Tailleur Clément
GZYG1Q

Projet laboratory

Advanced data analysis

ESME sudria
Lille - Paris - Lyon

Summary:

Problematic: How can we determine a specific group of player with data?

I] Introduction

1) Datamining general and sport history
2) Rapidminer in general

II] My Data

1) Presentation
2) Goal
3) Tools (operator…)

II] Detection

1) Goalkeeper
2) Defender
3) Midfielder
4) Forward

I] Introduction

1) Data Mining

Data Mining is the process of extracting information knowledge from large volume of raw data.

Data mining finds these patterns and relationships using data analysis tools and techniques to build models. There are two main kinds of models in data mining.

- The first one is predictive models, which use data with known results to develop a model that can be used to explicitly predict values.
- The second is descriptive models, which describe patterns in existing data.

All the models are abstract representations of reality, and can be guides to understanding business and suggest actions.

Today, there is another way to use this process. In recent years, Sport Data Mining has experienced rapid growth. Tools and techniques began to be developed to better measure both player and team performance. Then more and more franchises are using data mining in order to reduce the risk of error when they buy a player.

Professional sports organizations can be multi-million dollar enterprises with millions of dollars spent on a single decision. With this amount of capital at stake, just one bad or misguided decision has the potential of setting an organization back by several years. With such a large array of risk and a critical need to make good decisions, the sports industry is an attractive environment for data mining applications.

2) RapidMiner

In order to solve this problem, we use a software called RapidMiner. Developed in 2001 by Ralph Klinkenberg, Ingo Mierswa and Simon Fischer at Dortmund. Starting in 2006 under the YALE name, the name changed to RapidMiner in 2007.

II] My Data

1) Presentation

I used data from English football premier league describing statistics of each player during the season 2011/2012 for my project.

How does it present?

It is an excel file including 10370 lines (each player of the league multiplied by each match he played) and 211 column (representing different statistics for each player and each match).

	A	B	C	D	N	O	P	Q	R	S	T	U
1	Date	Player ID	Player Surname	Player Forename	Substitute On	Substitute Off	Goals	First Goal	Winning Goal	Shots On Target inc goals	Shots Off Target inc woodwork	Blocked Shots
9472	2011-12-31	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9473	2012-03-17	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9474	2011-11-27	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9475	2012-05-06	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9476	2011-11-19	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9477	2011-10-22	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9478	2011-09-24	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9479	2011-10-29	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9480	2011-12-03	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9481	2012-04-01	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9482	2012-01-21	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9483	2012-02-11	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9484	2012-03-11	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9485	2012-02-04	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9486	2012-01-15	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9487	2012-03-03	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9488	2012-01-31	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9489	2012-04-28	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9490	2012-04-11	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9491	2011-12-17	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9492	2011-12-27	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9493	2011-08-20	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9494	2011-10-02	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9495	2012-04-14	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9496	2012-01-02	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9497	2011-08-27	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9498	2011-12-21	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9499	2011-09-10	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9500	2012-04-21	39215	Vorm	Michel	0	0	0	0	0	0	0	0
9501	2011-12-21	59304	Vuckic	Haris	0	0	0	0	0	1	3	1
9502	2011-12-30	59304	Vuckic	Haris	0	1	0	0	0	0	1	0
9503	2011-08-28	59304	Vuckic	Haris	1	1	0	0	0	0	0	0
9504	2011-12-17	59304	Vuckic	Haris	1	0	0	0	0	0	0	0
9505	2011-12-20	19602	Vukcevic	Simon	0	1	0	0	0	0	2	1
9506	2011-12-03	19602	Vukcevic	Simon	1	0	0	0	0	0	1	0
9507	2011-12-17	19602	Vukcevic	Simon	0	1	0	0	0	0	0	0
9508	2011-09-17	19602	Vukcevic	Simon	1	0	0	0	0	0	0	1
9509	2011-10-23	19602	Vukcevic	Simon	0	1	0	0	0	0	1	0
9510	2011-12-11	19602	Vukcevic	Simon	0	0	1	1	0	1	0	0
9511	2011-12-26	19602	Vukcevic	Simon	1	0	0	0	0	0	0	0
9512	2011-12-21	20467	Walcott	Theo	0	0	0	0	0	1	0	0
9513	2011-12-10	20467	Walcott	Theo	0	1	0	0	0	1	2	0
9514	2012-01-02	20467	Walcott	Theo	0	1	0	0	0	1	2	1
9515	2011-11-19	20467	Walcott	Theo	0	1	0	0	0	2	1	1
9516	2011-09-24	20467	Walcott	Theo	0	0	0	0	0	1	2	3
9517	2012-04-16	20467	Walcott	Theo	0	0	0	0	0	0	0	1
9518	2011-10-23	20467	Walcott	Theo	0	1	0	0	0	0	0	0

Figure 1

2) Goals

With all of these data, my goal was to make specific groups containing the same kind of player. Then I had to be able to differentiate the better players in each position. Something a manager could ask if he was looking for a specific player.

3) Tools

As expected, in order to solve this problem, I used the software RapidMiner. At the beginning it is really difficult to understand the way it works because of the fullness potential things you can do. But when you get it well, it becomes easier. To answer my problematic consisting to make groups, I used as main operator, the clustering.

What is the operator clustering? Clustering is concerned « grouping objects together that are similar to each other and dissimilar to the objects belonging to other clusters ».

How does it work ? You juste have to select a number of groups (cluster) you want and then the software differentiate groups with same kind of player.

How does it do ? The software works with as dimensions as attributes you selected and makes a kind of cloud where it will gather players who are closest.

Here we can see that we use 8 clusters.

Later, we will talk about how choose the perfect number of cluster and why.

Figure 2

If we want our process runs, we have to follow a particular architecture, like a road.

Here it is my final process :

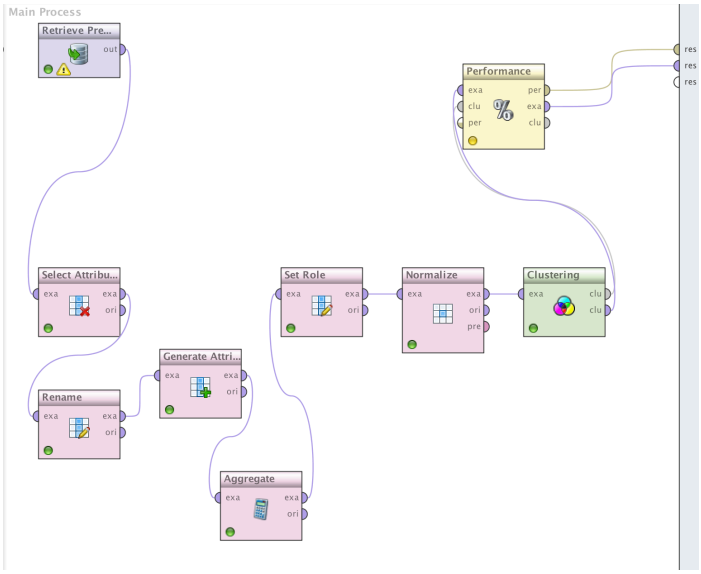

Figure 3

At the beginning of the process, it is our excel file gathering all our data. But to make it works, we have to modify it make it understandable by the clustering operator.

First of all I used an operator “Select attributes” in order to delete attributes I will not need like date, ID or team.

Then I used a “Generate attributes” to execute operations on my statistics. All my operations consisted to do a ratio Successful/Successful+Unsuccessful that represents the best performance indicator.

But I figured out that I had to delete spaces in attributes names if I wanted to use the operator “Generate attributes”. This is why the operator “Rename” is useful.

The “Aggregate” operator indicates which attributes we want to work on. Then you can guess that he is going to change for each position we will look for.

We need the “Set Role” operator to tell the software, in function of which attribute we want to work. It is our return value. It will be the same for each process and that it will be “Player Surname”.

Finally, the “Normalize” operator is used in order to give for each attribute the same scale. Because, without the same scale, the point positions in the “cloud” could be distorted du to wrong coefficients. It means that without this operator, for example, time played and goal scored will not have the same impact. So I used a scale between 0 and 10.

And I added a “Performance” operator indicating how big is our cloud. The bigger the cloud is, the less precise our results are.

III] Detection

1) Goalkeepers

I will show how to found goalkeepers who played this season and divide theme in three groups from the better to the worst.

I change all attributes to put just keeper skills like:

aggregation attribute	aggregation functions
ɔpearances	sum
ean Sheets	sum
ıves Made	sum
ıtches	sum
K Successful Distribution	sum

Figure 4

I choose to define 8 groups and found this:

Attribute	cluster_5	cluster_7	cluster_0	cluster_1	cluster_2	cluster_3	cluster_4	cluster_6
m(Appearances)	3.579	3.127	1.113	2.111	3.230	1.171	6.804	0.249
m(Clean Sheets)	6.471	3.529	1.412	0	0.004	0	0	0.015
m(Saves Made)	6.731	5.583	2.356	0	0	0	0.035	0.052
m(Catches)	6.718	4.952	2.087	0	0	0	0.021	0.051
m(GK Successful Distribution)	8.144	4.420	2.234	0.002	0.001	0	0.013	0.057

Figure 5.a

Figure 5.b

In our results we can see that the cluster 5 defines the better goalkeepers. The cluster 7 represents the good goalkeeper but not the better whereas the cluster 0 represents the worst keepers.

2) Defenders

In the same spirit, even if it was more complicated, I found the best defenders of the league.

I keep a number of 8 clusters and completely changed the attributes in order to put specifics characteristics of a defender. The thing more complicated here is that I had to create new attributes from older to make ratio. Then I had to rename older to delete spaces:

old name	new name
Aerial Duels won	Aerial_Duels_W
Duels lost	Duel_L
Duels won	Duel_W
Ground Duels lost	Ground_Duel_L
Ground Duels won	Ground_Duel_W
Successful Short Passes	Successful_Short_P
Unsuccessful Short Passes	Unsuccessful_Short_P
Tackles Lost	Tackles_L
Tackles Won'	Tackles_W

Figure 6

After that I created my ratios with the "Generate Attributes" operator:

attribute name	function expressions
Aerial_Duel_Ratio	Aerial_Duels_W/(Aerial_Duels_W+Aerial_Duels_L)
Duel_Ratio	Duel_W/(Duel_L+Duel_W)
Ground_Duel_Ratio	Ground_Duel_W/(Ground_Duel_L+Ground_Duel_W)
Short_Pass_Ratio	Successful_Short_P/(Unsuccessful_Short_P+Successful_
Tackles_Ratio	Tackles_W/(Tackles_L+Tackles_W)

Figure 7

Then I instantiated my new attributes in my "Aggregate" operator:

aggregation attribute	aggregation functions
Aerial_Duel_Ratio	sum
Appearances	sum
Blocks	sum
Duel_Ratio	sum
Ground_Duel_Ratio	sum
Interceptions	sum
Short_Pass_Ratio	sum
Tackles_Ratio	sum

Figure 8

And my results are:

Attribute	cluster_4	cluster_2	cluster_0	cluster_1	cluster_3	cluster_5	cluster_6	cluster_7
m(Aerial_Duel_Ratio)	8.083	4.302	0.983	0.757	1.830	2.427	2.300	0.179
m(Appearances)	7.912	3.418	1.451	2.961	2.948	3.413	2.081	0.347
m(Blocks)	7.696	5.445	0.576	0.129	0.644	2.004	3.039	0.083
m(Duel_Ratio)	7.549	3.810	1.228	1.542	2.436	3.354	2.224	0.275
m(Ground_Duel_Ratio)	7.643	3.817	1.257	1.471	2.556	3.562	2.268	0.279
m(Interceptions)	6.972	3.755	0.881	0.554	1.818	3.928	2.301	0.148
m(Short_Pass_Ratio)	7.889	3.643	1.511	3.082	3.011	3.633	2.170	0.420
m(Tackles_Ratio)	7.283	3.727	1.232	0.972	2.720	4.031	2.137	0.237

Figure 9.a

Figure 9.b

Then we can see that the better defenders are gathered in the cluster 4 and the others who are not as good are in the second.

3) Midfielders

The way I found the best midfielders is the same that I did for defender except for the attributes where I obviously had to create new ratio. And there are the attributes I choose for midfielders:

aggregation attribute	aggregation functions
Fouls Won in Danger Area inc pens	sum
Dribbles_Ratio	sum
Passes_All_Ratio	sum
Assists	sum
Third_Passes_Ratio	sum
Key Passes	sum
Pass Forward	sum
Third_Passes_M_Ratio	sum
Opposition_Passes_Ratio	sum
Short_Passes_Ratio	sum

Figure 10

And this is my result:

Attribute	cluster_1	cluster_3	cluster_0	cluster_2	cluster_4	cluster_5	cluster_6	cluster_7
sum(Fouls Won in Danger Area inc pens)	2.853	3.959	0.397	1.500	0.112	1.458	1.247	0.103
sum(Dribbles_Ratio)	5.988	5.664	1.022	5.272	1.116	2.968	2.211	0.246
sum(Passes_All_Ratio)	3.833	3.400	1.552	4.957	3.086	3.091	2.205	0.399
sum(Assists)	6.118	2.486	0.423	1.296	0.272	1.850	0.947	0.043
sum(Third_Passes_Ratio)	4.944	4.339	1.539	5.281	2.379	3.663	2.502	0.392
sum(Key Passes)	7.231	4.492	0.663	2.796	0.628	3.085	1.678	0.140
sum(Pass Forward)	2.911	2.024	1.133	5.267	4.526	2.126	1.295	0.237
sum(Third_Passes_M_Ratio)	3.991	3.521	1.485	4.836	2.684	3.165	2.216	0.375
sum(Opposition_Passes_Ratio)	4.382	3.879	1.544	5.080	2.529	3.386	2.390	0.396
sum(Short_Passes_Ratio)	3.928	3.461	1.572	4.950	3.446	3.143	2.221	0.418

Figure 11

Now we can see that the good midfielders are in the cluster 3 whereas the better are in the number 1. But who are they?

Figure 12.a

#	Player	Country	Team	Assists
1.	David Silva	Spain	Manchester City	17
2.	Antonio Valencia	Ecuador	Manchester United	14
3.	Mata	Spain	Chelsea FC	13
4.	Emmanuel Adebayor	Togo	Tottenham Hotspur	12
5.	Gareth Bale	Wales	Tottenham Hotspur	11
	Alex Song	Cameroon	Arsenal FC	11
7.	Kun Agüero	Argentina	Manchester City	10
	Nani	Portugal	Manchester United	10
	Robin van Persie	Netherlands	Arsenal FC	10
	Theo Walcott	England	Arsenal FC	10
11.	Samir Nasri	France	Arsenal FC Manchester City	9
	Stéphane Sessegnon	Benin	Sunderland AFC	9
	Ashley Young	England	Manchester United	9

Figure 12.b

We can see that apart for Walcott, the top 13 of best players in assists are all in the cluster number one.

If we look in the cluster 3 we will find Walcott, and then could understand why he is not is the cluster number 1.

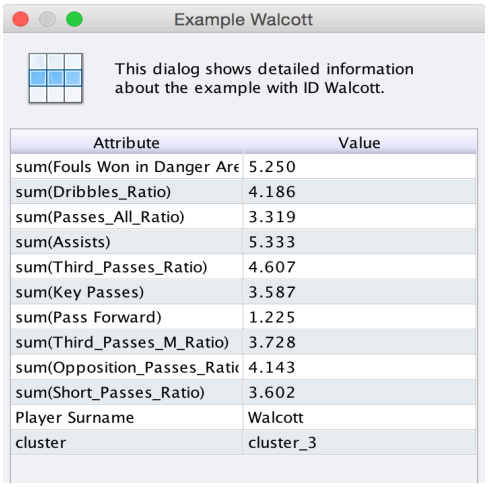

Figure 13

Here we can see that his statistics in assists are higher than others and good enough to be in the cluster 1.

But his others statistics are to lower, this is why he was too far from the “cloud” 1 and is in the 2.

But actually even if it is a practice way to check our results, it is not the good way to do and we cannot do that in each case.

This is why we use a specific “Performance” operator. I am going to show how it works with forwards.

4) Forwards

I decided to know who the best forwards were during the season 2011/2012. Like for defenders and midfielders, I had to create new attributes and implement theme in my "Aggregate" operator.

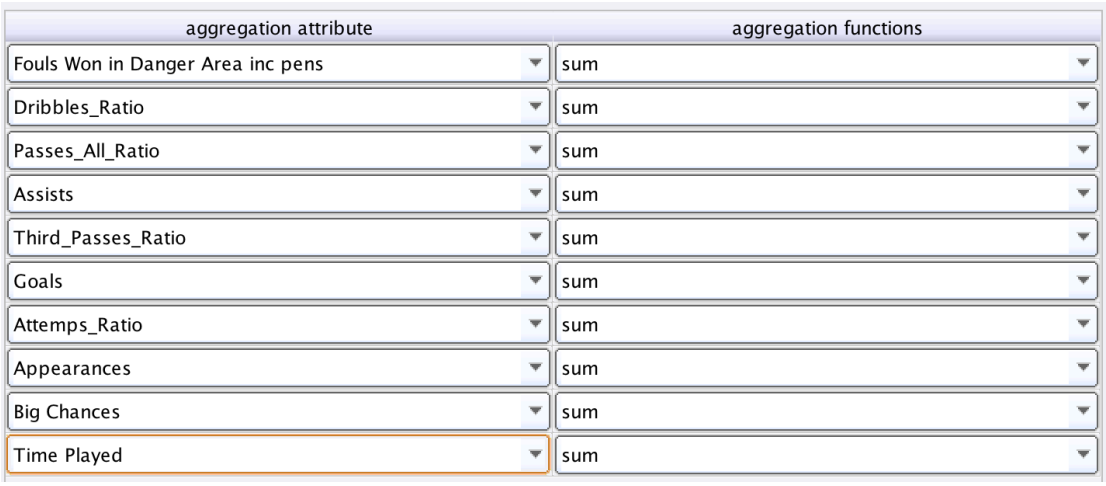

Figure 14

And my results are :

Attribute	cluster_0	cluster_1	cluster_5	cluster_2	cluster_3	cluster_4	cluster_6	cluster_7
sum(Fouls Won in Danger Area inc pens)	3.471	2.244	4.350	0.970	0.542	0.274	0.103	1.896
sum(Dribbles_Ratio)	4.291	2.528	6.763	4.225	1.391	1.116	0.250	4.262
sum(Passes_All_Ratio)	3.237	2.665	4.394	3.687	1.729	2.989	0.435	3.074
sum(Assists)	2.980	1.349	3.520	0.976	0.568	0.484	0.046	3.667
sum(Third_Passes_Ratio)	4.018	3.181	5.417	4.039	1.840	2.526	0.417	3.794
sum(Goals)	5.294	1.884	1.840	0.374	0.359	0.306	0.037	0.944
sum(Attemps_Ratio)	6.615	3.732	4.520	1.250	0.866	0.373	0.155	2.662
sum(Appearances)	3.208	2.757	4.256	3.447	1.679	3.139	0.377	2.993
sum(Big Chances)	5.269	1.638	1.383	0.292	0.271	0.238	0.037	0.845
sum(Time Played)	3.194	2.412	4.413	3.677	1.411	3.501	0.297	2.772

Figure 15

We can easily see that the top scorers are in the cluster 0 whereas the cluster 1 represents players who plays less but have good enough statistics. The cluster 5 represents provocative forwards who scores less but did more dribbles and assists.

Then we could do like we did for midfielders, but we will use the "Performance" operator.

I decided to draw a simplification of how clustering operator works.

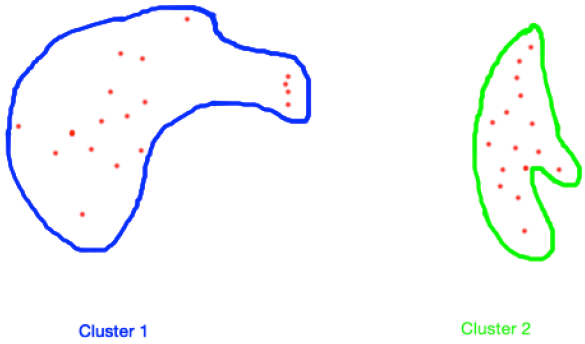

Figure 16

Here you have to imagine that each red point is a player, positioned in the space function of his attributes. If I ask RapidMiner 2 clusters, this what he is going to do.

As you can see, the cluster 1 is larger than the cluster 2, it mean that the cluster 2 contain players more similar than the first. In order to know how specific is a cluster, we use a "Performance" operator.

The performance operator I choose is the "Distance Performance" operator. Its function is to calculate in a cluster, the average distance between each "red point" and the gravity center. The smaller is the final average, the more specific our cluster.

PerformanceVector

```
PerformanceVector:
Avg. within centroid distance: -5.559
Avg. within centroid distance_cluster_0: -24.880
Avg. within centroid distance_cluster_1: -8.545
Avg. within centroid distance_cluster_2: -6.642
Avg. within centroid distance_cluster_3: -3.420
Avg. within centroid distance_cluster_4: -3.624
Avg. within centroid distance_cluster_5: -25.754
Avg. within centroid distance_cluster_6: -0.748
Avg. within centroid distance_cluster_7: -8.748
Davies Bouldin: -1.369
```

Figure 17

Here you can see that the clusters 0, and 5, our strikers, have the biggest average. It means that for the attributes I put, strikers are very different between themselves, but they are less different than other cluster's players.

This is how we could represent clusters in a draw:

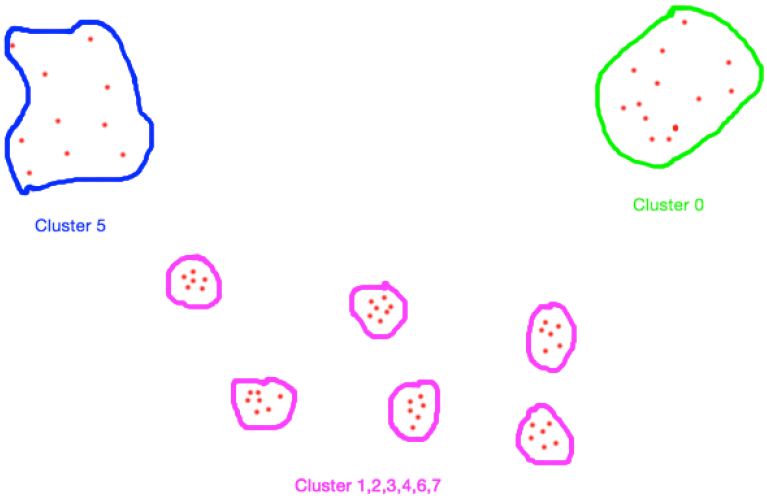

Figure 18

How to do if we want something more specific?

By increasing the number of cluster. Take a look on this new draw to understand easier:

Instead of:

Figure 16

Do:

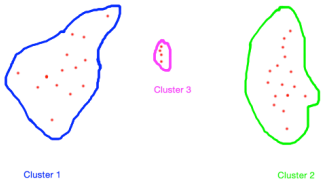

Figure 19

As you can see, RapidMiner give us an average (the first average Figure 17) of all cluster's average distance.

The smaller is this average the more specific are all clusters. Then, I am going to increase the number of cluster (k) in order to see how this average changes.

k	Performence Indicator
2	12,412
3	9,354
4	8,076
5	7,386
6	6,523
7	5,983
8	5,559
9	5,24
10	4,887
12	4,321
14	4,163
16	3,916
18	3,843
20	3,64
25	3,44
30	3,008
40	2,783
50	2,517

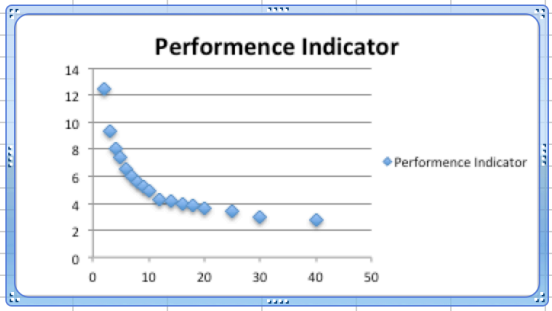

Figure 20

This is my result. The curve represents the number of clusters in function of the performance indicator (final average) for the forward's process. As expected, this curve decreases when k increases.

The preferential number of cluster is when the curve stops to decrease, the perfect number should be around 13.

Conclusion:

The more difficult thing in this project was to know which operator choose, and how use it. RapidMiner is very powerful, and it is a software I did not know until this semester. Here we saw that by using good attributes and with good tools and mathematical reflection, we could reach our main goal: Making numbers meaningful.

Bibliography:

http://www.worldfootball.net/assists/eng-premier-league-2011-2012/

http://ai.arizona.edu/research/sports_data/

http://www.fil.univ-lille1.fr/~decomite/ue/MFFDD/tp1/rapidminer.pdf

http://en.wikipedia.org/wiki/Data_mining