

Design document

Name: Branko Chomic

Date: 13/04/2016

Table of content

Introduction

System Architecture

Parser

Predictions

GUI

Data storage

Updated content

GUI

Predictions

Requirements

References

Introduction

The purpose of this document is to explain how the system will work as well as providing GUI of how the web application should look. The goal of this project is to create web application which would help people in making decisions when placing bets. The web application should include predictions for at least 2 sports and 2 different leagues, it should also provide information for users about the sports such as(latest results, current league table, live scores, upcoming fixtures).

System Architecture

In this part of document complete functionality of the project will be described and how it will work.

Parser

First step in this project is to create parser in Java which would scrap content dynamically from webpages. To scrape the content Jsoup library will be used . First I will need to come up with a way to get statistics for each team and create training file which will consist of all the results from start of the season and a test file which will be used to predict outcome of matches using different machine learning algorithms.

Predictions

Two algorithms were implemented(multi-layer perceptron and linear regression) in Java and tested to see which one performs the best and give the most accurate predictions. WEKA library was used for this part of the project. Training file was

fed into these algorithms so it can learn the pattern, after the algorithm went through many iterations over the training file to build a decent model so it could predict outputs as accurate as possible. The test file was then used to classify each instance. Each algorithm was tested with the same test file to see which algorithm would work best for this problem. The test file consisted of 50 football matches and these are the results that were achieved

1. Multi-layer perceptron gave me the most accurate results with an average of 57% of accuracy. Settings that were used for this test were learning rate 0.8, number of hidden layers were set to 2 and number of iterations was set to 500.
2. Linear regression gave me an accuracy of 48% which was a lot lower. There were no settings used for this algorithm as in linear regression settings cannot be set.

GUI

At this point the GUI was designed. A template was downloaded from the internet and was customized to my needs by deleting/adding content, changing CSS, JavaScript files and adding HTML pages. The goal of the GUI should be easy to use and eye-catching. Design for the user's use of material design was used. The navigation bar had to be clear and easy to use.

Updated content

The next part was to host the website on cloud services such as Amazon services. I chose to use Amazon services for the following reasons:

[1] Advantages:

Low Cost

AWS offers low, pay-as-you-go pricing with no up-front expenses or long-term commitments.

Flexibility and Instant Elasticity

AWS provides a massive ecumenical cloud infrastructure that sanctions you to expeditiously innovate, experiment and iterate. In lieu of waiting weeks or months for hardware, you can instantly deploy incipient applications, instantly scale up as your workload grows, and instantly scale down predicated on demand.

Open and Flexible

AWS is a language and operating system agnostic platform.

Secure

AWS is a secure, durable technology platform with industry-apperceived certifications and audits: PCI DSS Level 1, ISO 27001, FISMA Moderate, FedRAMP, HIPAA, and SOC 1 (formerly referred to as SAS 70 and/or SSAE 16) and SOC 2 audit reports

Next step was to make website content automatically updated every day to provide users fresh content. These were the steps taken to achieve this:

1. Buy domain – Domain was purchased on Namecheap.com
2. Create a bucket in s3 instance on amazon and give same name as domain name
3. Go through few tutorials of changing and setting up bucket to act as a webapp
4. Upload html, css, javascript files to the bucket
5. Redirect the domain name home page to bucket
6. Create ec2 instance which is Linux virtual machine running on cloud
7. Ssh into it using Git bash
8. Export java applications into jar files
9. Upload website and jar files into linux instance
10. Create cronjob to run jar files automatically every day
11. Create python file to execute command that will replace all files updated in the bucket

Data storage

In this project all the content of the website is scrapped using Java and Jsoup library from various sports websites dynamically.

Jsoup library

[2]Jsoup is a Java library which is used for working with real-world HTML. It provides an API for extracting and manipulating data, using the best of DOM, CSS, and jquery-like methods.

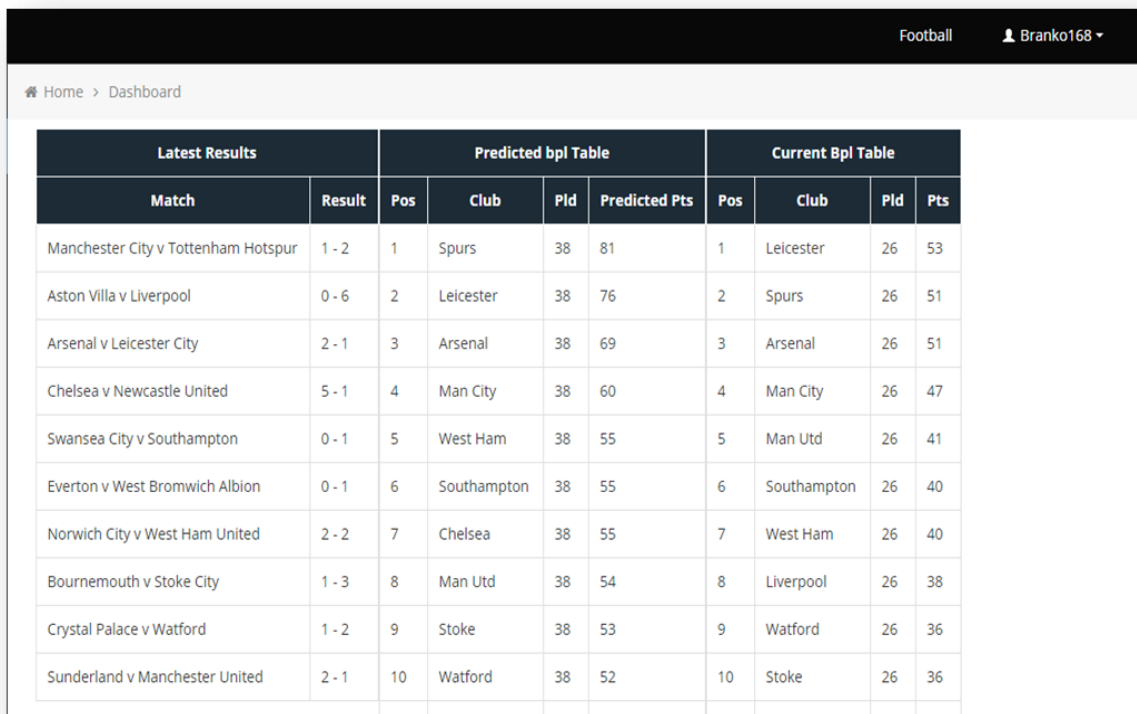
jsoup implements the WHATWG HTML5 specification, there are number of things that can be done using Jsoup such as:

- scrape and parse HTML from a URL, file, or string
- find and extract data, using DOM traversal or CSS selectors
- manipulate the HTML elements, attributes, and text
- clean user-submitted content against a safe white-list, to prevent XSS attacks
- output tidy HTML

All the data scrapped are customized(removing tags, adding classes, replacing names.....) in Java and are written in html pages replacing tags. For example upcoming fixtures that were scrapped will be written in index.html and will replace everything between starting and ending tag(<upcoming> </upcoming>) that way every time new content is parsed from the website it will remove the old content and replace it with new updated content. Data are also stored into text files which are used when predicting outcomes of matches. Html , javascript and css files are stored on cloud using Amazon services in s3 instance bucket. Same files are also stored on Linux virtual machine which was also created and set up using Amazon services, the reason to store them also there is to be able to run jar files which are updating html content and by running python file it will transfer and replace updated html pages from ec2 instance(Linux virtual machine) to s3 instance(bucket).

GUI for my web application

First Design of my web application



The screenshot shows a web application interface for football. At the top right, it says "Football" and "Branko168". Below that is a breadcrumb "Home > Dashboard". The main content is a table with three sections: "Latest Results", "Predicted bpl Table", and "Current Bpl Table".

Latest Results		Predicted bpl Table				Current Bpl Table			
Match	Result	Pos	Club	Pld	Predicted Pts	Pos	Club	Pld	Pts
Manchester City v Tottenham Hotspur	1 - 2	1	Spurs	38	81	1	Leicester	26	53
Aston Villa v Liverpool	0 - 6	2	Leicester	38	76	2	Spurs	26	51
Arsenal v Leicester City	2 - 1	3	Arsenal	38	69	3	Arsenal	26	51
Chelsea v Newcastle United	5 - 1	4	Man City	38	60	4	Man City	26	47
Swansea City v Southampton	0 - 1	5	West Ham	38	55	5	Man Utd	26	41
Everton v West Bromwich Albion	0 - 1	6	Southampton	38	55	6	Southampton	26	40
Norwich City v West Ham United	2 - 2	7	Chelsea	38	55	7	West Ham	26	40
Bournemouth v Stoke City	1 - 3	8	Man Utd	38	54	8	Liverpool	26	38
Crystal Palace v Watford	1 - 2	9	Stoke	38	53	9	Watford	26	36
Sunderland v Manchester United	2 - 1	10	Watford	38	52	10	Stoke	26	36

Predicted Results		Upcoming Fixtures		
Match	Predicted Result	Saturday 27 February 2016		
West Ham v Sunderland	2 - 0	12:45	West Ham v Sunderland	Boleyn Ground
Leicester v Norwich	2 - 0	15:00	Leicester v Norwich	King Power Stadium
Southampton v Chelsea	2 - 0	15:00	Southampton v Chelsea	St. Mary's Stadium
Stoke v Aston Villa	2 - 0	15:00	Stoke v Aston Villa	Britannia Stadium
Watford v Bournemouth	0 - 0	15:00	Watford v Bournemouth	Vicarage Road
West Brom v Crystal Palace	1 - 0	17:30	West Brom v Crystal Palace	The Hawthorns
Liverpool v Everton	0 - 2	Sunday 28 February 2016		
Newcastle v Man City	0 - 2	POSTP	Liverpool v Everton	Anfield
Man Utd v Arsenal	0 - 2	POSTP	Newcastle v Man City	St. James' Park
Spurs v Swansea	1 - 0	14:05	Man Utd v Arsenal	Old Trafford
		14:05	Spurs v Swansea	White Hart Lane

This was my first design for my project as you can see it is very simple and is not very eye catching. There was no background it didn't look like a webpage that I would use if I would have to be user. Predicted results table was also very simple table with little information which I didn't like and wanted to change it. Also this design didn't contain any buttons to view stats, to change league or to view live scores.

HOME FOOTBALL BASKETBALL

Premier League Liga BBVA View live scores

Latest Results	
Event	Score
Manchester City v Manchester United	0 - 1
Newcastle United v Sunderland	1 - 1
Southampton v Liverpool	3 - 2
Tottenham Hotspur v AFC Bournemouth	3 - 0
Chelsea v West Ham United	2 - 2

Current Bpl Table		
Club	Pld	Pts
Leicester	31	66
Spurs	31	61
Arsenal	30	55
Man City	30	51
West Ham	30	50
Man Utd	30	50
Southampton	31	47
Stoke	31	46
Liverpool	29	44

Upcoming fixtures		
Date	Time	Event
02 Apr	12:45	Aston Villa v Chelsea
02 Apr	15:00	West Ham United v Crystal Palace
02 Apr	15:00	Bournemouth v Manchester City
02 Apr	15:00	Norwich City v Newcastle United
02 Apr	15:00	Stoke City v Swansea City

Predicted bpl Table		
Club	Pld	Pts
Leicester	38	80
Spurs	38	72
Arsenal	38	76
Man City	38	58
West Ham	38	62
Man Utd	38	60
Southampton	38	54
Stoke	38	62
Liverpool	38	61

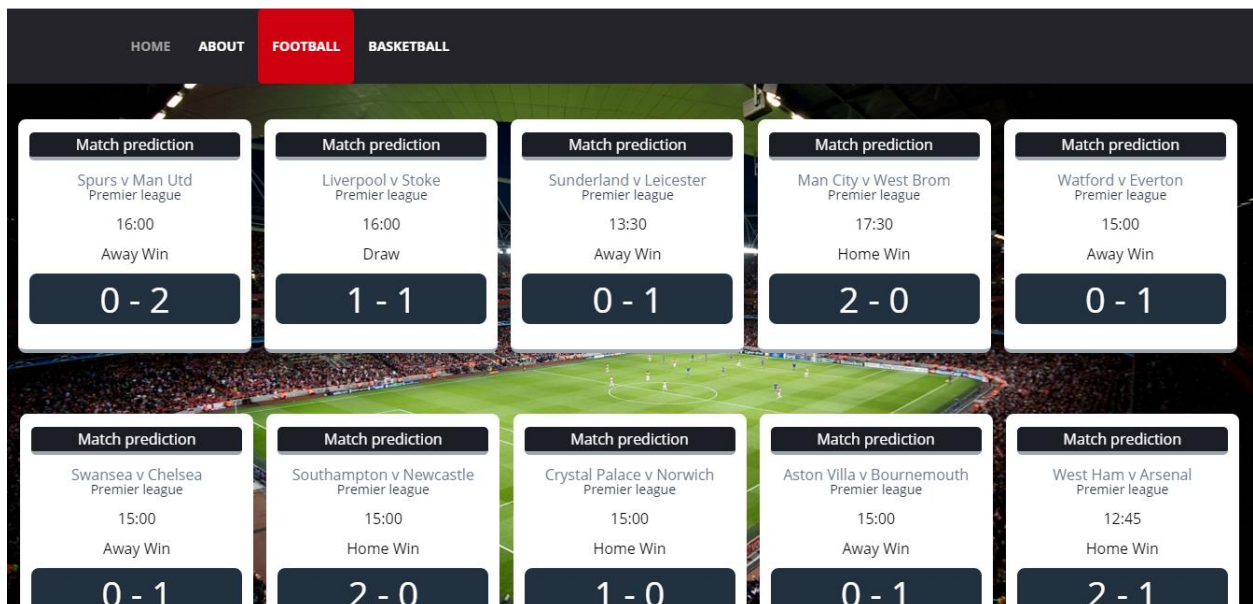
To change design I download template from internet which was modified to my needs(deleting content, adding buttons, changing colors, modifies javascript, adding content). As you can see design has improved a lot since the first one tables are more clear, more readable the website is eye-catching . It provides nice menu where you can choose sport, button to switch between leagues and overall it looks and feels better than the previous design.

Top Menu bar

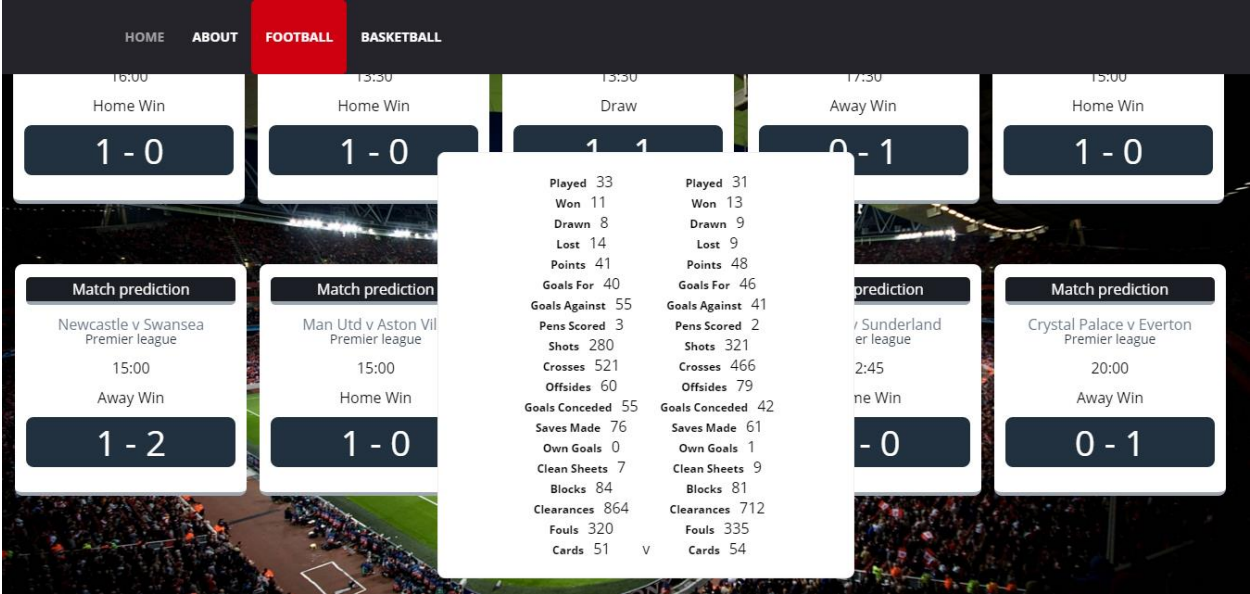
- Home button that takes user to home page
- Football button take user to football page
- Basketball button displays basketball content

Buttons

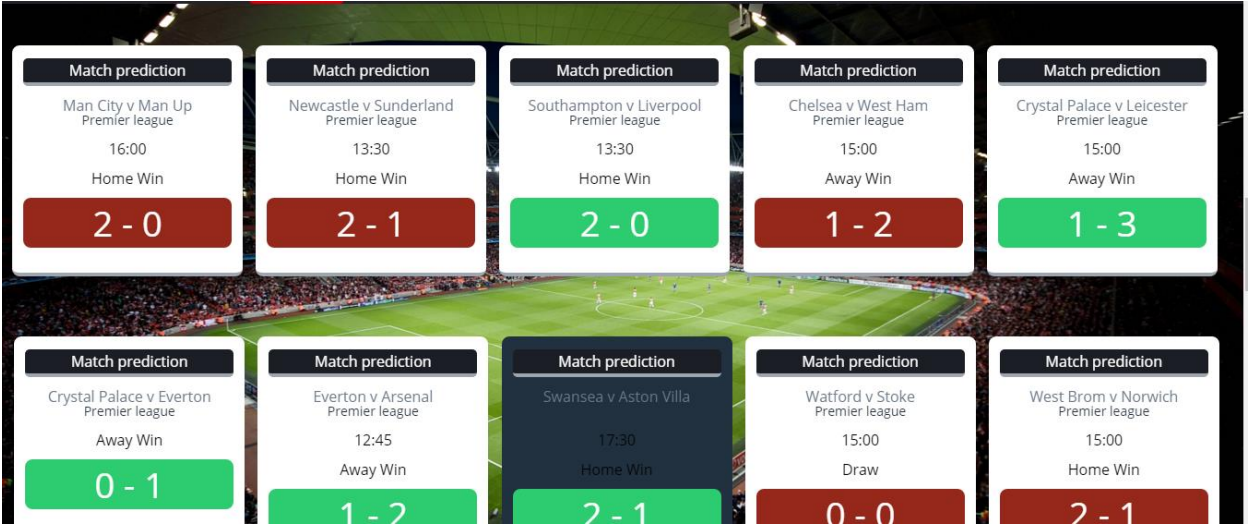
- Liga BBVA button display content for Spanish league
- Premier league button displays content for premier league
- View live score button allow user to see any live scores available



Predicting tables has been completely redesigned because it didn't look modern and also it didn't provide sufficient information. I created this small square divs and added more information like competition, time, and 1x2 predictions.



Also I added click function in jquery when any prediction div is clicked statistics home and away team are displayed.



Predictions were added from last week and java method was written which checks what results were predicted correctly so i added different colors to indicate if prediction was correct(green) incorrect(red).

Predictions

Training File:

First training file that I have created included only 50 matches results and the statistics consisted of home team last 5 matches setting 1 for win 0 for draw and -1 for lost and away team last matches. The problem with this training file was that it didn't provide enough results and enough stats for each team to be able to build decent model. I tested it with 40 football matches giving me an average of 41% accuracy which wasn't good enough.

Training File:

```
@attribute Apoints2 numeric
@attribute Apoints3 numeric
@attribute class1 {0,1,2,3,4,5,6,7,8,9}
@attribute class2 {0,1,2,3,4,5,6,7,8,9}
@data
20,30,3,7,20,16,22,57,2,228,523,36,57,97,3,5,86,777,354,60,2,30,16,10,4,58,53,24,5,371,438,61,24,71,3,10,109,848,367,58,0,2
13,30,10,8,12,38,38,47,3,262,473,52,47,69,0,7,83,786,294,48,16,30,8,9,13,33,30,40,5,256,459,71,40,92,4,7,100,849,310,46,3,2
9,30,12,7,11,43,32,36,5,251,447,79,36,100,0,10,114,624,326,43,7,30,12,8,10,44,38,30,2,303,588,75,30,67,0,12,103,881,331,53,1,0
18,30,6,7,17,25,31,54,1,247,483,74,54,82,2,4,83,991,344,56,4,29,15,6,8,51,52,31,2,340,521,44,31,71,1,13,47,722,307,48,0,0
11,29,10,9,10,39,30,36,1,210,514,54,36,88,2,10,137,786,293,58,6,29,13,8,8,47,37,27,3,227,503,54,27,72,1,13,63,591,361,52,1,0
15,29,9,6,14,33,32,39,4,273,572,67,39,87,3,6,142,757,368,49,8,28,12,8,8,44,43,37,2,301,442,77,37,55,1,9,70,660,311,49,1,2
14,29,10,7,12,37,29,30,5,232,402,36,30,92,1,10,80,750,339,56,1,29,17,9,3,60,52,31,8,300,476,40,31,77,0,10,123,772,313,43,0,1
4,29,15,6,8,51,52,31,2,340,521,44,31,71,1,13,47,722,307,48,20,30,3,7,20,16,22,57,2,228,523,36,57,97,3,5,86,777,354,60,4,0
7,30,12,8,10,44,38,30,2,303,588,75,30,67,0,12,103,881,331,53,17,29,6,7,16,25,35,54,1,237,356,57,54,117,1,3,107,930,306,52,1,1
12,28,9,11,8,38,51,39,3,273,453,43,39,90,0,8,105,712,235,39,5,29,13,10,6,49,45,33,1,306,465,67,33,96,2,10,133,811,284,44,2,3
10,29,10,10,9,40,43,39,1,288,424,61,39,89,2,8,87,614,319,47,9,30,12,7,11,43,32,36,5,251,447,79,36,100,0,10,114,624,326,43,1,1
19,28,6,6,16,24,28,53,1,211,460,44,53,92,2,5,119,622,302,49,13,30,10,8,12,38,38,47,3,262,473,52,47,69,0,7,83,786,294,48,1,3
16,30,8,9,13,33,30,40,5,256,459,71,40,92,4,7,100,849,310,46,18,30,6,7,17,25,31,54,1,247,483,74,54,82,2,4,83,991,344,56,1,0
2,30,16,10,4,58,53,24,5,371,438,61,24,71,3,10,109,848,367,58,3,29,15,7,7,52,46,30,1,322,432,66,30,99,3,12,103,660,266,36,2,2
8,28,12,8,8,44,43,37,2,301,442,77,37,55,1,9,70,660,311,49,4,29,15,6,8,51,52,31,2,340,521,44,31,71,1,13,47,722,307,48,3,0
6,29,13,8,8,47,37,27,3,227,503,54,27,72,1,13,63,591,361,52,14,29,10,7,12,37,29,30,5,232,402,36,30,92,1,10,80,750,339,56,1,0
3,29,15,7,7,52,46,30,1,322,432,66,30,99,3,12,103,660,266,36,16,30,8,9,13,33,30,40,5,256,459,71,40,92,4,7,100,849,310,46,1,2
9,30,12,7,11,43,32,36,5,251,447,79,36,100,0,10,114,624,326,43,19,28,6,6,16,24,28,53,1,211,460,44,53,92,2,5,119,622,302,49,1,0
5,29,13,10,6,49,45,33,1,306,465,67,33,96,2,10,133,811,284,44,2,30,16,10,4,58,53,24,5,371,438,61,24,71,3,10,109,848,367,58,1,0
20,30,3,7,20,16,22,57,2,228,523,36,57,97,3,5,86,777,354,60,12,28,9,11,8,38,51,39,3,273,453,43,39,90,0,8,105,712,235,39,1,3
13,30,10,8,12,38,38,47,3,262,473,52,47,69,0,7,83,786,294,48,7,30,12,8,10,44,38,30,2,303,588,75,30,67,0,12,103,881,331,53,2,0
17,29,6,7,16,25,35,54,1,237,356,57,54,117,1,3,107,930,306,52,15,29,9,6,14,33,32,39,4,273,572,67,39,87,3,6,142,757,368,49,2,2
1,29,17,9,3,60,52,31,8,300,476,40,31,77,0,10,123,772,313,43,11,29,10,9,10,39,30,36,1,210,514,54,36,88,2,10,137,786,293,58,2,2
18,30,6,7,17,25,31,54,1,247,483,74,54,82,2,4,83,991,344,56,10,29,10,10,9,40,43,39,1,288,424,61,39,89,2,8,87,614,319,47,1,2
```

This is my second training file and the current I am using to feed into neural network. To get more accurate results I had to think about statistics that are most

decisive in matches such as (league position, goals scored, goals conceded, shots per game....) which are all very important when deciding the outcome of the game.

Test File:

```
@attribute Apasses numeric
@attribute Attackles numeric
@attribute Apoints numeric
@attribute Apoints2 numeric
@attribute Apoints3 numeric
@attribute class1 {0,1,2,3,4,5,6,7,8,9}
@attribute class2 {0,1,2,3,4,5,6,7,8,9}
@data
3,29,15,7,7,52,46,30,1,322,432,66,30,99,3,12,103,660,266,36,11,29,10,9,10,39,30,36,1,210,514,54,36,88,2,10,137,786,293,58,?,?
8,28,12,8,8,44,43,37,2,301,442,77,37,55,1,9,70,660,311,49,10,29,10,10,9,40,43,39,1,288,424,61,39,89,2,8,87,614,319,47,?,?
8,28,12,8,8,44,43,37,2,301,442,77,37,55,1,9,70,660,311,49,12,28,9,11,8,38,51,39,3,273,453,43,39,90,0,8,105,712,235,39,?,?
6,29,13,8,8,47,37,27,3,227,503,54,27,72,1,13,63,591,361,52,15,29,9,6,14,33,32,39,4,273,572,67,39,87,3,6,142,757,368,49,?,?
19,29,6,6,17,24,28,54,1,219,475,45,54,92,2,5,125,650,311,49,4,29,15,6,8,51,52,31,2,340,521,44,31,71,1,13,47,722,307,48,?,?
17,29,6,7,16,25,35,54,1,237,356,57,54,117,1,3,107,930,306,52,12,28,9,11,8,38,51,39,3,273,453,43,39,90,0,8,105,712,235,39,?,?
5,29,13,10,6,49,45,33,1,306,465,67,33,96,2,10,133,811,284,44,14,29,10,7,12,37,29,30,5,232,402,36,30,92,1,10,80,750,339,56,?,?
12,28,9,11,8,38,51,39,3,273,453,43,39,90,0,8,105,712,235,39,3,29,15,7,7,52,46,30,1,322,432,66,30,99,3,12,103,660,266,36,?,?
10,29,10,10,9,40,43,39,1,288,424,61,39,89,2,8,87,614,319,47,5,29,13,10,6,49,45,33,1,306,465,67,33,96,2,10,133,811,284,44,?,?
15,29,9,6,14,33,32,39,4,273,572,67,39,87,3,6,142,757,368,49,1,30,18,9,3,63,53,31,8,304,496,40,31,79,0,11,128,790,317,43,?,?
14,29,10,7,12,37,29,30,5,232,402,36,30,92,1,10,80,750,339,56,9,30,12,7,11,43,32,36,5,251,447,79,36,100,0,10,114,624,326,43,?,?
11,29,10,9,10,39,30,36,1,210,514,54,36,88,2,10,137,786,293,58,18,30,6,7,17,25,31,54,1,247,483,74,54,82,2,4,83,991,344,56,?,?
16,30,8,9,13,33,30,40,5,256,459,71,40,92,4,7,100,849,310,46,20,30,3,7,20,16,22,57,2,229,522,36,57,97,3,5,86,777,354,60,?,?
19,29,6,6,17,24,28,54,1,219,475,45,54,92,2,5,125,650,311,49,17,29,6,7,16,25,35,54,1,237,356,57,54,117,1,3,107,930,306,52,?,?
7,30,12,8,10,44,38,30,2,303,588,75,30,67,0,12,103,881,331,53,8,28,12,8,8,44,43,37,2,301,442,77,37,55,1,9,70,660,311,49,?,?
4,29,15,6,8,51,52,31,2,340,521,44,31,71,1,13,47,722,307,48,6,29,13,8,8,47,37,27,3,227,503,54,27,72,1,13,63,591,361,52,?,?
2,30,16,10,4,58,53,24,5,371,438,61,24,71,3,10,109,847,367,58,13,30,10,8,12,38,38,47,3,262,473,52,47,69,0,7,83,786,294,48,?,?
20,30,3,7,20,16,22,57,2,229,522,36,57,97,3,5,86,777,354,60,10,29,10,10,9,40,43,39,1,288,424,61,39,89,2,8,87,614,319,47,?,?
3,29,15,7,7,52,46,30,1,322,432,66,30,99,3,12,103,660,266,36,14,29,10,7,12,37,29,30,5,232,402,36,30,92,1,10,80,750,339,56,?,?
13,30,10,8,12,38,38,47,3,262,473,52,47,69,0,7,83,786,294,48,4,29,15,6,8,51,52,31,2,340,521,44,31,71,1,13,47,722,307,48,?,?
18,30,6,7,17,25,31,54,1,247,483,74,54,82,2,4,83,991,344,56,19,29,6,6,17,24,28,54,1,219,475,45,54,92,2,5,125,650,311,49,?,?
```

This is a test file used in this project it consists of upcoming fixtures till the end of the season. This testing data sets gets updated every day it is important as statistics for each team changes every time team is playing a match so for example if team is at first position in the table this week and in few weeks same team drops to like 8th position it will be very important to keep the stats updated to get more accurate results for every match.

Requirements

This are the following requirements needed to finish this project:

- Jsoup library
- IDE for writing code such as eclipse
- Install Java and all the components needed
- WEKA library
- Amazon services
- Html template

References

[1] Name: Amazon services : Amazon, accessed 18 October 2015

<https://aws.amazon.com/about-aws/>

[1] Name: Jsoup: Amazon, accessed 20 October 2015

<http://jsoup.org/>