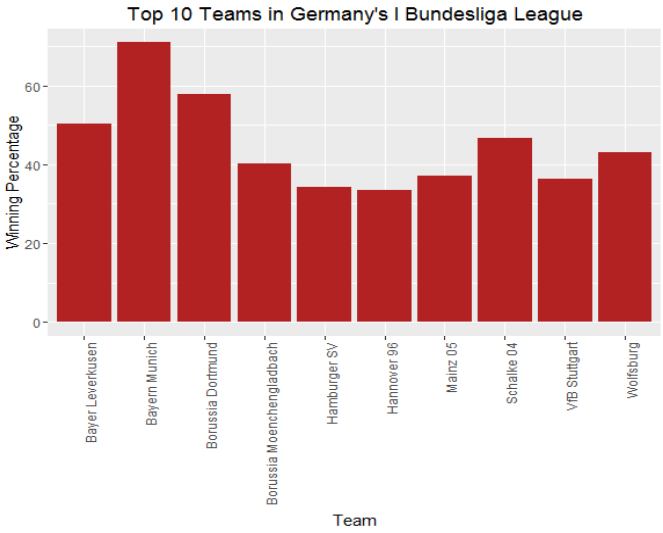
- Displaying 33 skillset of each of the top 20 players in a manner of a polygon.
- Upon hovering on one polygon the data points show how radically the players performance changes with each skillset

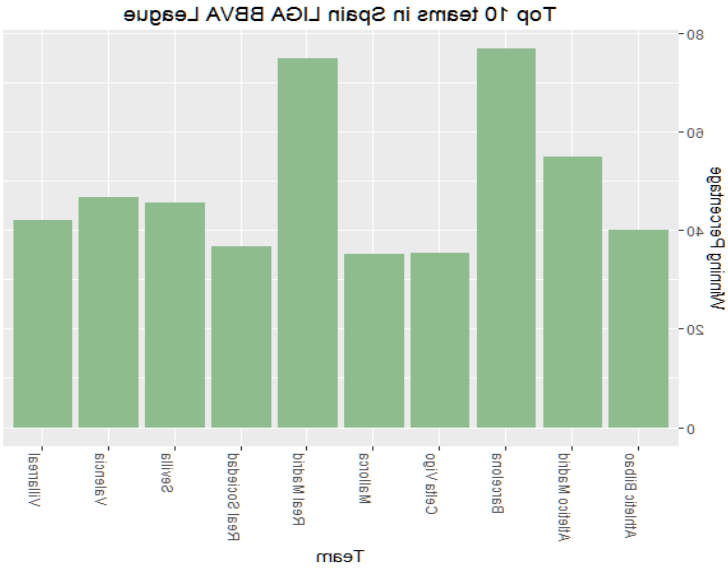
- The largest area swept by the polygon describes as the best player when compared with all the other players.

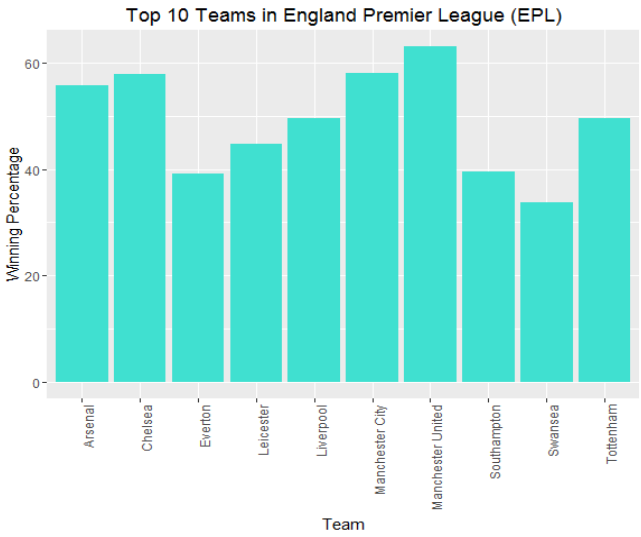
R Visualization

This visualization is build using R language to predict the top ten teams in all the premier leagues.









Data Preprocessing

- Count the number of matches the particular team has played at home
- Combining two data frames consisting of home_match and away_match

Business Value

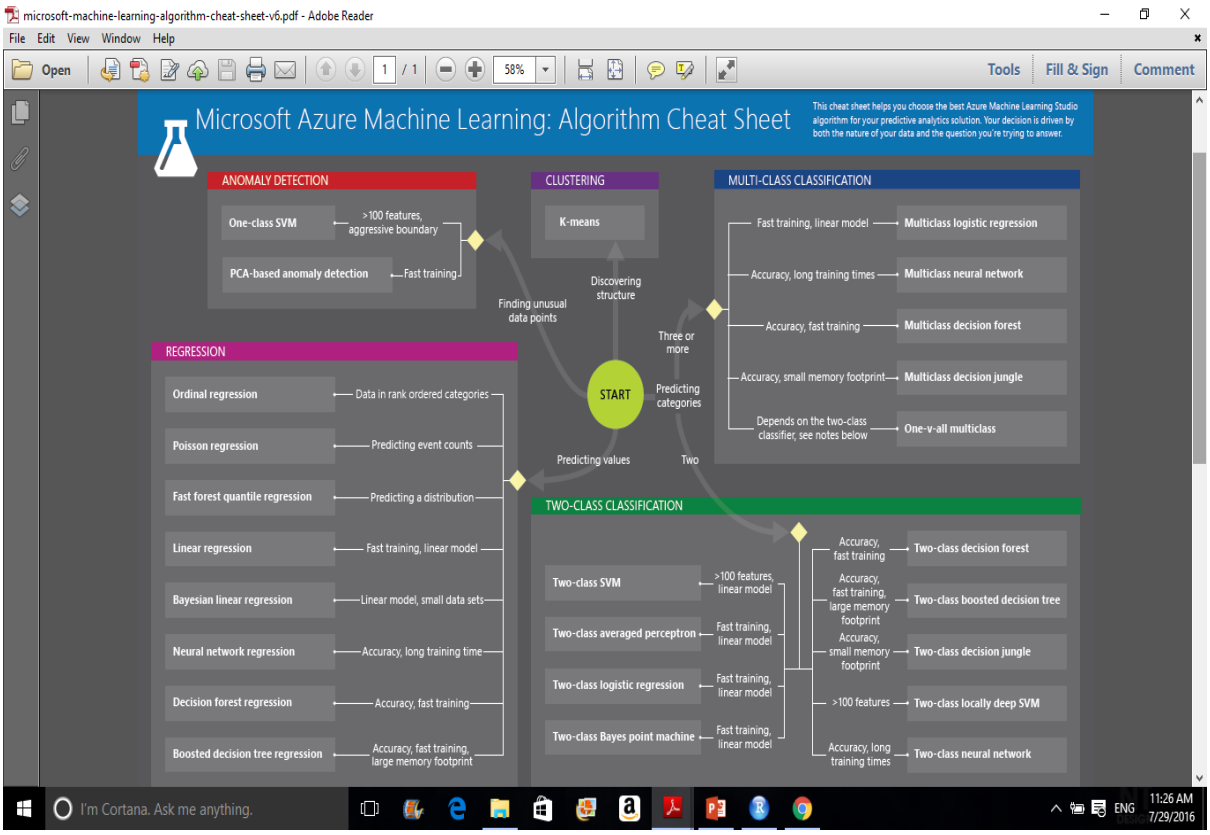
- For betting below statistics can be considered:
    1. Overall matches played by a team
    2. How many matches the team has won at home and away

Winning percentage(wins/total_matches*100)

6. Machine Learning Algorithms:

CLASSIFICATION OF MATCH RESULT

- Our Classification includes classifying whether a match is going to end in a home win or Draw or an away win. For this we made use of the “home goal” column aswell as “away goal” column in building the “Result” column which we have built during feature engineering.
- Based on the information provided in the following link we have applied 4 algorithms for multi class classification and evaluated their performances against each other. https://azure.microsoft.com/en-us/documentation/articles/machine-learning-algorithm-choice/

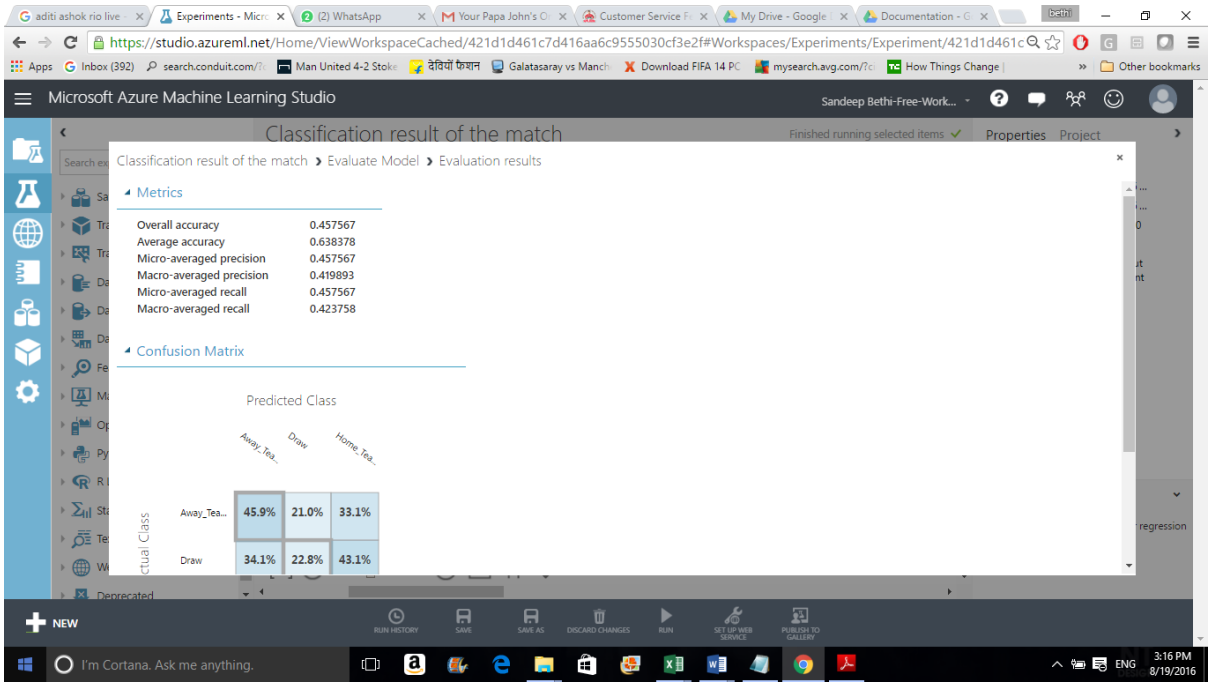


- We built four models for multi class classification and evaluated them against each other based on their performance metrics. We selected one model for classification which has better performance metrics in going ahead and deploying the webservice.
    A) Multi-Class Decision forest
    B) Multi-Class Decision Jungle
    C) Multi-Class Logistic Regression
    D) Multi-Class Neural Network
- The above link contains information regarding How to choose algorithms for Microsoft Azure Machine Learning.

- It has Machine Learning Algorithm Cheat Sheet which tells what algorithm to be used for different type of problems like clustering, classification and regression.
- It also gives the advantages and disadvantages of each algorithm for a set of problem.
- Before building these models we have done **feature selection** by using filter based feature selection module present in Microsoft Azure ML Studio.
  - We selected this module because the features present in our dataset are of both categorical and numeric type.
  - Hence we selected this module for determining the importance of features in building the models.

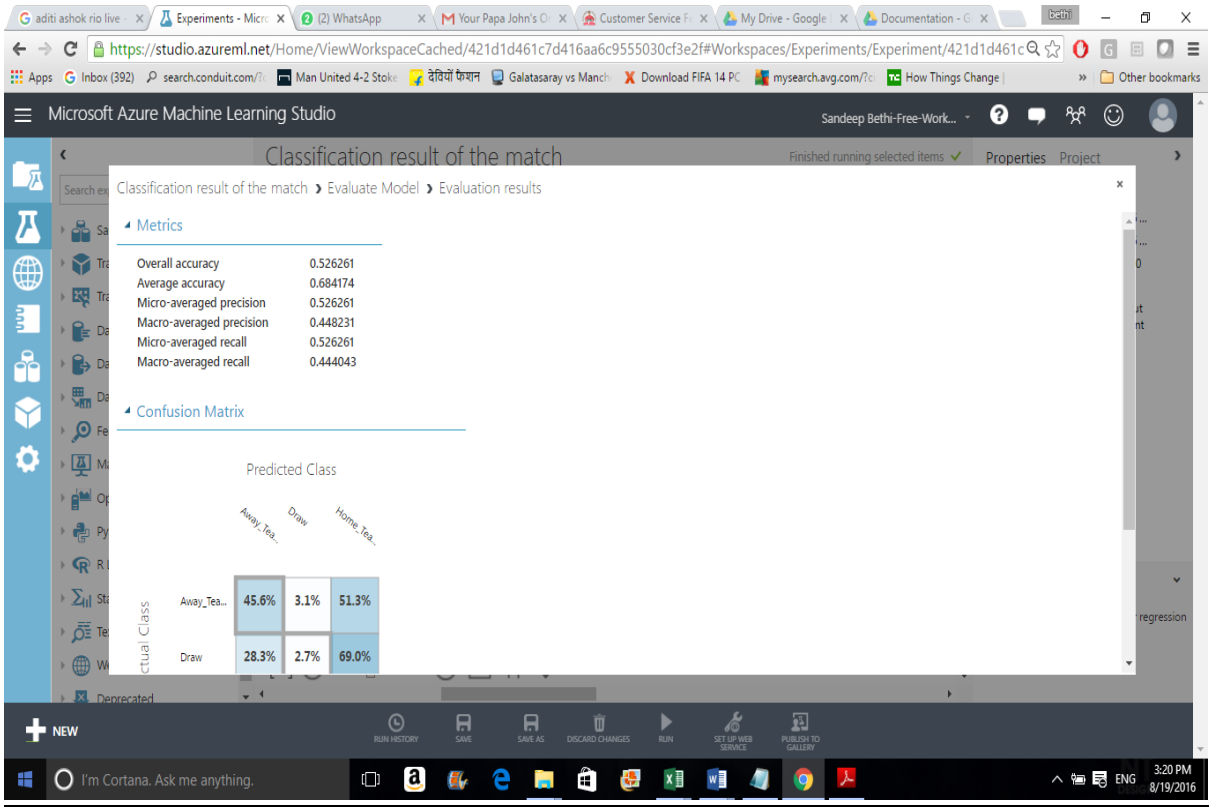
A) Multi-class Decision forest

This particular model gave us the following performance metrics



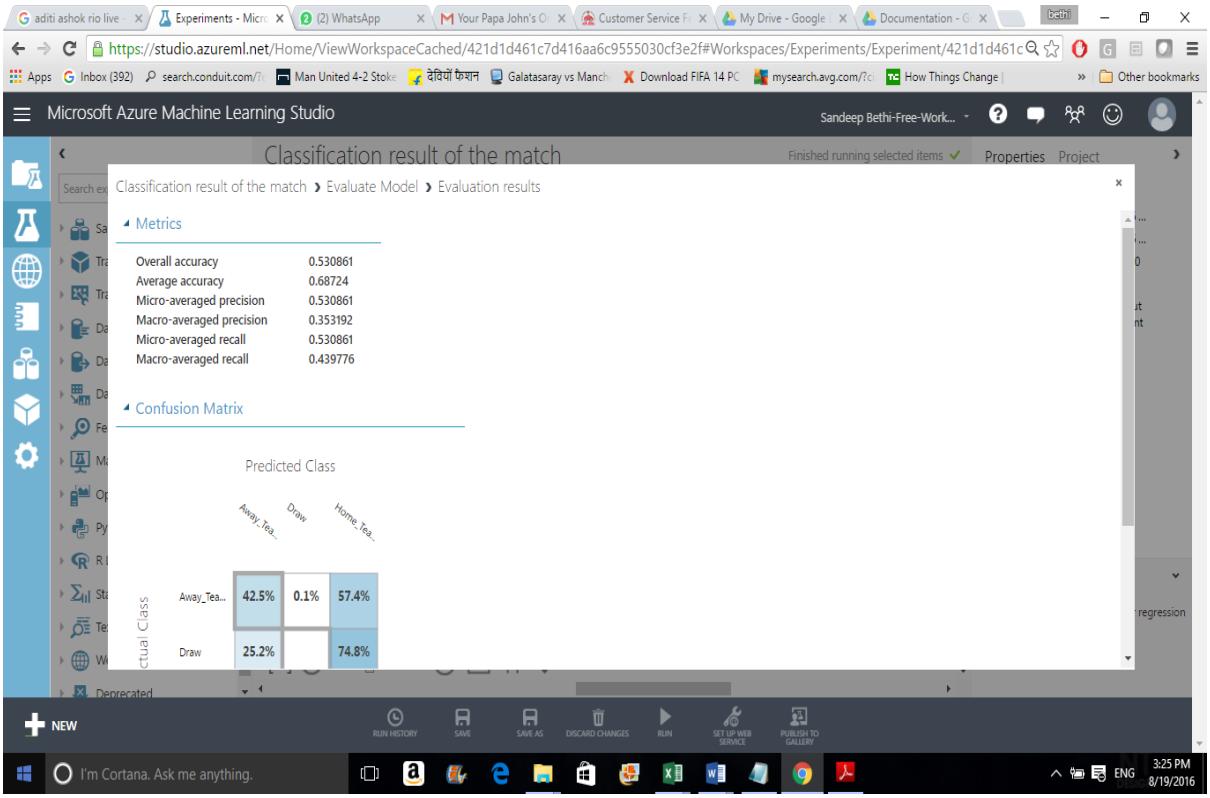
B) Multi-Class Decision Jungle

This particular model gave us the following performance metrics



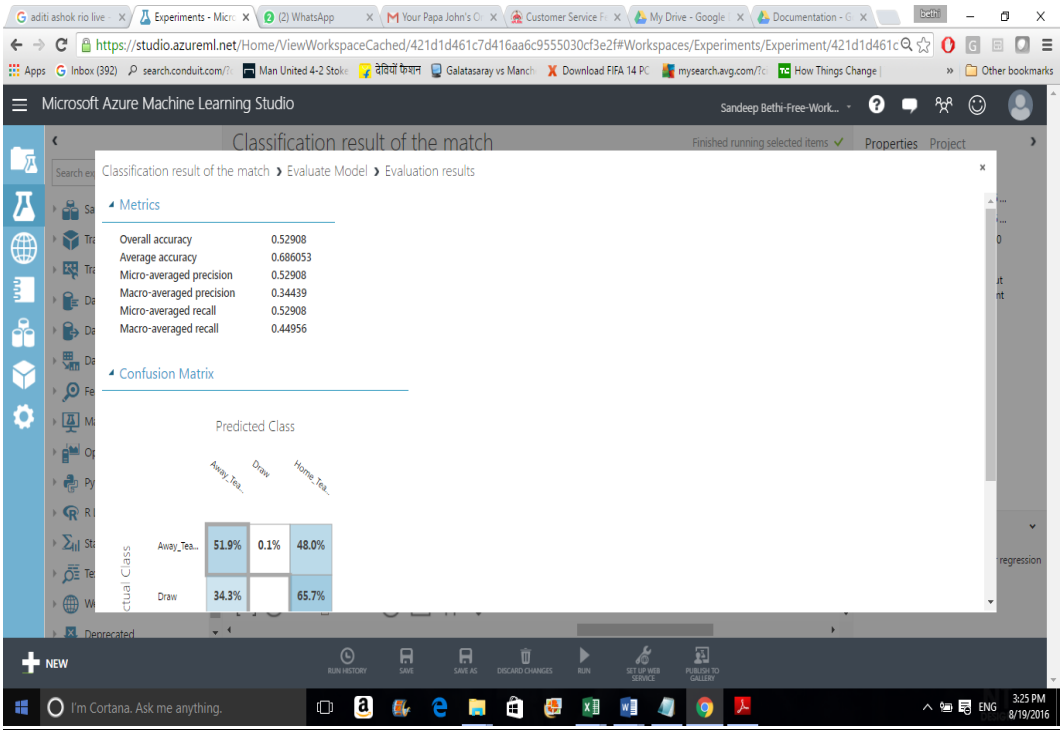
C) Multi-Class Logistic Regression

This particular model gave us the following performance metrics



- **D) Multi-Class Neural Network**

This particular model gave us the following performance metrics



CONCLUSION:

This table gives the comparison of performance metrics of 4 different algorithms used in clustering

	Multi-class Decision Forest	Multi-Class Decision Jungle	Multi Class Logistic Regression	Multi-Class Neural Network
Overall Accuracy	0.45	0.52	0.53	0.52
Average Accuracy	0.63	0.68	0.68	0.68
Micro Averaged Precision	0.45	0.52	0.53	0.52
Micro Averaged Recall	0.45	0.52	0.53	0.52

By looking the above table we can clearly say that the overall performance i.e **Multi-class logistic regression Algorithm** is much better than the rest of the Algorithms.

Even the other performance metrics such as Accuracy, Precision and Recall are better for Multi-Class logistic regression Algorithm.

Hence for the Classification of result of the match we use **Multi-Class logistic regression** Algorithm and deployed the web service using this model.

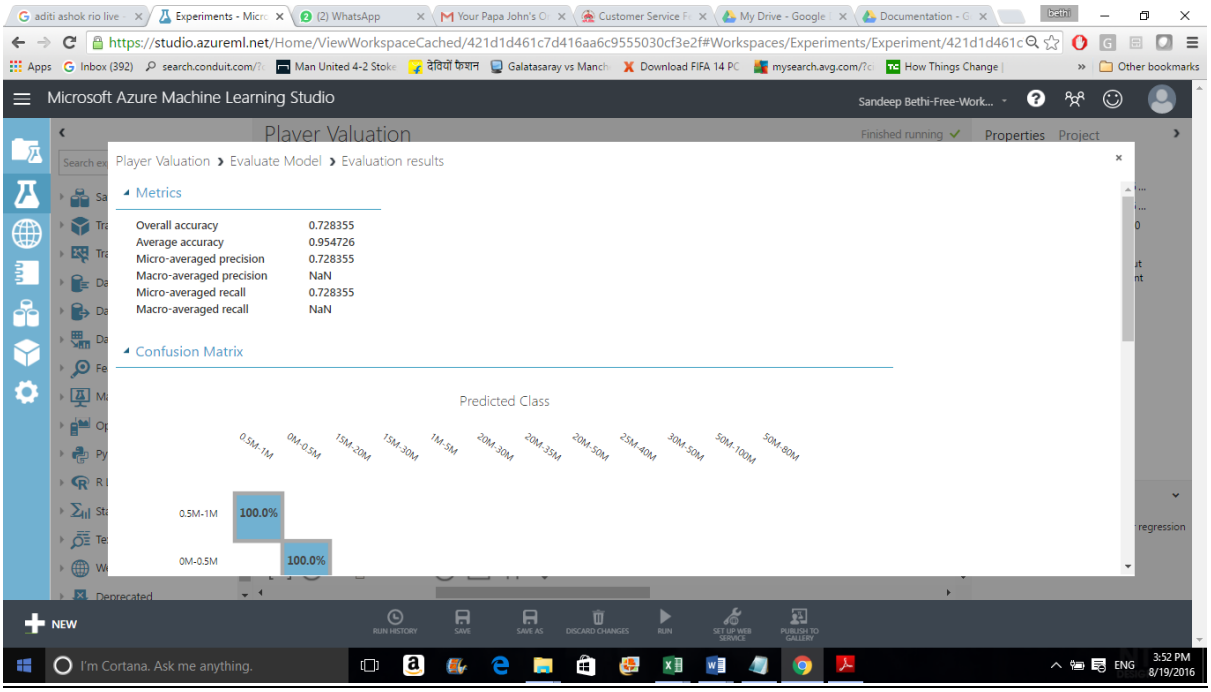
Business Value of this Classification: By classifying the result of the match a lot of general public can place their money on betting and also it helps the managers of different clubs to prepare their teams in a much better fashion.

PLAYER VALUATION:

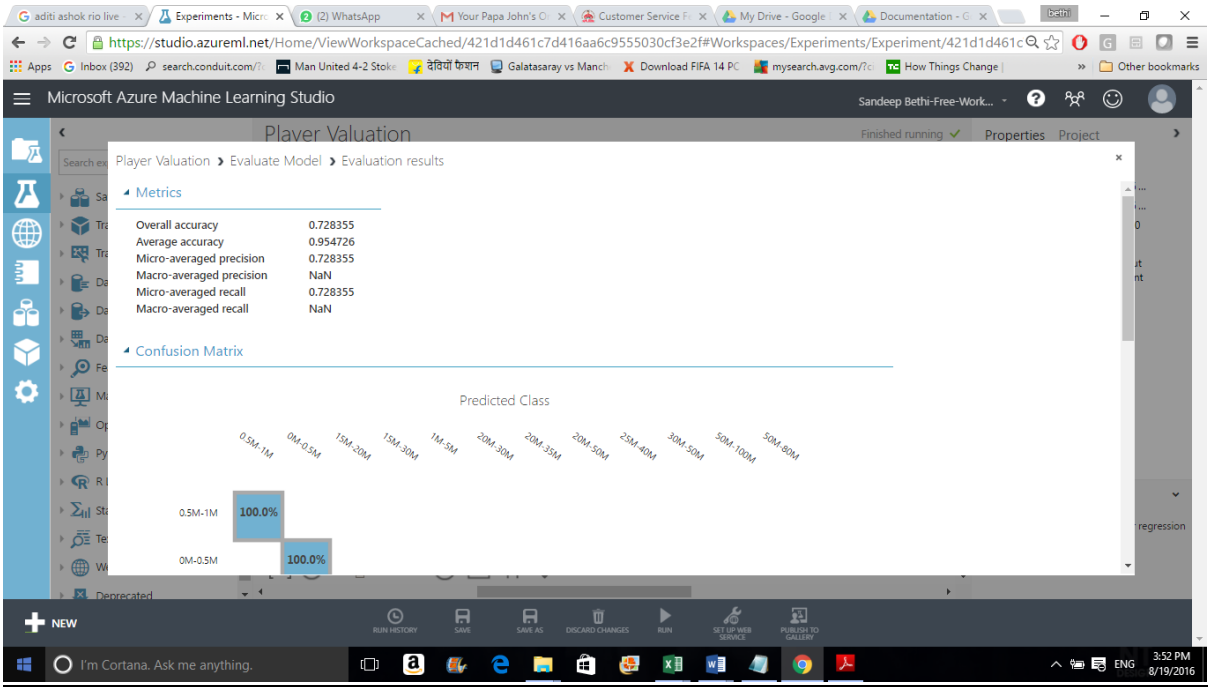
- In this we are going to evaluate and forecast the price of any particular player based on the potential, overall rating and his age.
- We built the following three models for Player Valuation and evaluated them against each other based on their performance metrics. We selected one model for player valuation of price range which has better performance metrics in going ahead and deploying the web service
    A) Multi-Class Decision forest
    B) Multi-Class Decision Jungle
    C) Multi-Class Logistic Regression

- Before building these models we have done **feature selection** by using filter based feature selection module present in Microsoft Azure ML Studio.
- We selected this module because the features present in our dataset are of both categorical and numeric type.
- Hence we selected this module for determining the importance of features in building the models.

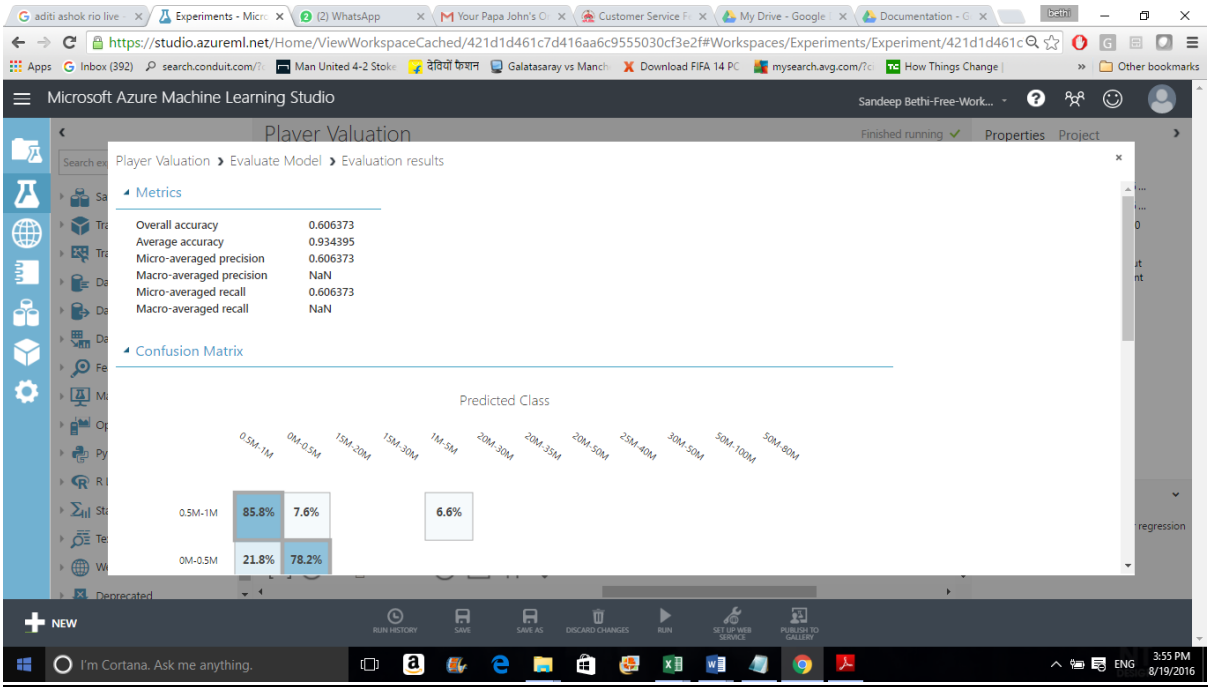
A) <u>MULTI-CLASS DECISION FOREST:</u> This model gave the following Performance metrics



B) <u>MULTI-CLASS DECISION JUNGLE:</u> This model gave the following Performance metrics



C) MULTI-CLASS LOGISTIC REGRESSION: This model gave the following Performance metrics



CONCLUSION

This table gives the comparison of performance metrics of 3 different algorithms used in player valuation

	MULTI-CLASS DECISION FOREST	MULTI-CLASS DECISION JUNGLE	MULTI-CLASS LOGISTIC REGRESSION		
OVERALL ACCURACY	0.85	0.72	0.60		
AVERAGE ACCURACY	0.97	0.95	0.93		
MICRO-AVERAGE PRECISION	0.82	0.72	0.60		
MICRO-AVERAGE RECALL	0.82	0.72	0.60		

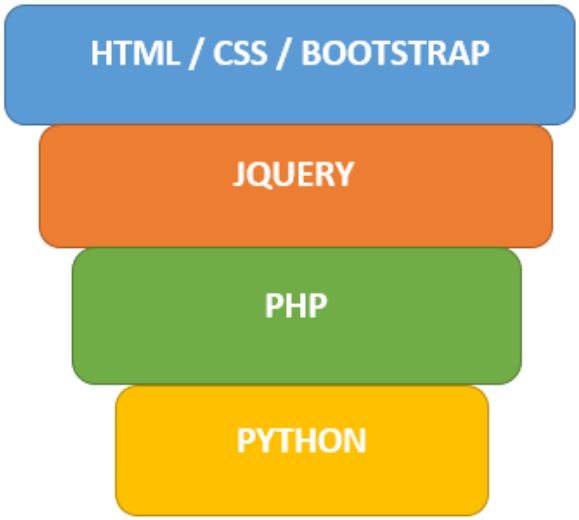
By looking the above table we can clearly say that the overall performance of **MULTI CLASS DECISION Algorithm** is much better than the rest of the Algorithms. Because the accuracy is much better compared to other models.

Hence for the Player valuation of price ranges the MULTI CLASS DECISION Algorithm is better and we choose this model for deploying the web service.

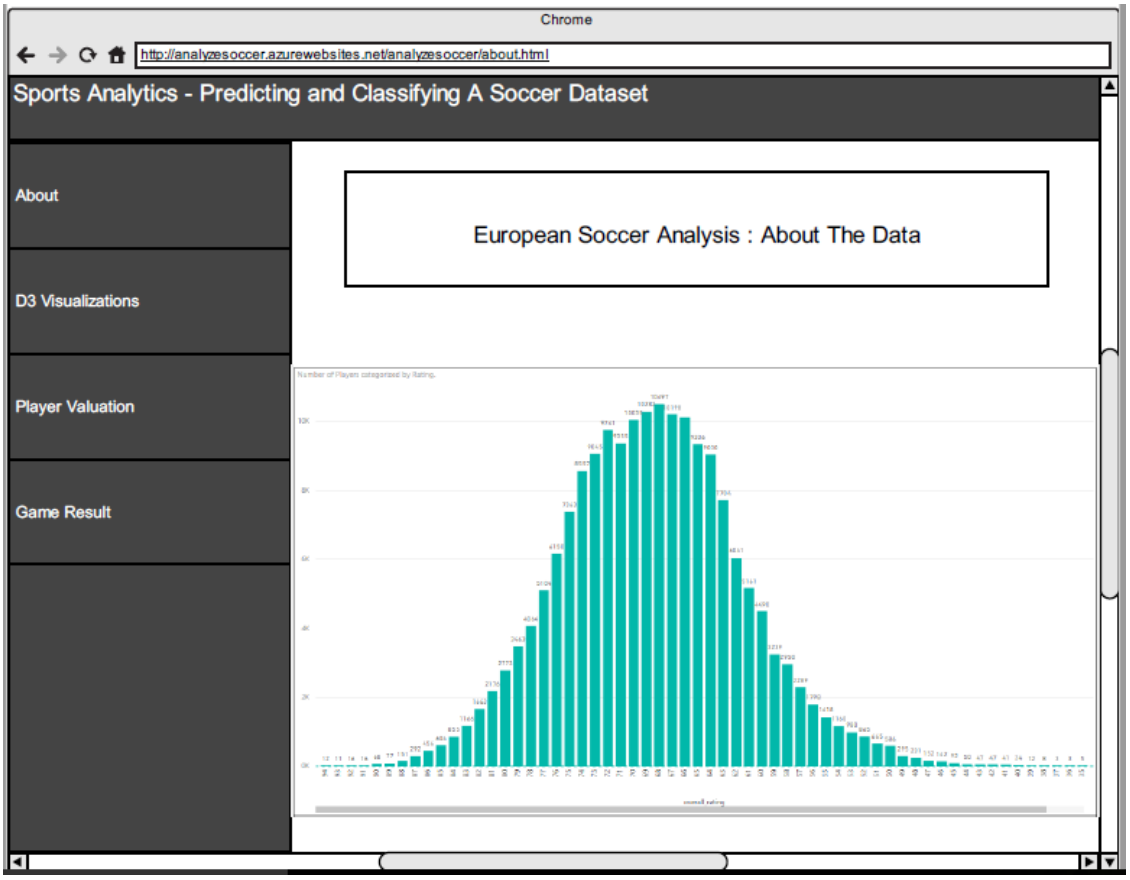
Business Value of Player Valuation: By forecasting the player valuation it helps the management of different clubs in transfer market to buy and sell players. It helps the clubs to do shrewd business in transfers. It helps both the player agents aswell as club management to perform better business

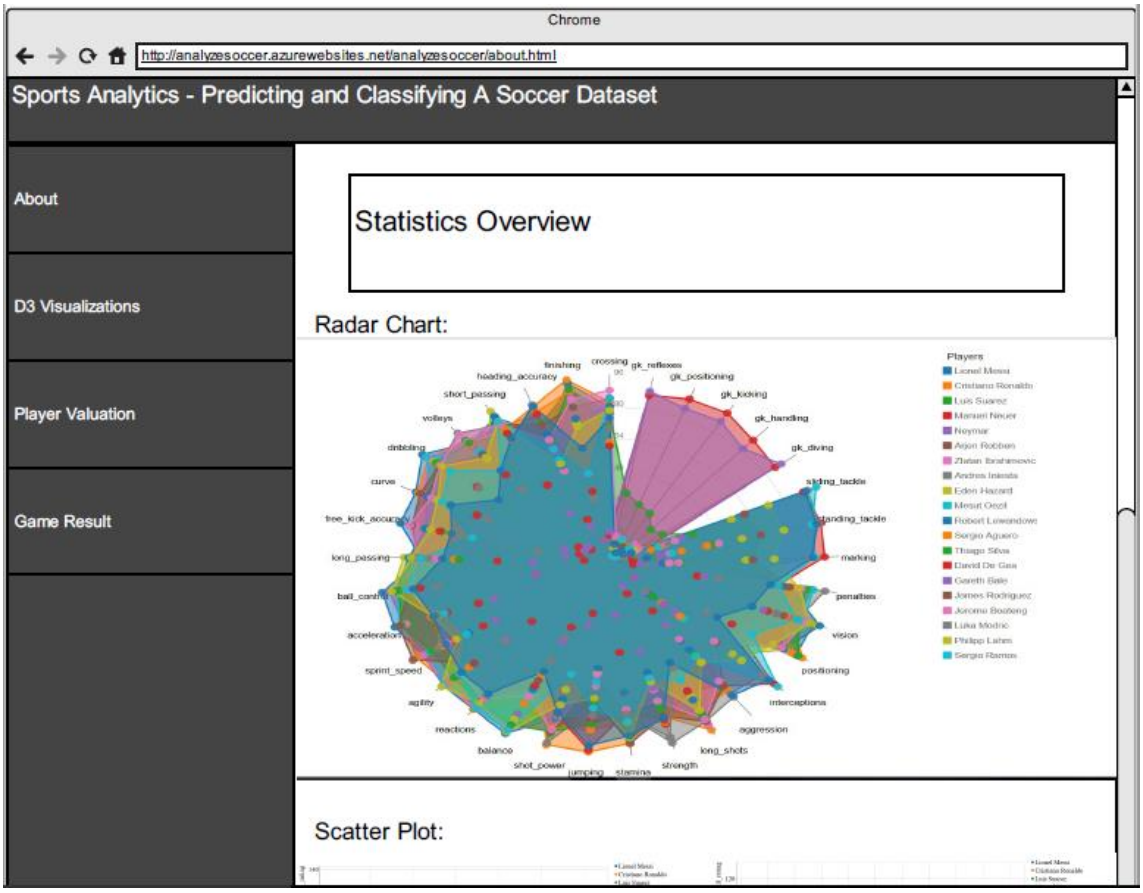
7. Web Site Design

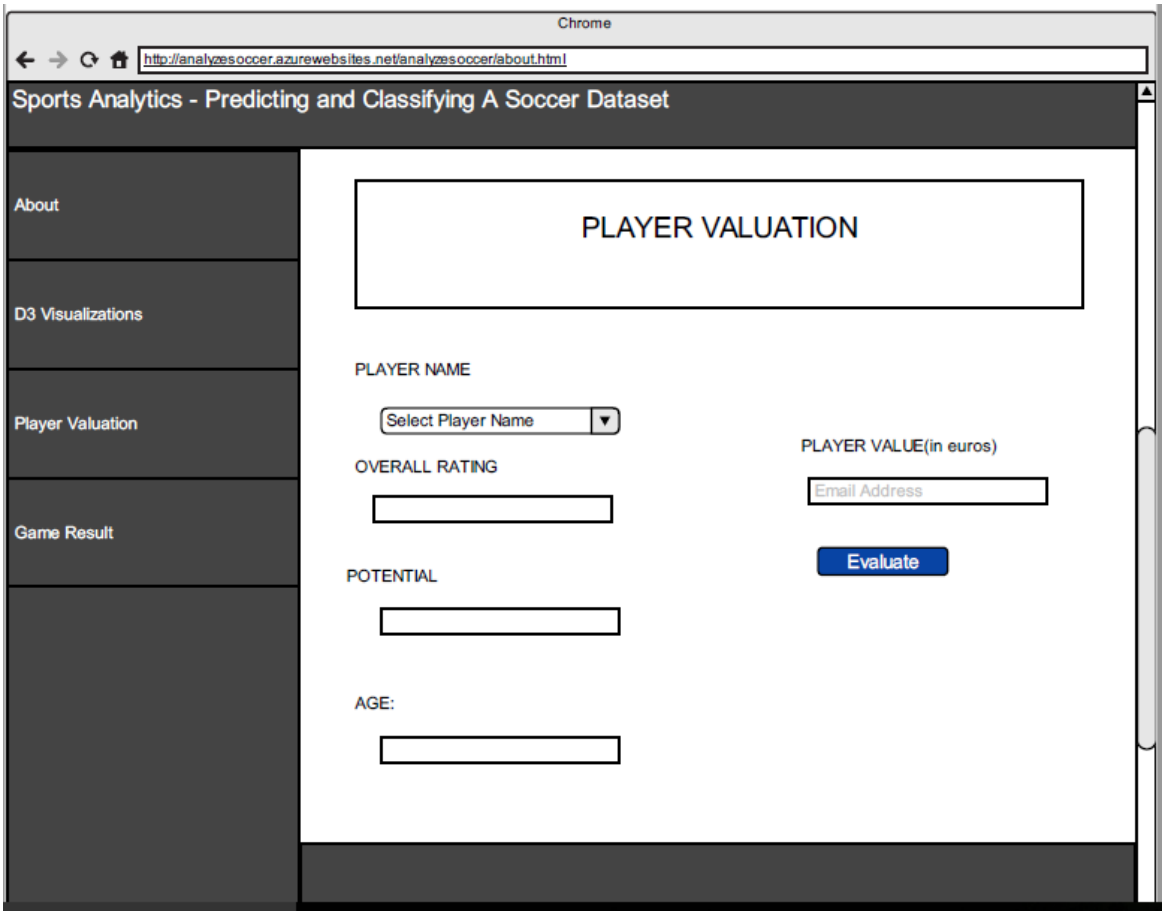
- The website was designed using HTML, CSS, Bootstrap, JQuery, PHP, MS SQL Server and Python.
- The process flow design can be depicted as:

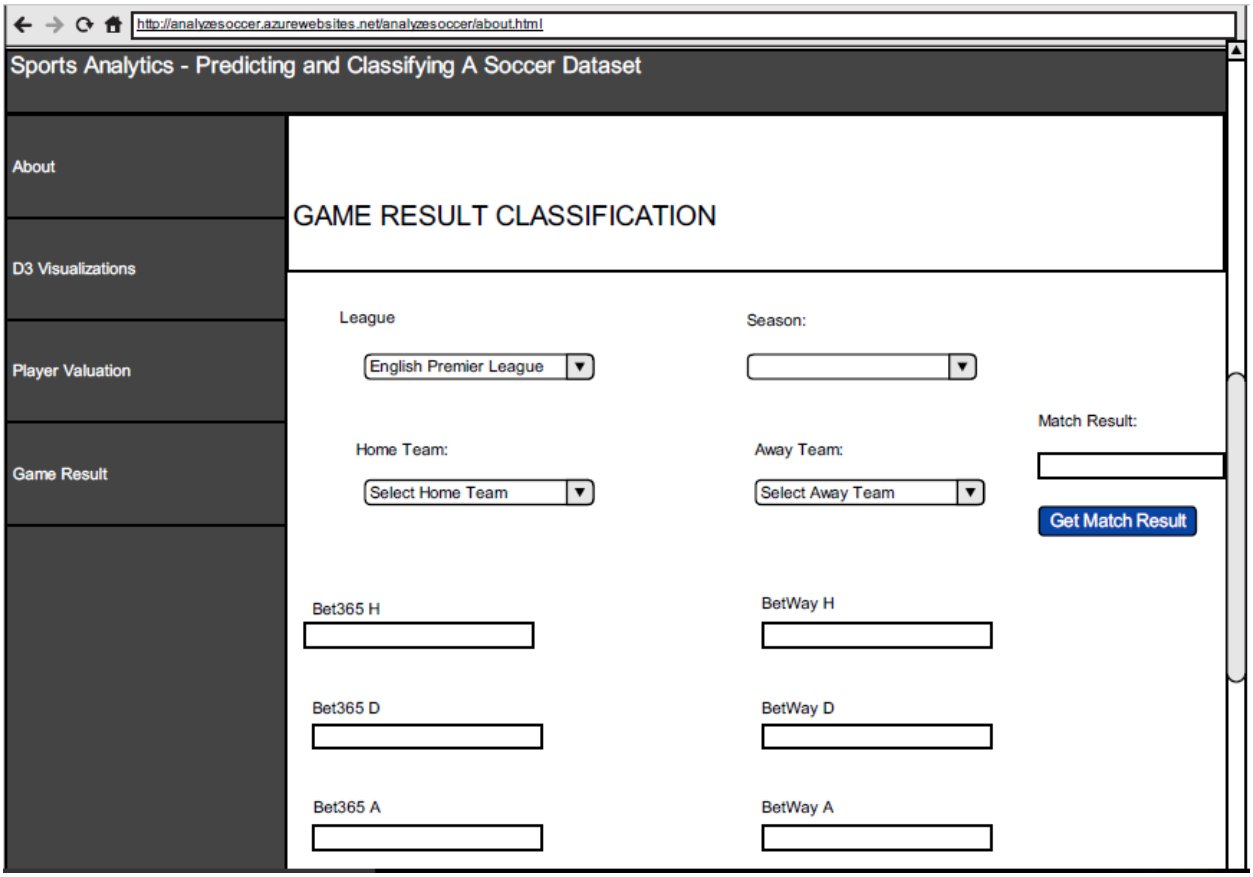


- The Main Page is loaded using HTML, CSS and JavaScript.
- The Visualization on the dashboard are implemented by embedding POWERBI into the webpage.
- Using JQuery we a call a PHP page which in turn executes a python script.
- The python script invokes the webservice which retrieves the data based on the input provided.
- MS SQL Server is used to store the data in the database, and the web application retrieves the data from the database.
- We built the wire frame of our website using 'MOQUPS' and you can find the design of the web pages below:


Chrome
http://analyzesoccer.azurewebsites.net/analyzesoccer/about.html
Sports Analytics - Predicting and Classifying A Soccer Dataset
About
D3 Visualizations
Player Valuation
Game Result
European Soccer Analysis : About The Data


Chrome
http://analyzesoccer.azurewebsites.net/analyzesoccer/about.html
Sports Analytics - Predicting and Classifying A Soccer Dataset
About
D3 Visualizations
Player Valuation
Game Result
Statistics Overview
Radar Chart:
Scatter Plot:


Chrome
http://analyzesoccer.azurewebsites.net/analyzesoccer/about.html
Sports Analytics - Predicting and Classifying A Soccer Dataset
About
D3 Visualizations
Player Valuation
Game Result
PLAYER VALUATION
PLAYER NAME
Select Player Name
OVERALL RATING
POTENTIAL
AGE:
PLAYER VALUE(in euros)
Email Address
Evaluate


http://analyzesoccer.azurewebsites.net/analyzesoccer/about.html
Sports Analytics - Predicting and Classifying A Soccer Dataset
About
D3 Visualizations
Player Valuation
Game Result
GAME RESULT CLASSIFICATION
League
English Premier League
Season:
Home Team:
Select Home Team
Away Team:
Select Away Team
Match Result:
Get Match Result
Bet365 H
Bet365 D
Bet365 A
BetWay H
BetWay D
BetWay A

8. Hosting the Web Application

We chose Microsoft Azure to host the web application as it is a Microsoft product and we were already using ML Studio to perform analytics.
Microsoft DreamSpark is a free subscription within Azure which gives us the capability to host our own website under the azurewebsites.net domain easily.
We are using a FTP Client 'FileZilla' to upload the web app files to the azure cloud and navigating to the website using this URL →
http://analyzesoccer.azurewebsites.net/analyzesoccer/about.html

The configuration steps on Azure are as follows:

a. Login in to portal.azure.com
b. Click on the App Services tab, and click on Add.
c. Enter your App name, 'Create new' Resource Group or 'Use existing'.
d. Select the App and set up the deployment credentials if they are not set. Install 'FileZilla' and use the credentials to login remotely to the server to deploy your app files in the appropriate location.
e. Additional ways to deploy your application can be found on this link: https://azure.microsoft.com/en-us/documentation/articles/web-sites-deploy/

9. References

- https://www.kaggle.com/hugomathien/soccer/kernels
- https://azure.microsoft.com/en-us/documentation/articles/machine-learning-algorithm-choice/