**Goal 1: Profile of Major Buyers both on website and in the shop**

Success Criteria: We can assume that we have completed this goal if we manage to identify different profiles that constitute the major buyers online/in the shop. After applying our algorithm we should be able to identify the archetypes of the top buyers meaning their age, country, gender, status… Ideally our output should be something like: French single women that are between 18 and 25 are top buyers online.

**First Method: A decision tree**

Dataset Selected: To fulfill this goal, we naturally need data that allows us to make out a profile, so we will need as input the attributes: id, range of age, gender, status, country and work. Moreover we need the attributes we are going to base our study on meaning the budget of each person and the buying frequency. To fit the algorithm I wanted to use I had to add a new attribute to tag if the customer was a top buyer or not, I had to use an excel function (=if ((budgetfrequency>2 and budget>4);”topbuyer”) if the customer didn’t fill the condition he was tagged as “false”. I named this new attribute class.

Technique chosen: I decided to work with the decision tree technique. Since the decision tree technique is commonly used to help identify a strategy most likely to reach a goal, here our goal is to identify the top buyers, the class attribute would be the one I added, the branches that have as final node “topbuyers” would constitute the different profiles of these top buyers.

Algorithm chosen: I have chosen to work with Weka’s Reptree which is fast decision tree learner. The algorithm builds a decision/regression tree using information gain/variance reduction and prunes it using reduced-error pruning (with back fitting). As I said before my class attribute is going to be whether the customer is a top buyer or not the two final nodes being either “topbuyer” or “false”. Before running this algorithm I sorted the attributes and removed on Weka the irrelevant ones which are id (An artificial key that hasn’t any meaning) and the budget, buying frequency (since my class attribute replaced these two). The output is a tree.

**Tree for the Website**

The tree is on the following page, we have identified 5 profiles of top buyers on the online website.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Country | Gender | Status | Work | Age |
| Spain | Female | Not given | Not given | <84.5 and >=71.5 |
| Spain | Not given | Divorced | Not given | <51.5 |
| Spain | Not given | Single | Not given | <62.5 |
| Poland | Not given | Divorced | Not given | >=30 and <38 |
| Poland | Male | Divorced | Not given | >=52 and <71.5 |
| Italy | Female | Widowed | < 3  Meaning she is either a  1: Farmer or a 2: Artisan, shopper, CEO. | Not given |

Explanations: As I said following the branches that have as final node topbuyer leads you to find these profiles.

**Tree for the Shop**

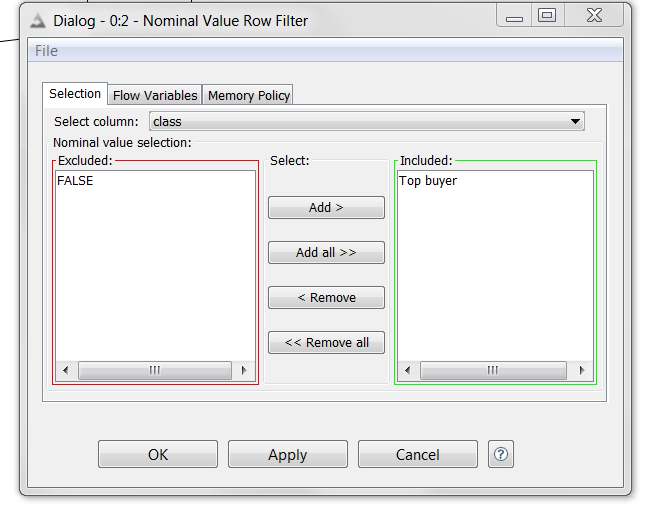
The tree we obtain being huge I divided it to 4 screenshots. We obtain here 17 profiles.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Country | Gender | Status | Work | Age |
| Turkey | Not given | Single | 4,5,6 | >46.5 |
| Poland | Not given | Single | 1,2 | Not given |
| France | Not given | Single | 4,5,6 | >=47.5 |
| England | Not given | Single | 4,5,6 | Not given |
| Russia | Not given | Single | 4,5,6 | Not given |
| Not given | Not given | Civil Union | 2,3,4,5,6 | <22.5 |
| Germany | Not given | Civil Union | 2,3,4,5,6 | >=22.5 |
| Poland | Not given | Civil Union | 2,3,4,5,6 | >=22.5 and <68 |
| France | Female | Civil Union | 2,3,4,5,6 | >=22.5 |
| Turkey | Not given | Civil Union | 2,3,4,5,6 | >=22.5 |
| Germany | Not given | Married | 1 | Not given |
| Germany | Not given | Divorced | 1 | Not given |
| Poland | Not given | Not given | 1 | Not given |
| Spain | Not given | Widowed | 1 | Not given |
| Spain | Not given | Married | 1 | Not given |
| Russia | Not given | Cohabiting | 1 | Not given |
| Russia | Not given | Single | 1 | Not given |

**Second method: Clustering**

This second method is maybe more relevant considering that these technique is very much used to find profiles.

Dataset Selected: As with the first method, we will need the attributes that give us the profile meaning: the age, work, status… I also had to tag the customers in order to know who were “top buyers” and who were not. However, the main issue is that clustering algorithms require numerical attributes rather than nominal. If I run a clustering algorithm without modifying my data I would have clusters based on only age and work which is irrelevant. Considering this constraint, I created derived numerical attributes from my nominal attributes, I changed the gender from male female to 1, 2 for the status I considered people who lived alone (single, divorced, widowed) as one part (I labeled them as 1) the other were assigned the value 2 for the country I put eastern Europe countries as 1 and the others as 2. All that so my data could fit the clustering algorithm. Furthermore what I did with KNIME was removing all the rows that were not “top buyers” (were labeled as false) as can be seen below:

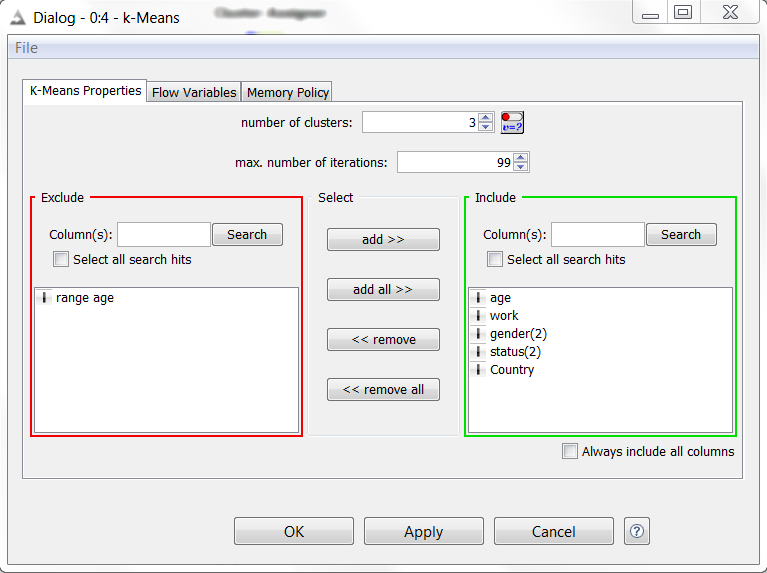
Technique chosen : Clustering is the task of grouping a set of objects in such a way that objects in the same group behave the same way, are more similar to each other than to those in other groups. Since our goal is to obtain different profile of top buyers, each group of top buyer will constitute a cluster with its own characteristics.

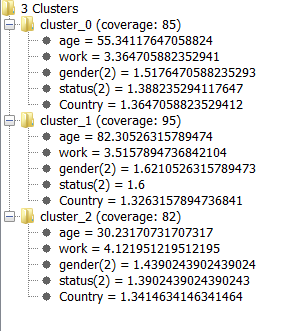
Algorithm chosen: *k-Means* is a rather simple but well known algorithms for grouping objects, clustering. Again all objects need to be represented as a set of numerical features. In addition the user has to specify the number of groups (referred to as k) he wishes to identify. We are also going to run the fuzzy c-means which is very similar to the k-means and only differs in some very technical aspects.

**Clustering for the website**

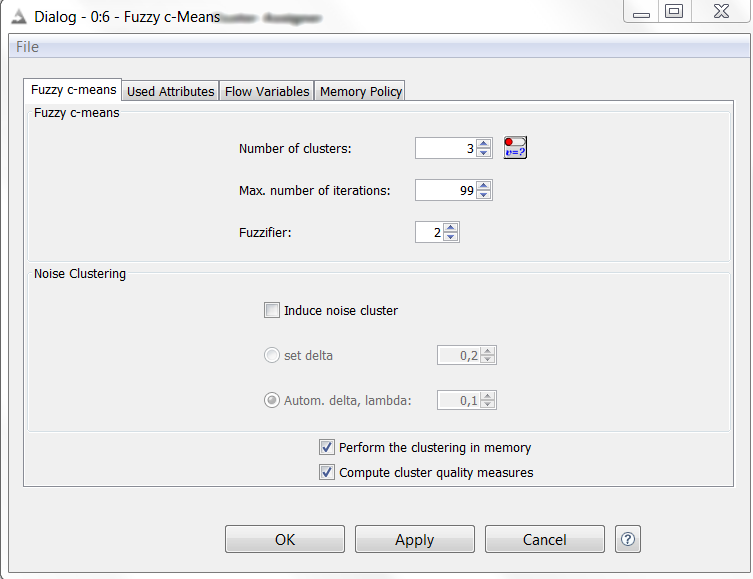
With K-means

Here is the configuration I used to run k-means on KNIME:

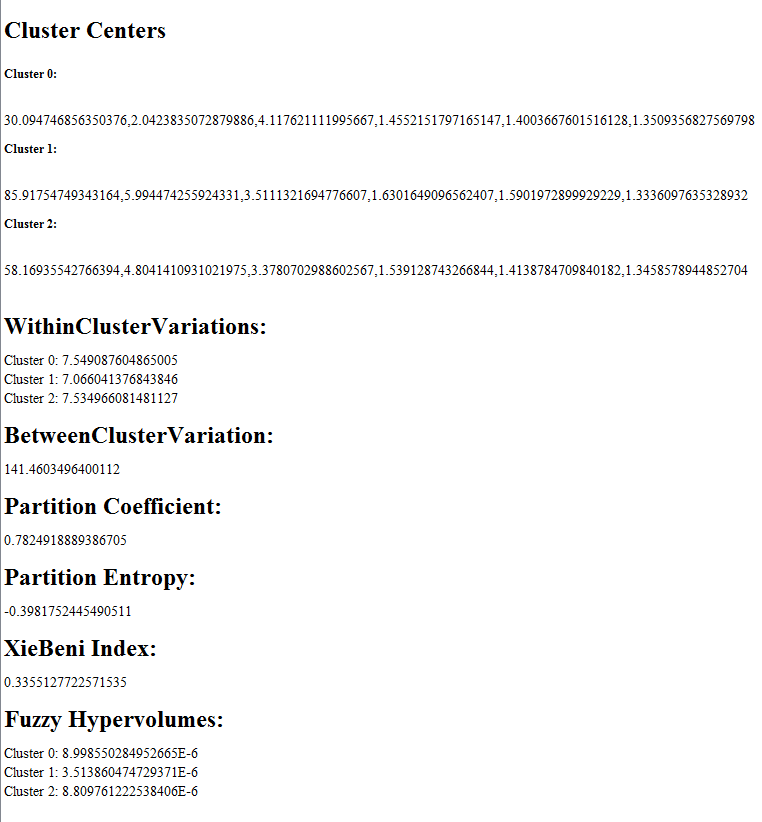
And here are the results:



With Fuzzy C-means



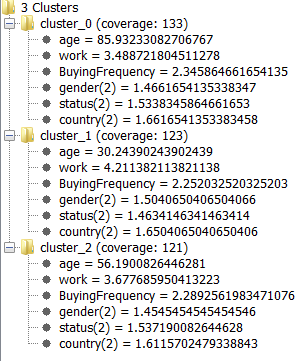
And here are the results:

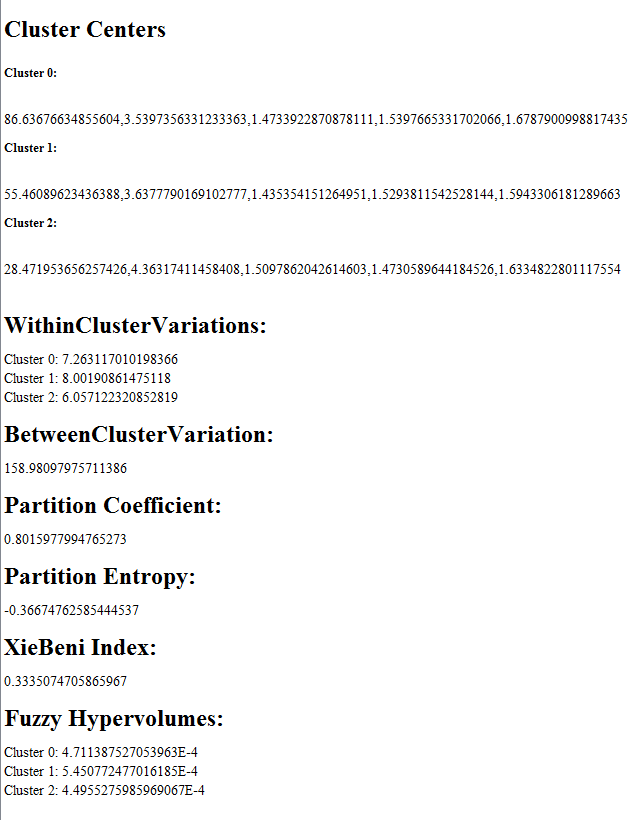


**Clustering for the shop**

We operate in the exact same manner for the shop, meaning we turn the nominal attributes into numerical, we only keep the topbuyers, we use the same configurations and then we do clustering, the results are the following:

With K-means



 With Fuzzy C-means

Conclusion

The clustering seems more relevant with the task we want to reach, indeed the decision tree doesn’t give us profiles of top buyers but rather profiles of people who are **not not** topbuyers, which makes this method kind of inaccurate for this DM goal. At the opposite clustering methods give us directly the profiles of topbuyers, therefore it’s the most accurate method.

**Goal 2: behavior of shoppers**

**Goal 2a,i : behavior of shoppers, general, for website**

**Data set**: as specified in data preprocess, goal2abi.csv

**Success Criteria**: we want to obtain different representative behaviors of shoppers.

**Techniques**:

* Clustering
  + Hierarchical Clustering
  + Fuzzy c-means

Clustering

Hierarchical Clustering

In order to know the number of clusters, we are going to use the hierarchical clustering.

**Parameters** :

* Number output cluster : Which level of the hierarchy to use for the output column.
* Distance function : Which distance measure to use for the distance between points.
* Linkage type : Which method to use to measure the distance between points (as described above)
* Distance cache : Caching the distances between the data points drastically improves performance especially for high-dimensional datasets. However, it needs much memory, so you can switch it off for large datasets.

**Input** : The data that should be clustered using hierarchical clustering. Only numeric columns are considered, nominal columns are ignored.

**Output** : The input data with an extra column with the cluster name where the data point is assigned to.

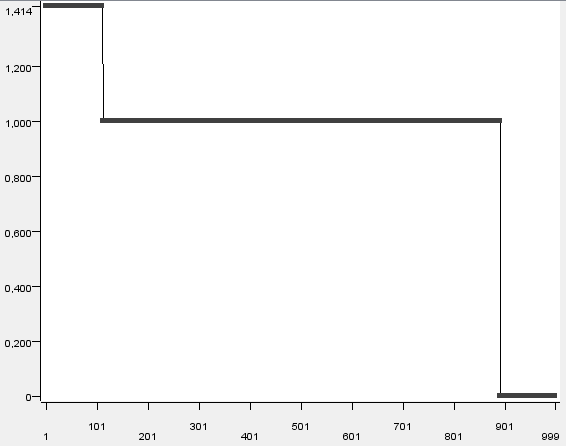
**Configuration 1:**

First of all we will try to find a clustering with all the attributes in the data set.

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | All |
| Number Cluster | 3 |
| Distance function | Euclidean |
| Linkage type | Single |
| Distance cache | Yes |

**Results:**

Distance



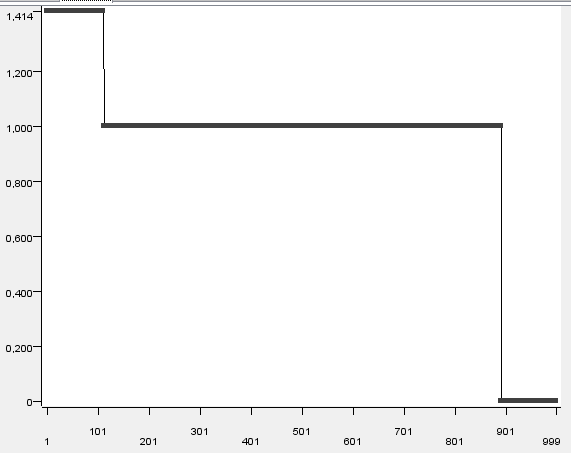
**Configuration 2:**

To see if 3 is the right number, we use the same algorithm with 4.

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | all |
| Number Cluster | 4 |
| Distance function | Euclidean |
| Linkage type | Single |
| Distance cache | Yes |

**Results** : Distance

We can see that 3 is the right number.



Fuzzy c-means

Now we will use the Fuzzy c-means clustering.

**Parameters** :

* Number of clusters :Number of clusters to use for the algorithm.
* Maximum number of iterations : This is the maximum number of iterations to be performed.
* Fuzzifier : Indicates how much the clusters are allowed to overlap.
* Induce noise cluster : Whether to induce a noise cluster or not.
* Set delta : Delta is the fixed distance from every datapoint to the noise cluster.
* Set delta automatically, specify lambda : Delta is updated in each iteration, based on the average interpoint distances. However, a lambda paramater has to be set, according to the shape of the clusters.
* Perform the clustering in memory : If this option is selected, the clustering is performed in the memory, which speeds up the process.
* Compute cluster quality measures :Whether to calculate quality measures for the clustering. This can be time and memory consuming with large datasets.

**Input** : Datatable with training data. Must be normalized.

**Output** : Input table extended by cluster membership

**Configuration 1:**

Normalization with “Normalizer”, min-max 0-1, for all attributes.

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | All numerics (budget, frequency, changinmind, looking for attributes) |
| Number Cluster | 3 |
| Max number iteration | 99 (default) |
| Fuzzifier | 2 (default) |
| Induce noise cluster | No |
| Set Delta | / |
| Set delta automatically, specify lambda | / |
| Perform in memory | Yes |
| Compute cluster quality measures | Yes |

**Results:**

**WithinClusterVariations:**

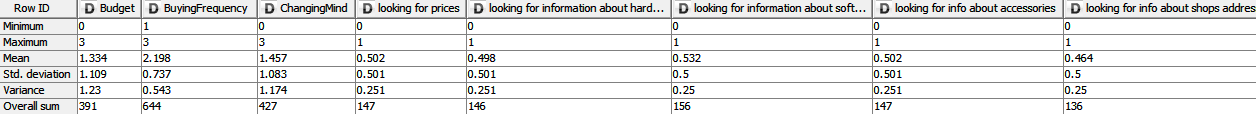
Cluster 0: 2.0896162064715282  
Cluster 1: 1.940692477946374  
Cluster 2: 2.051209640417251

**BetweenClusterVariation:**

3.6981121526267517

The within cluster variation isn’t that good. With the statistics view, we will have a better idea of the distribution of attributes for each cluster. (after denormalizer)

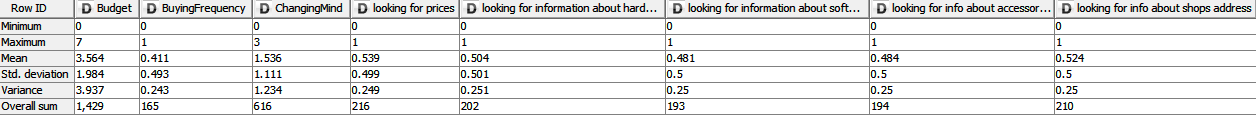
**Cluster 0 :**



**Cluster 1 :**



**Cluster 2 :**



We decide to keep the budget, frequency and changingmind.

**Configuration 2:**

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | Budget, Frequency and ChangingMind |
| Number Cluster | 3 |
| Max number iteration | 99 (default) |
| Fuzzifier | 2 (default) |
| Induce noise cluster | No |
| Set Delta | / |
| Set delta automatically, specify lambda | / |
| Perform in memory | Yes |
| Compute cluster quality measures | Yes |

**Results:**

**WithinClusterVariations:**

Cluster 0: 0.4204324960710409 (blue)  
Cluster 1: 0.4181562332570936 (red)  
Cluster 2: 0.4737703200727217 (green)

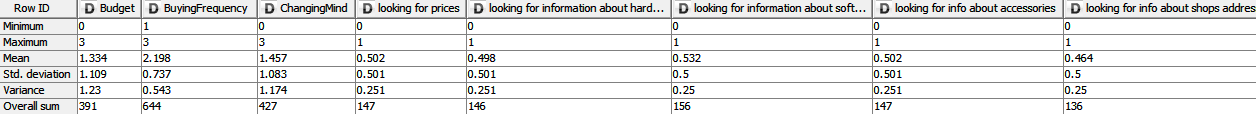
**BetweenClusterVariation:**

0.519318304286104

We can see that even if the WithinClusterVariations is smaller than before, the BetweenClusterVariation isn’t high enough.

We decide to keep the Configuration 1 for analysis, we prefer to have a higher difference between two clusters.

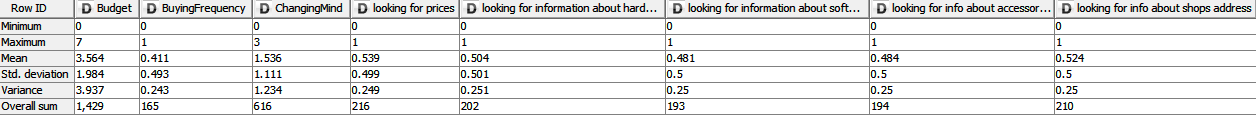
**Cluster 0 : (blue)**



**Cluster 1 : (green)**



**Cluster 2 : (red)**



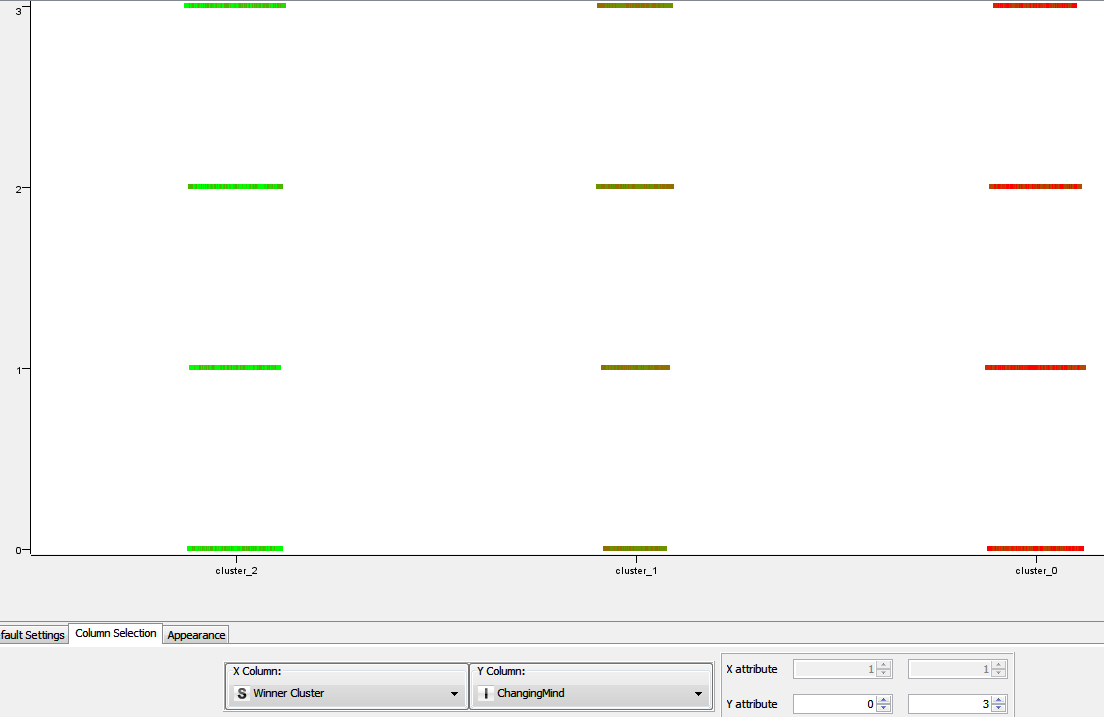
For the cluster 0, the budget is low (between 0-3), whereas for the cluster 1 it’s high : between 4-7.

We can see that for these two clusters, the buying frequency is never null.

We are going to use others viewing tools to have a better ideas of the different behaviors.

* **ChangingMind= f(Winner Cluster)**

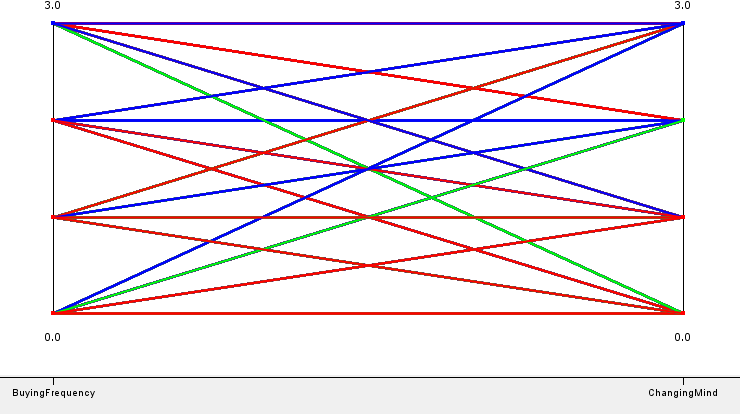
Budget Colored: Red for lowest budget, Green for highest



We can see that the budget is very well marked between clusters: the cluster 0 has the lowest budget, whereas cluster 1 has a medium and the cluster 2 has the highest.

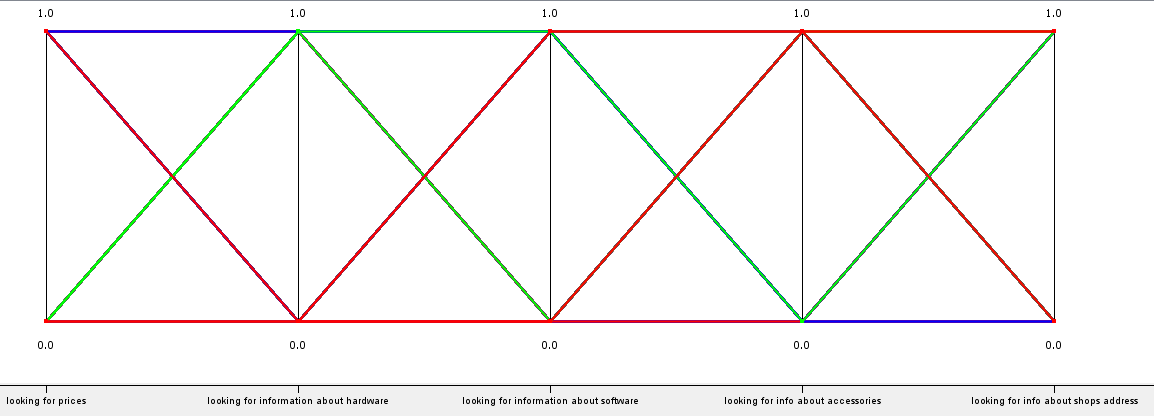
For the changingMind, we can see that less people in the cluster 1 never change their mind. For the other clusters we can’t see any representative difference here.

* **Buying Frequency and ChangingMind, cluster colored. (cluster 0 blue / cluster 1 green / cluster 2 red)**



For the cluster 1: people with a high buying frequency never change their minds, but for cluster 2, they change their mind more often. We know that for cluster 2 they also have a higher budget.

* **Looking for attributes, cluster colored (cluster 0 blue / cluster 1 green / cluster 2 red)**



For the cluster 2, this is non representative. For cluster 0, we can see that they are more looking for information about prices and hardware, but they aren’t interested in accessories and software. This cluster has a the lowest budget, we can consider offering cheap hardware on the website for this population.

The buyers from cluster 1 aren’t interested in prices, but in software and shop address. They have a medium budget and a higher changingMind, we can think about creating a link on the shop address research page on the website, with non-expensive software ads.

**Evaluation :**

* Data Mining evaluation: the variations (between and within) are good enough.
* Business Success evaluation: we have 3 different representative website buyers’ behaviors.

**Goal 2a,ii : behavior of shoppers, general, for shops**

**Data set:** as specified in data preprocess, goal2abii.csv

**Success Criteria:** we want to obtain different representative behaviors of shoppers.

**Techniques:**

* Clustering
  + Fuzzy c-means

Clustering

Fuzzy c-means

**Parameters** :

* Number of clusters :Number of clusters to use for the algorithm.
* Maximum number of iterations : This is the maximum number of iterations to be performed.
* Fuzzifier : Indicates how much the clusters are allowed to overlap.
* Induce noise cluster : Whether to induce a noise cluster or not.
* Set delta : Delta is the fixed distance from every datapoint to the noise cluster.
* Set delta automatically, specify lambda : Delta is updated in each iteration, based on the average interpoint distances. However, a lambda paramater has to be set, according to the shape of the clusters.
* Perform the clustering in memory : If this option is selected, the clustering is performed in the memory, which speeds up the process.
* Compute cluster quality measures :Whether to calculate quality measures for the clustering. This can be time and memory consuming with large datasets.

**Input** : Datatable with training data. Must be normalized.

**Output** : Input table extended by cluster membership

We replace the buying process time in string with the equivalent number of days, to be able to use it with the clustering.

**Configuration 1:**

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | All numeric (budget, frequency, changing mind, and buying process time) |
| Number Cluster | 3 |
| Max number iteration | 99 (default) |
| Fuzzifier | 2 (default) |
| Induce noise cluster | No |
| Set Delta | / |
| Set delta automatically, specify lambda | / |
| Perform in memory | Yes |
| Compute cluster quality measures | Yes |

**Results:**

**WithinClusterVariations:**

Cluster 0: 0.48225287580016873  
Cluster 1: 0.4934709917490485  
Cluster 2: 0.6438628383572099

**BetweenClusterVariation:**

0.7754534150413014

The BetweenClusterVariation is too close to the within, and not high enough. We are selectionning less attributes.

**Configuration 2:**

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | Budget, frequency, process time |
| Number Cluster | 3 |
| Max number iteration | 99 (default) |
| Fuzzifier | 2 (default) |
| Induce noise cluster | No |
| Set Delta | / |
| Set delta automatically, specify lambda | / |
| Perform in memory | Yes |
| Compute cluster quality measures | Yes |

**Results:**

**WithinClusterVariations:**

Cluster 0: 0.35620661559632216  
Cluster 1: 0.46237926667711554  
Cluster 2: 0.34466978814951704

**BetweenClusterVariation:**

1.2060546370918472

**Configuration 3:**

Let’s have an idea if we choose 4 clusters

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | Budget, frequency, process time |
| Number Cluster | 4 |
| Max number iteration | 99 (default) |
| Fuzzifier | 2 (default) |
| Induce noise cluster | No |
| Set Delta | / |
| Set delta automatically, specify lambda | / |
| Perform in memory | Yes |
| Compute cluster quality measures | Yes |

**Results:**

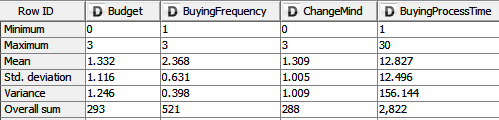
**WithinClusterVariations:**

Cluster 0: 0.259792401422038  
Cluster 1: 0.2752824921686866  
Cluster 2: 0.46077086005559326  
Cluster 3: 0.3169731171735624

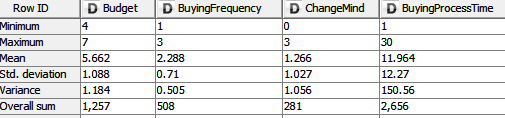
**BetweenClusterVariation:**

1.588054792967043

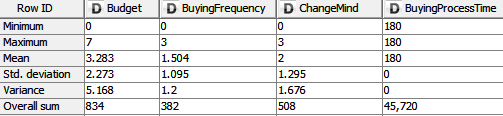
**Cluster 0:**



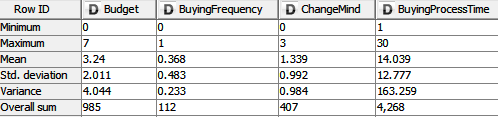
**Cluster 1:**



**Cluster 2:**



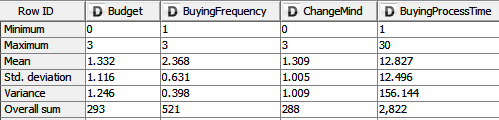
**Cluster 3:**



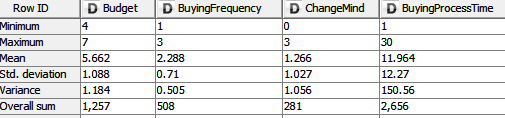
We have 4 buyers’ behavior representative clusters.

Regarding to the results, we decide to keep the configuration 3.

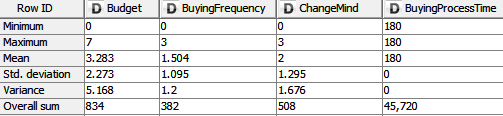
**Cluster 0: (blue)**



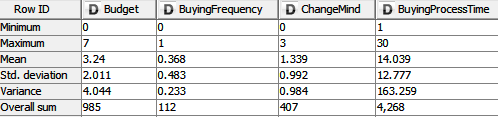
**Cluster 1: (green)**



**Cluster 2: (red)**



**Cluster 3: (pink)**

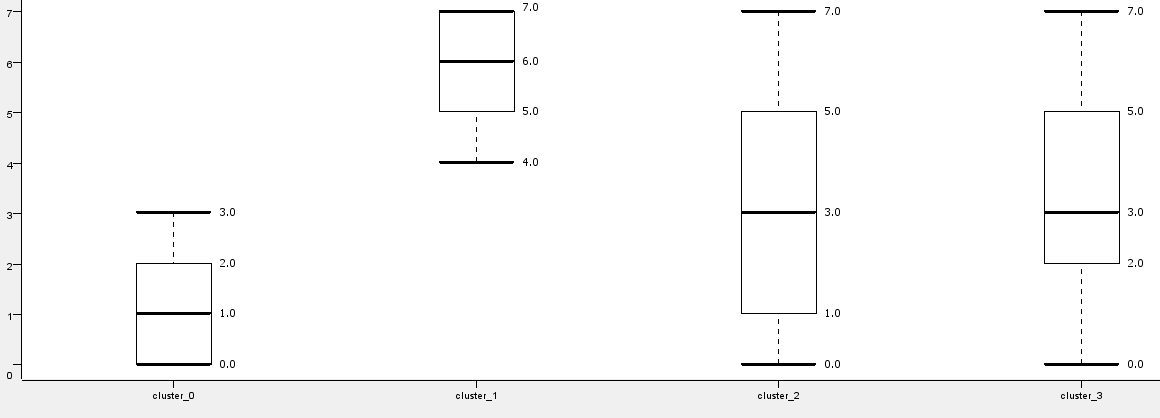


For the cluster 0, the budget is low (0-3), for the cluster 1, high (4-7). For both, the buying frequency is never 0, and the buying process time never high (180 days).

For the cluster 2, the changingMind is higher, with a mean of 2, and the buying process time is always high, 180 days.

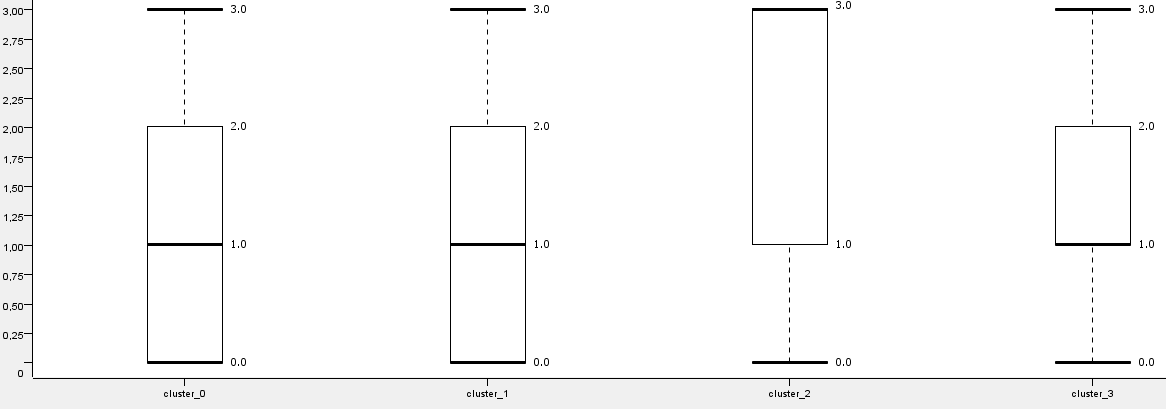
For the cluster 3, the buying frequency is low (0-1), as the buying process time (1-30).

* **Budget by cluster with a BoxPlot tool:**



We can see that for clusters 2 and 3 the budget is well distributed. Cluster 1 has the highest budget, with a mean at 6, whereas the cluster 0 has the lowest, with a mean at 1.

* **ChangingMind by cluster with a BoxPlot tool :**

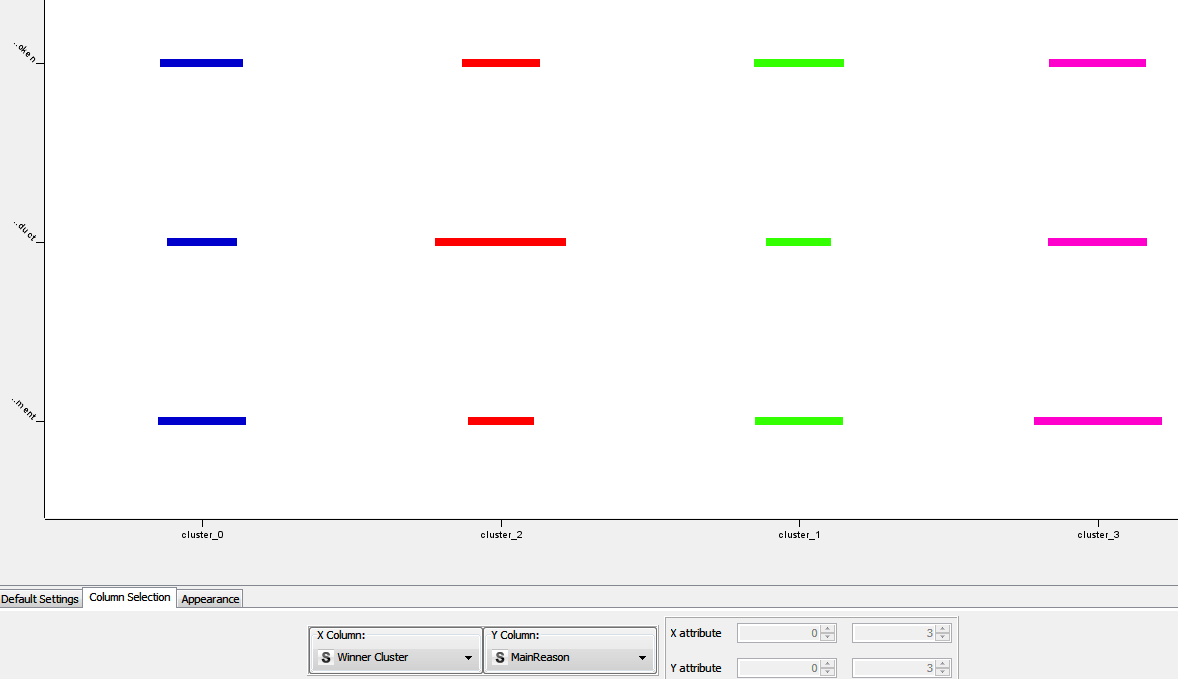


We can see that the distribution of changindMind is the same for cluster 0 and 1 whereas they don’t have the same budget.

* **MainReason= f(Cluster), cluster colored :**

**(cluster 0 blue / cluster 1 green / cluster 2 red / cluster 3 pink)**

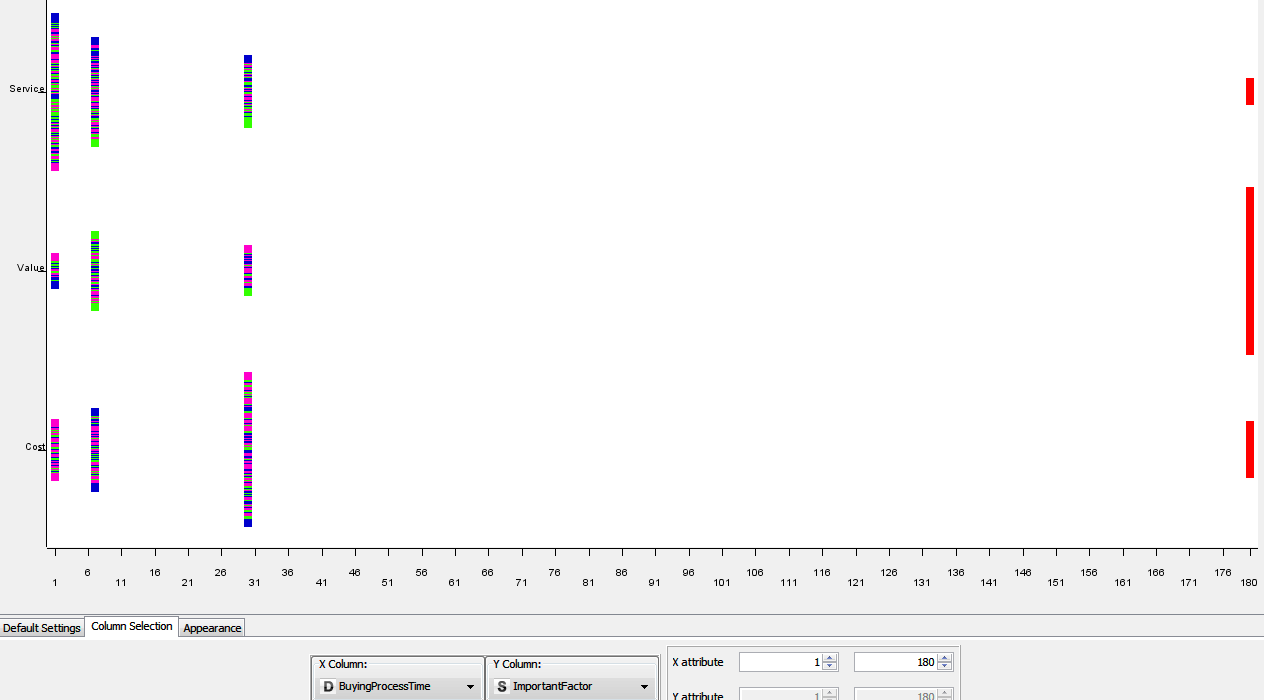
(MainReason: from top to down: Previous is broken/ Update previous product / New requirement)



For cluster 0, 2 and 3, people buy less to update a previous product. For cluster 2 it’s the opposite, they buy more to update a previous product.

* **ImportantFactor = f(Buying Process Time), cluster colored.**

**(cluster 0 blue / cluster 1 green / cluster 2 red / cluster 3 pink)**

****

We can see that the cluster 2 is the one with the highest buying process time. For these buyers, the most important factor will be the Value.

For the 3 others clusters, we can see a dependence between buying process time and the important factor. Most of the buyers who think Cost is important, will have a buying process time of 30 days. For those who prefer the service, they will have a very short buying process time, 1 day. We can assume that by hiring qualify sellers, whom advices are relevant, and offering a large rank of services in the shop, these people will be more driven to buy in the day.

**Evaluation**

* Data Mining evaluation: the variations (between and within) are good enough.
* Business Success evaluation: we have 4 different representative shop buyers’ behaviors.

**Goal 2b,i : behavior of shoppers, country depending, for website**

**Data set :** as specified in data preprocess, goal2abi.csv

**Success Criteria:** we want to have an idea of how the behaviors of the website buyers change depending on the country.

**Techniques:**

* Classification :
  + Naive Bayes
* Clustering
  + K-means

Classification

Naive Bayes

With this algorithm, we will try to define specific behaviours depending on the country.

**Parameters**:

* Classification Column : The class value column.
* Skip missing values (incl. class column) : The node ignores missing values in the model if this option is ticked. If it's not ticked the node treats the missing values as a normal value and considers them during the class probability calculation.
* Maximum number of unique nominal values per attribute : All nominal columns with more unique values than the defined number will be skipped during learning. If the column contains missing values and the 'Skip missing values' option is not skipped the missing value counts as one value!

**Input**: Training data

**Output**: Learned naive Bayes model

**Configuration 1:**

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | all |
| Classification Column | country |
| Skip missing values | no |
| Maximum number of unique nominal values per attribute | 20 (default) |

**Results:**

We use 96% of the data to learn, and then with the predictor we try to say the country of the 4% left.

False: 36/ True: 4

We can see that this classification isn’t good. We are going to use less attributes.

**Configuration 2:**

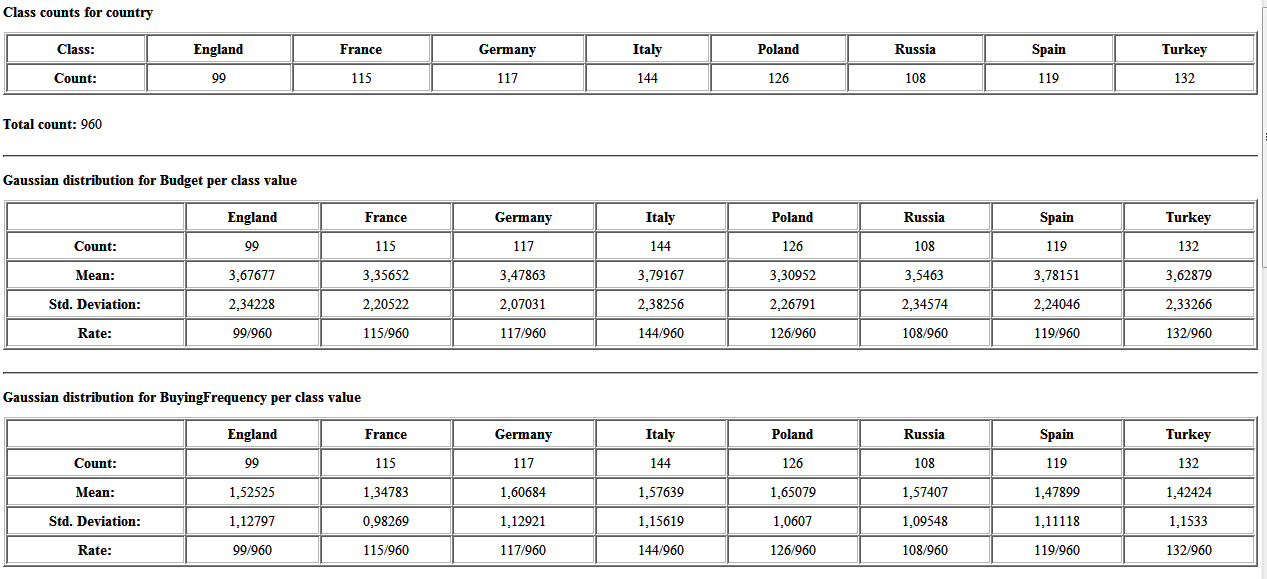
|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | BuyingFrequency, Budget, ChangingMind |
| Classification Column | country |
| Skip missing values | no |
| Maximum number of unique nominal values per attribute | 20 (default) |

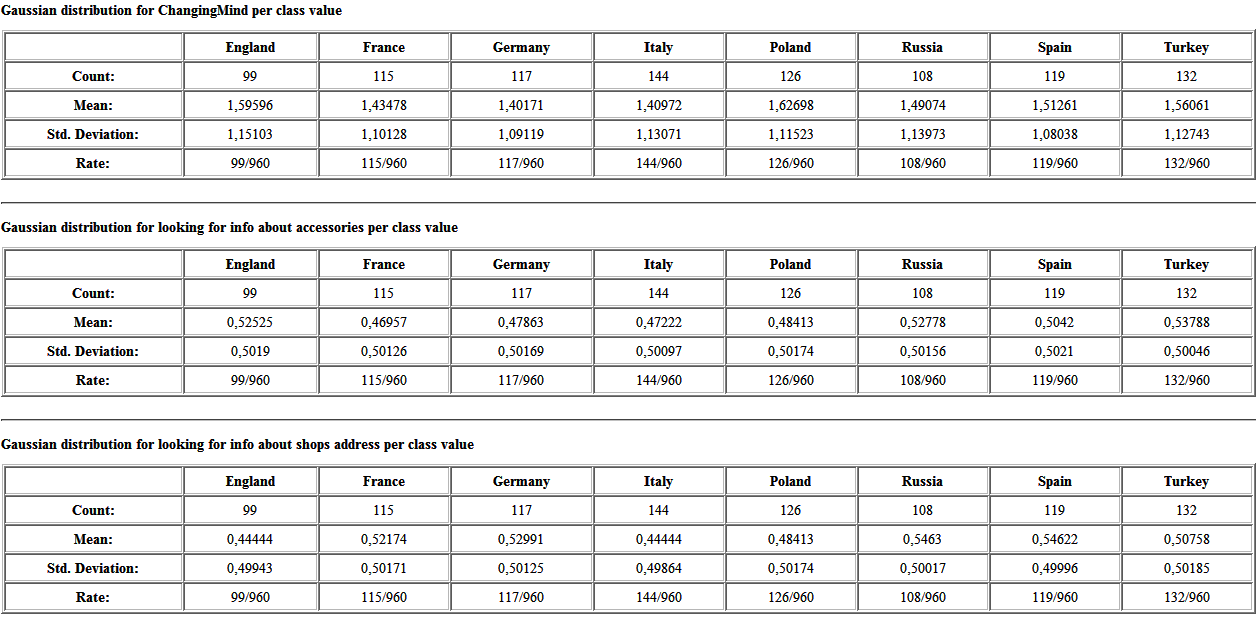
**Results:**

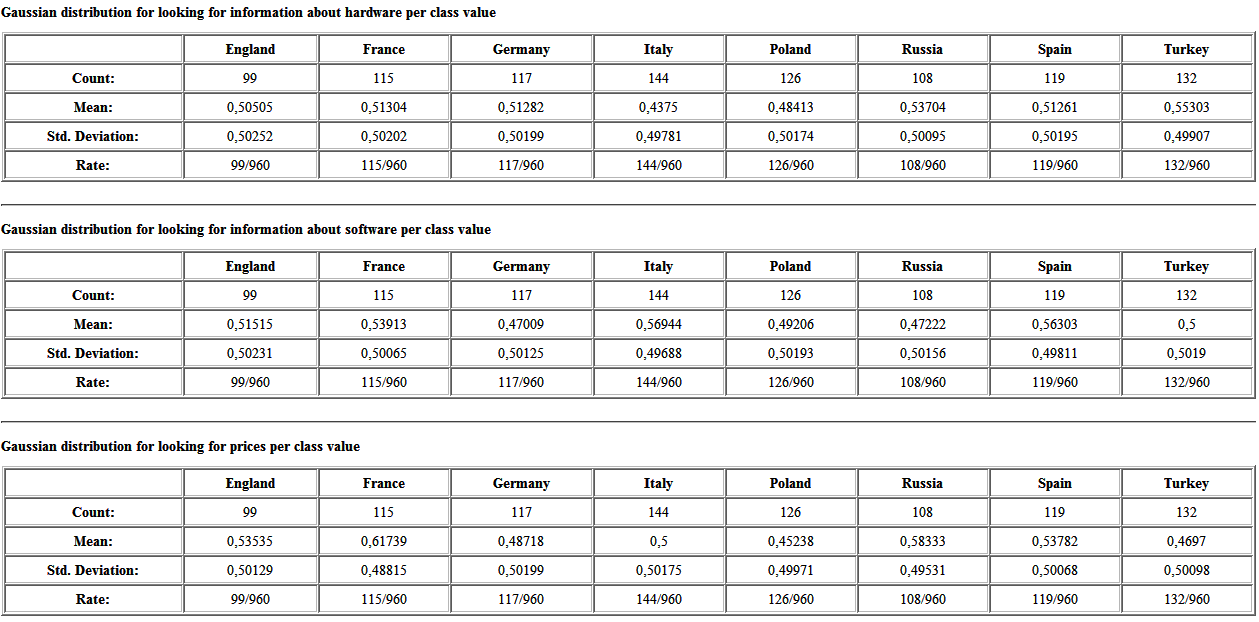
False: 33/ True: 7

These results are better, but still not good enough.

But we can use the Naïve Bayes Learner results to have an idea of the difference between attributes depending on the country.







We can see that for most of the attributes, the mean is quite the same and the deviation also.

We are going to use now the K-means clustering, to know if inside each country, we can have different representative behaviors, and see if we find the same clusters in the different countries.

AS the results were better with the 2nd configuration, we will use the budget, frequency and chinging mind.

Clustering

K-means

**Parameters** :

* number of clusters : The number of clusters (cluster centers) to be created.
* max number of iterations : The number of iterations after which the algorithm terminates, independent of the accuracy improvement of the cluster centers.

**Input** : Input to clustering. All numerical values and only these are considered for clustering.

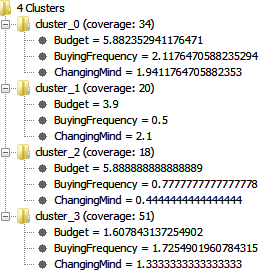
**Output** : The input data labeled with the cluster they are contained in.

**Configuration 1:**

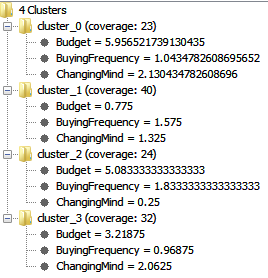
|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | BuyingFrequency, ChangingMind, Budget |
| Number Cluster | 4 |
| Max number iteration | 99 |

**Results:**

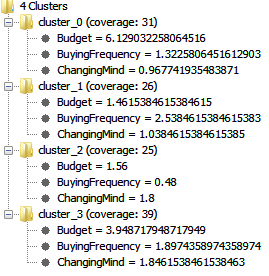
**Spain**



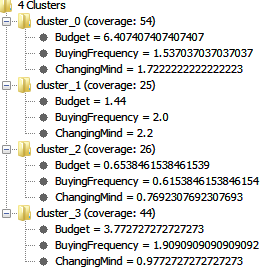
**France**



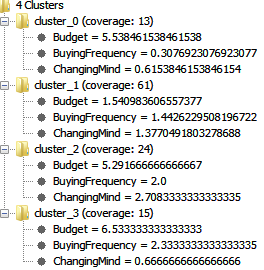
**Germany**



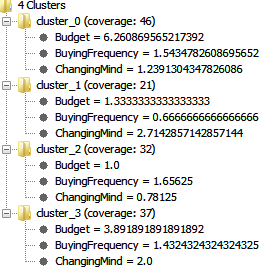
**Italy**



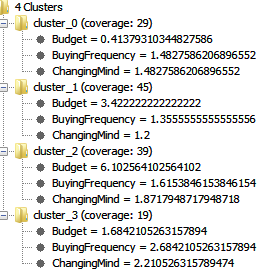
**Russia**



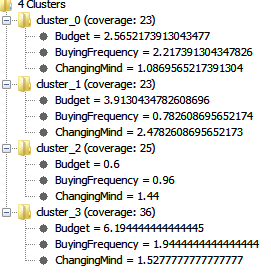
**Turkey**



**Poland**



**England**



We can see different clusters depending on the country.

We put together all the data in a table.

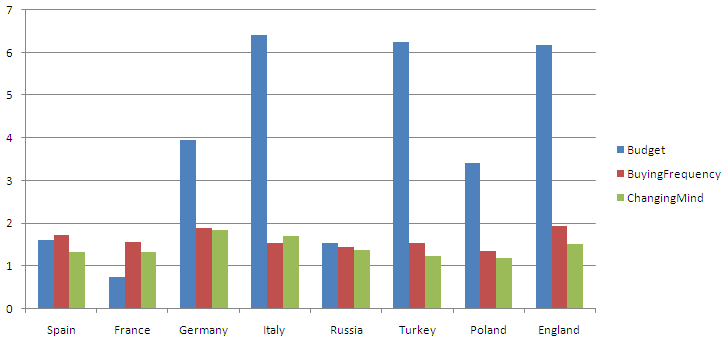
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Country | Cluster | Coverage | Coverage (%) | Budget | BuyingFrequency | ChangingMind |
|  |  |  |  |  |  |  |
| Spain | 0 | 34 | 28 | 5,88 | 2,11 | 1,94 |
| 1 | 20 | 16 | 3,9 | 0,5 | 2,1 |
| 2 | 18 | 15 | 5,89 | 0,78 | 0,44 |
| 3 | 51 | 41 | 1,61 | 1,73 | 1,33 |
|  |  |  |  |  |  |  |
| France | 0 | 23 | 19 | 5,95 | 1,04 | 2,13 |
| 1 | 40 | 34 | 0,75 | 1,58 | 1,33 |
| 2 | 24 | 20 | 5,08 | 1,83 | 0,25 |
| 3 | 32 | 27 | 3,22 | 0,97 | 2,06 |
|  |  |  |  |  |  |  |
| Germany | 0 | 31 | 26 | 6,13 | 1,32 | 0,97 |
| 1 | 26 | 21 | 1,46 | 2,54 | 1,04 |
| 2 | 25 | 21 | 1,56 | 0,48 | 1,8 |
| 3 | 39 | 32 | 3,95 | 1,9 | 1,84 |
|  |  |  |  |  |  |  |
| Italy | 0 | 54 | 36 | 6,41 | 1,54 | 1,72 |
| 1 | 25 | 17 | 1,44 | 2 | 2,2 |
| 2 | 26 | 17 | 0,65 | 0,62 | 0,77 |
| 3 | 44 | 30 | 3,77 | 1,91 | 0,98 |
|  |  |  |  |  |  |  |
| Russia | 0 | 13 | 12 | 5,54 | 0,31 | 0,62 |
| 1 | 61 | 54 | 1,54 | 1,44 | 1,38 |
| 2 | 24 | 21 | 5,29 | 2 | 2,71 |
| 3 | 15 | 13 | 6,53 | 2,33 | 0,67 |
|  |  |  |  |  |  |  |
| Turkey | 0 | 46 | 34 | 6,26 | 1,54 | 1,24 |
| 1 | 21 | 15 | 1,33 | 0,67 | 2,71 |
| 2 | 32 | 24 | 1 | 1,66 | 0,78 |
| 3 | 37 | 27 | 3,89 | 1,43 | 2 |
|  |  |  |  |  |  |  |
| Poland | 0 | 29 | 22 | 0,41 | 1,48 | 1,48 |
| 1 | 45 | 34 | 3,42 | 1,36 | 1,2 |
| 2 | 39 | 30 | 6,1 | 1,62 | 1,87 |
| 3 | 19 | 14 | 1,68 | 2,68 | 2,21 |
|  |  |  |  |  |  |  |
| England | 0 | 23 | 21 | 2,57 | 2,22 | 1,09 |
| 1 | 23 | 21 | 3,91 | 0,78 | 2,48 |
| 2 | 25 | 23 | 0,6 | 0,96 | 1,44 |
| 3 | 36 | 34 | 6,19 | 1,94 | 1,53 |

We can see some similarities: for the clusters with the highest budget, in France, Spain, Poland, the changingMind is also high. But in Russia for the cluster with highest budget, the changingMind is low.

Inside some countries, attributes don’t differ that much, for example in France, for all the clusters, the frequency is never very low. In Russia for example we can see that there is more difference, the frequency is very low for clusters 0 but high for cluster 3.

People who buy the less in Germany have medium changingMind and low budget, whereas in Spain they have high changingMind and high budget. In Turkey they have a high changingMind but a very low budget.

This graphic show for each country the cluster with the highest coverage of the population. With this we can study the behavior of the bigest part of the buyers depending on the country.

Obviously the majority of buyers in Italy, Turkey and England have a high budget, whereas in Germany and Poland the budget is medium, and the lowest is reached with France, Spain and Russia. For these population, we can consider to offer cheaper products.

The buyingFrequency is medium for the majority for most of the country. England has the highest, with a high budget, people buy often and expensive products, with a medium changingMind, try to improve the services on website, advices, and highlight expensive quality product in this area would increase profits.

In Germany, the second with the highest buying frequency, and a medium budget, buyers change their minds very often, so we can increase the quality of advices, help, and comparison between products on the website.

**Evaluation**:

* Data Mining evaluation: the prediction isn’t good by country.
* Business Success evaluation: we can see different clusters of behavior inside each country, we will be able to compare them and know if we have the same clusters or not in all the countries.

**Goal 2b,ii : behavior of shoppers, country depending, for shops**

**Data set** : as specified in data preprocess, goal2abii.csv

**Success Criteria:** we want to have an idea of how the behaviors of the website buyers change depending on the country.

**Techniques:**

* Classification : predicts value of a given attribute (the classification of an example)
  + Naive Bayes
* Clustering
  + K-means

Classification

Naive Bayes

With the algorithm, we try to see if with some attributes we can predict the country. That would means buyers from a country would have a specific behavior.

**Parameters**:

* Classification Column : The class value column.
* Skip missing values (incl. class column) : The node ignores missing values in the model if this option is ticked. If it's not ticked the node treats the missing values as a normal value and considers them during the class probability calculation.
* Maximum number of unique nominal values per attribute : All nominal columns with more unique values than the defined number will be skipped during learning. If the column contains missing values and the 'Skip missing values' option is not skipped the missing value counts as one value!

**Input**: Training data

**Output**: Learned naive Bayes model

**Configuration 1:**

96% of the data for learning, 4% for predicting.

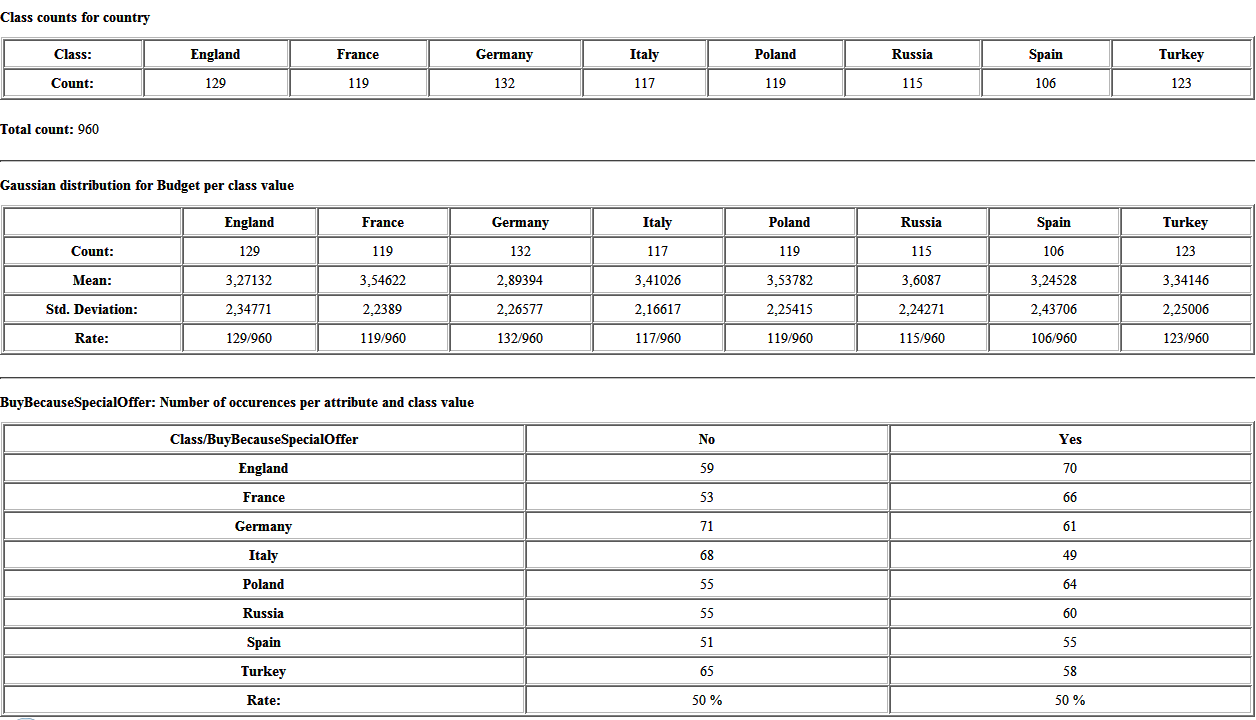
|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | all |
| Classification Column | country |
| Skip missing values | no |
| Maximum number of unique nominal values per attribute | 20 (default) |

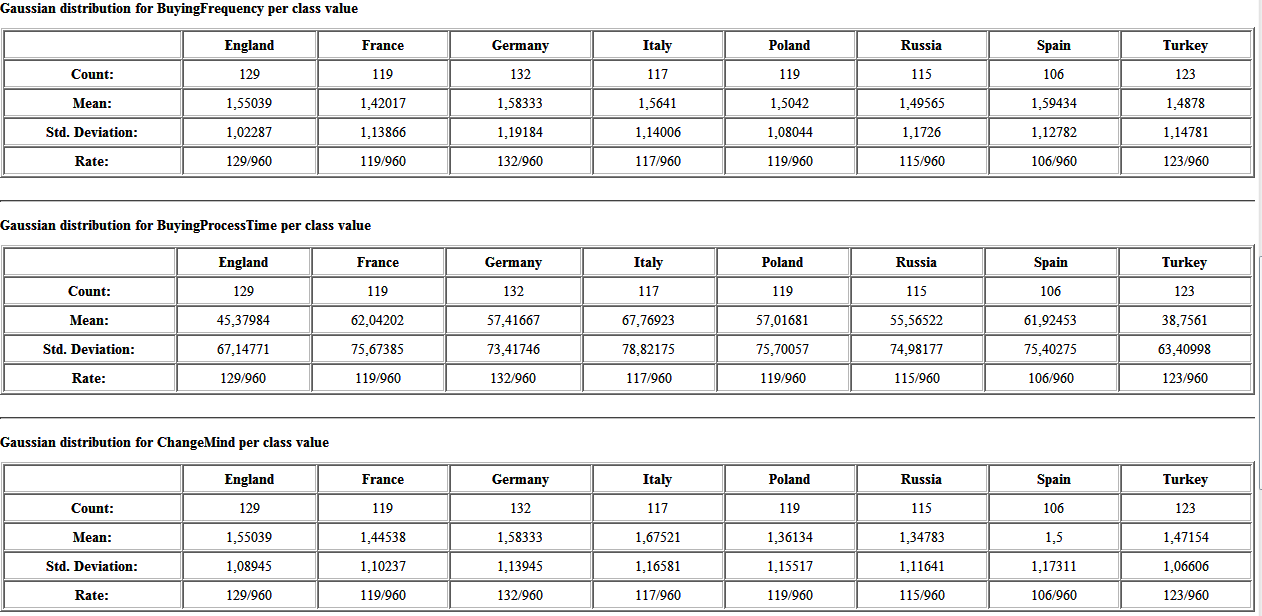
**Results**:

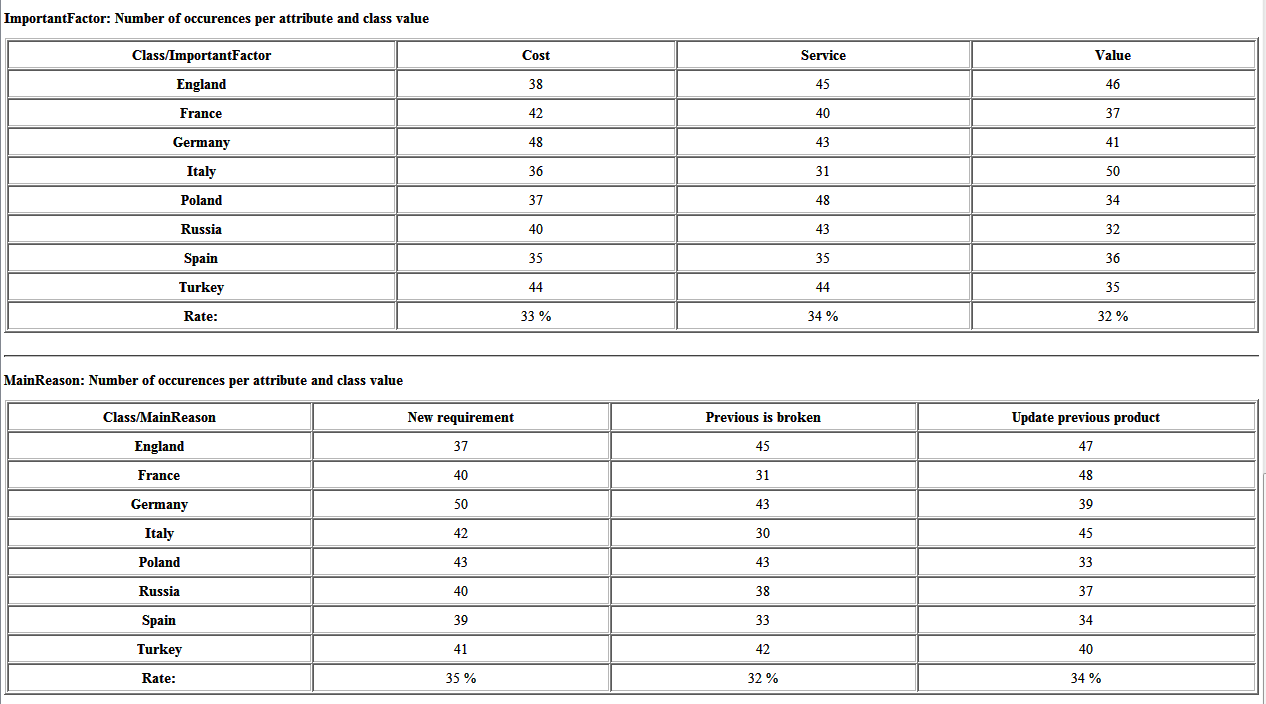
False: 32/ True: 8

Results aren’t good, we can’t find attributes that define a whole country shopping behavior.

We can still use the Naïve Bayes Learner results to have an idea of the difference between attributes depending on the country.







We can see that for most of the attributes, the mean is quite the same and the deviation also, or they aren’t that representative.

We are going to use now the K-means clustering, to know if inside each country, we can have different representative behaviors, and see if we find the same clusters in the different countries.

AS the results were better with the 2nd configuration, we will use the budget, frequency and chinging mind.

Clustering

K-means

**Parameters** :

* number of clusters : The number of clusters (cluster centers) to be created.
* max number of iterations : The number of iterations after which the algorithm terminates, independent of the accuracy improvement of the cluster centers.

**Input** : Input to clustering. All numerical values and only these are considered for clustering.

**Output** : The input data labeled with the cluster they are contained in.

**Configuration 1:**

|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | BuyingFrequency, ChangingMind, Budget, Process Time, country |
| Number Cluster | 4 |
| Max number iteration | 99 |

**Results :**

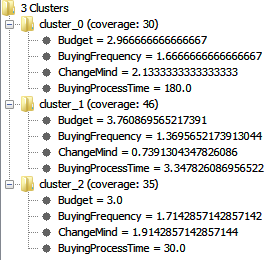
With 4 clusters, the 4th doesn’t cover any case. We will choose 3.

**Configuration 1:**

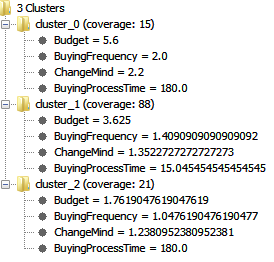
|  |  |
| --- | --- |
| Parameter name | Value |
| Attributes | BuyingFrequency, ChangingMind, Budget, Process Time, country |
| Number Cluster | 3 |
| Max number iteration | 99 |

**Results:**

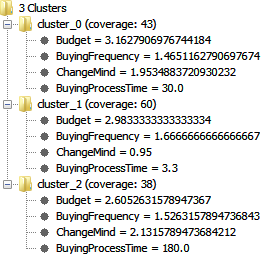
**Spain**



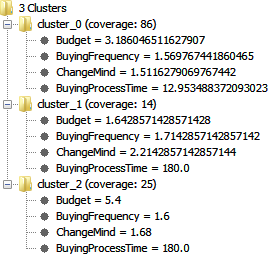
**France**



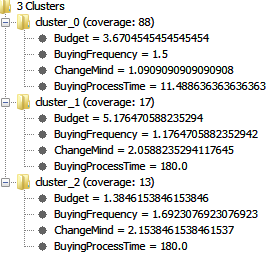
**Germany**



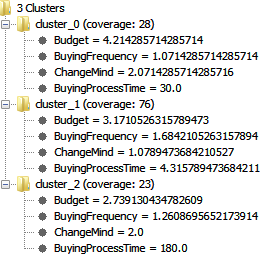
**Italy**



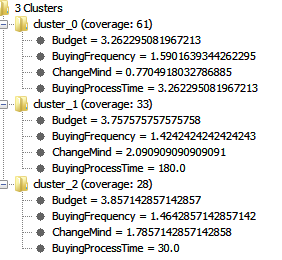
**Russia**



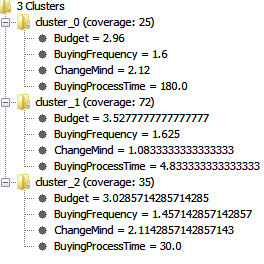
**Turkey**



**Poland**



**England**



We can see different clusters depending on the country.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Country | Cluster | Coverage | Coverage (%) | Budget | BuyingFrequency | ChangingMind | BuyingProcess  Time |
|  |  |  |  |  |  |  |  |
| Spain | 0 | 30 | 27 | 2,97 | 1,67 | 2,13 | 180 |
| 1 | 46 | 41 | 3,76 | 1,37 | 0,74 | 3,35 |
| 2 | 35 | 32 | 3 | 1,71 | 1,91 | 30 |
|  |  |  |  |  |  |  |  |
| France | 0 | 15 | 12 | 5,6 | 2 | 2,2 | 180 |
| 1 | 88 | 71 | 3,63 | 1,41 | 1,35 | 15,05 |
| 2 | 21 | 17 | 1,76 | 1,05 | 1,24 | 180 |
|  |  |  |  |  |  |  |  |
| Germany | 0 | 43 | 30 | 3,16 | 1,47 | 1,95 | 30 |
| 1 | 60 | 43 | 2,98 | 1,67 | 0,95 | 3,3 |
| 2 | 38 | 27 | 2,61 | 1,53 | 2,13 | 180 |
|  |  |  |  |  |  |  |  |
| Italy | 0 | 86 | 69 | 3,19 | 1,57 | 1,51 | 12,96 |
| 1 | 14 | 11 | 1,64 | 1,71 | 2,21 | 180 |
| 2 | 25 | 20 | 5,4 | 1,6 | 1,68 | 180 |
|  |  |  |  |  |  |  |  |
| Russia | 0 | 13 | 13 | 3,67 | 1,5 | 1,09 | 11,49 |
| 1 | 61 | 62 | 5,18 | 1,18 | 2,06 | 180 |
| 2 | 24 | 24 | 1,38 | 1,69 | 2,15 | 180 |
|  |  |  |  |  |  |  |  |
| Turkey | 0 | 28 | 22 | 4,21 | 1,07 | 2,07 | 30 |
| 1 | 76 | 60 | 3,17 | 1,68 | 1,08 | 4,32 |
| 2 | 23 | 18 | 2,74 | 1,26 | 2 | 180 |
|  |  |  |  |  |  |  |  |
| Poland | 0 | 61 | 50 | 3,26 | 1,59 | 0,77 | 3,26 |
| 1 | 33 | 27 | 3,76 | 1,42 | 2,09 | 180 |
| 2 | 28 | 23 | 3,86 | 1,46 | 1,79 | 30 |
|  |  |  |  |  |  |  |  |
| England | 0 | 25 | 19 | 2,96 | 1,6 | 2,12 | 180 |
| 1 | 72 | 55 | 3,53 | 1,63 | 1,08 | 4,83 |
| 2 | 35 | 27 | 3,03 | 1,46 | 2,11 | 30 |

For some countries, the budget is very disparate, with strong differences between clusters: Russia, Italy, France. For others there is less differences: England, Poland, Germany, Spain.

We can see that for all of the countries except France, the buyingFrequency isn’t very different depending on the cluster. For France, the 3 clusters have different frequencies, proportionally increasing with the budget.

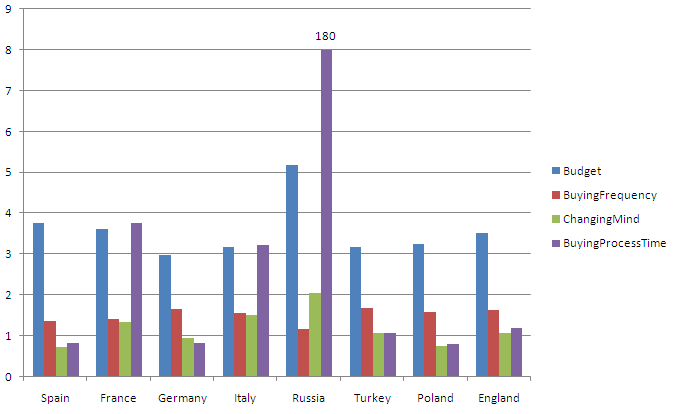
People who change their mind the less in Spain have a medium budget (but the highest inside the country), same for Germany, Russia, Turkey, Poland, England.

In all countries, people who change their mind the less have a buyingProcessTime the lowest.

The country with a cluster with the lowest budget is Russia, we can think about offer cheap products in these shops, to satisfy this part of the population.

This graphic show for each country the cluster with the highest coverage of the population. With this we can study the behavior of the bigest part of the buyers depending on the country.

(The buyingProcessTime is divided by 4 to fit in, except for the turkey which is too high, we put it manually to 8.)



For most of the population in each country except of Russia, the buyers have a medium budget. In Russia it’s higher.

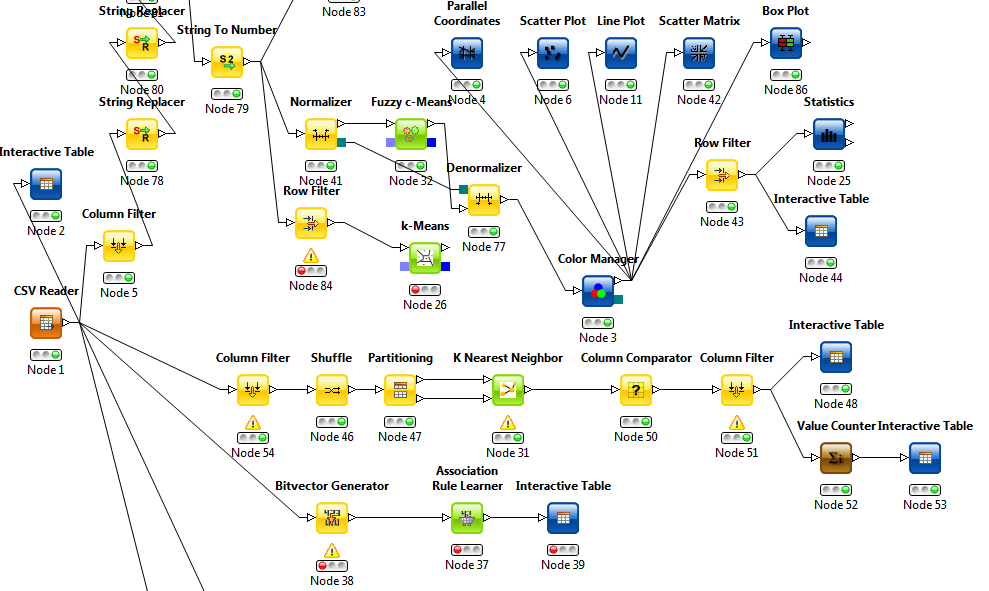
Country with a low buyingProcessTime : Spain, Germany, Poland, have quite the same buyingFrequency (medium) and changingMind (small).

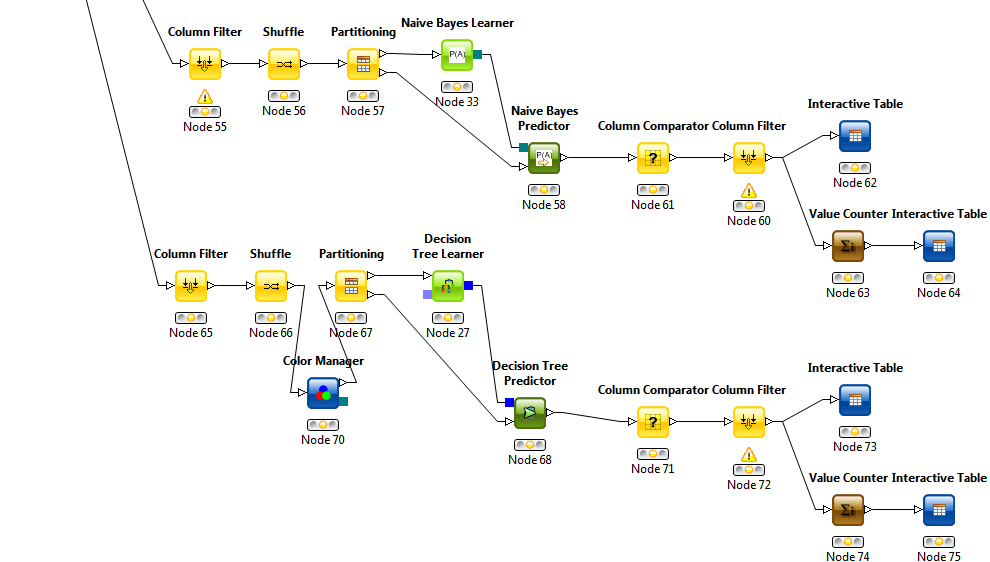
In all countries except Russia, the frequency is higher than the changingMind (if we compare the numbers). In Russia, it’s the opposite, most of buyers change often their mind, but buy less than in the other countries. We could consider establish a better guiding in these shops, to make them buy more often and be sure about their choices.

**Evaluation**:

* Data Mining evaluation: the prediction isn’t good by country.
* Business Success evaluation: we can see different clusters of behavior inside each country, we will be able to compare them and know if we have the same clusters or not in all the countries.

# ANNEXE Goal 2 : Knime workspace





# Third Goal: Identification of important parameters

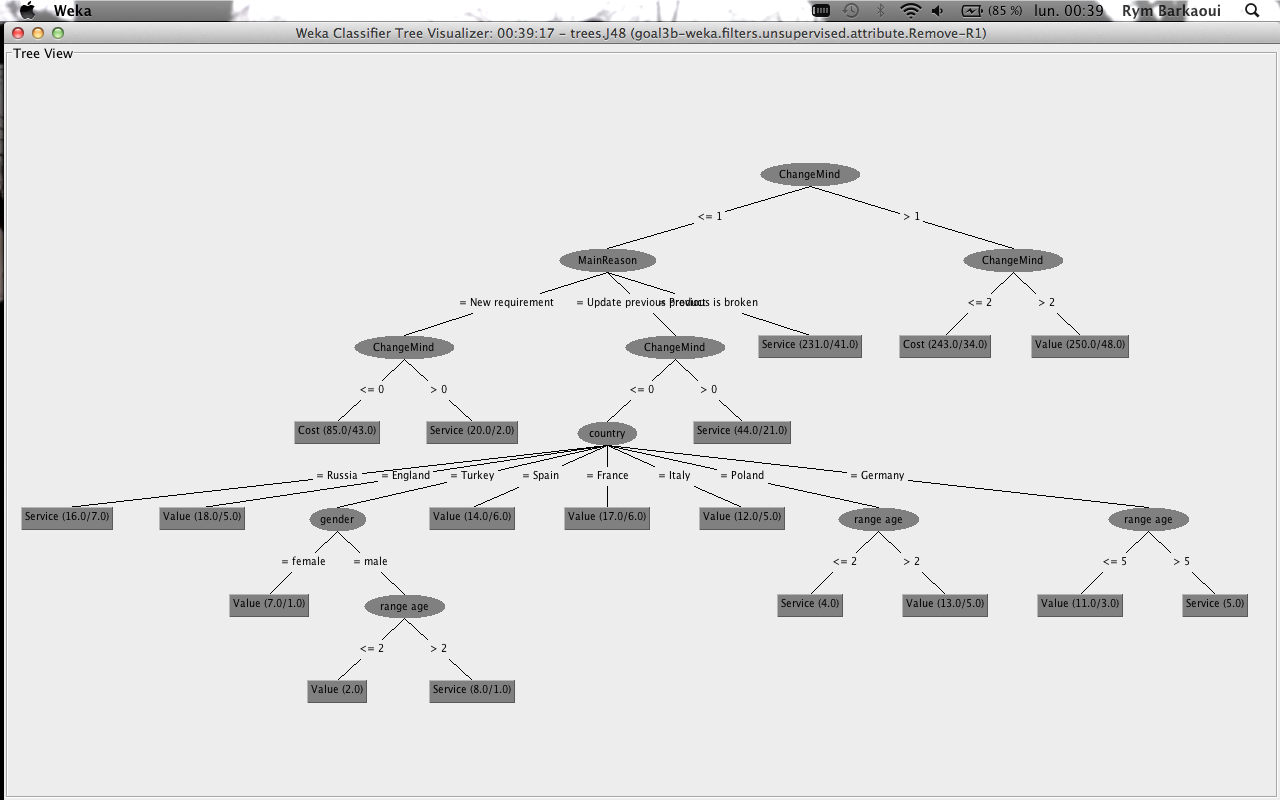
Success criteria:

The third data-mining goal is to define some important factors for the customer. We will know depending on the profile what is the most important for them.

For example we could be able to say that a 28 years old divorced woman considers that the price is very important.

In order to see the profile of the customer depending on the factor, the REPTree algorithm will be used on 2 different dataset:

1. Buyers in shops:



Main profile: most important factor is Cost:

|  |  |
| --- | --- |
| Changing mind | Main reason |
| Frequently/always | all |
| Almost never | New requirement |

Main profile: most important factor is Value:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Changing mind | Main reason | Country | Gender | Age |
| Never/Almost never | All | All | All | All |
| Never | Update previous | France/Italy/Spain/England | All | All |
| Never | Update previous | Turkey | Female | All |
| Never | Update previous | Turkey | Male | <25 |
| Never | Update previous | Poland | All | >25 |
| Never | Update previous | Germany | All | <55 |

Main profile: most important factor is Service:

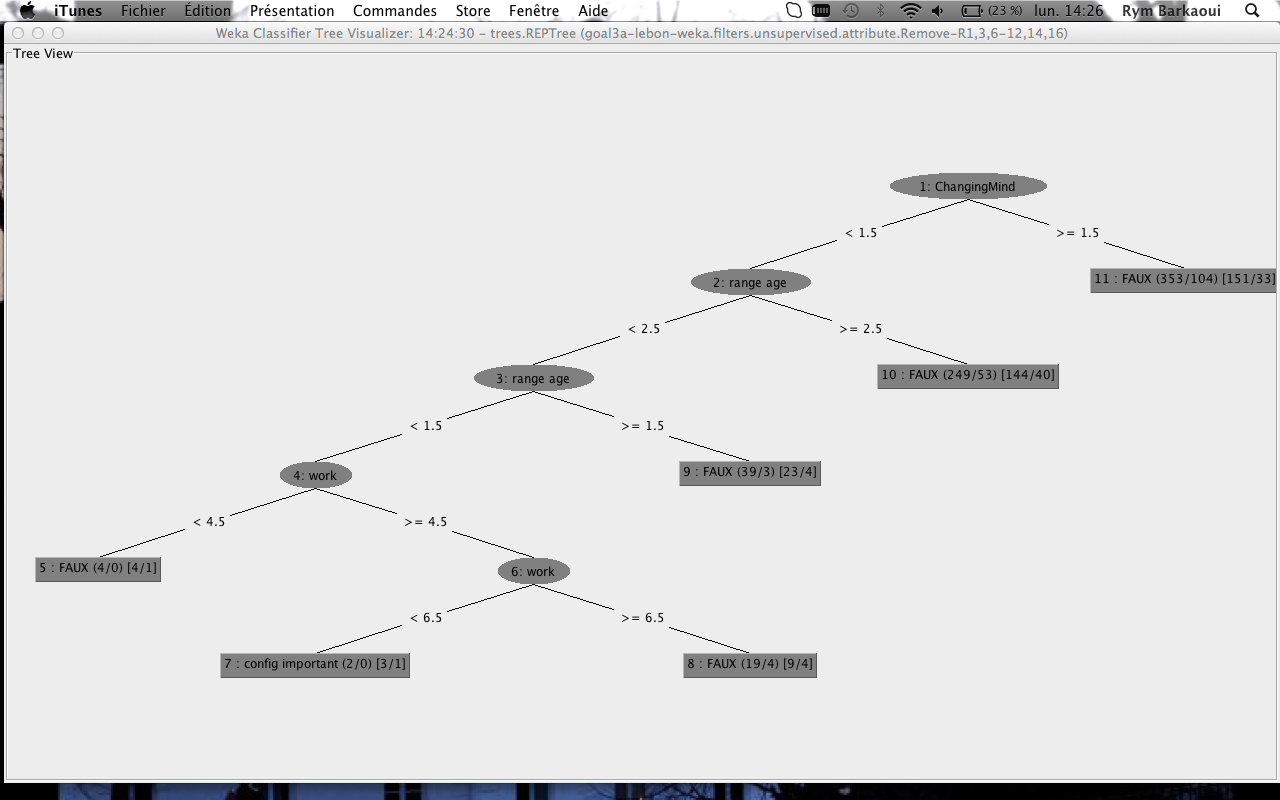
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Changing mind | Main reason | Country | Gender | Age |
| Never/Almost never | Previous is broken | All | All | All |
| Almost Never | New requirement | All | All | All |
| Almost Never | Update previous | All | All | All |
| Never | Update previous | Russia | All | All |
| Never | Update previous | Turkey | Male | >25 |
| Never | Update previous | Poland | All | <25 |
| Never | Update previous | Germany | All | >55 |

We can presume that somebody who changes his mind very often is mainly interested on the price whereas people who keep thinking on the same product give more importance to the value or the service. In most countries, Value is more important than the two other parameters.

1. Buyers in Websites:

# Technical configuration importance:

For some customers the technical configuration of the product is more important than the price, we can notice it if they look for information about the software and hardware. For this category, the main reason must be their work, their age that is why we have chosen to use only these attributes.

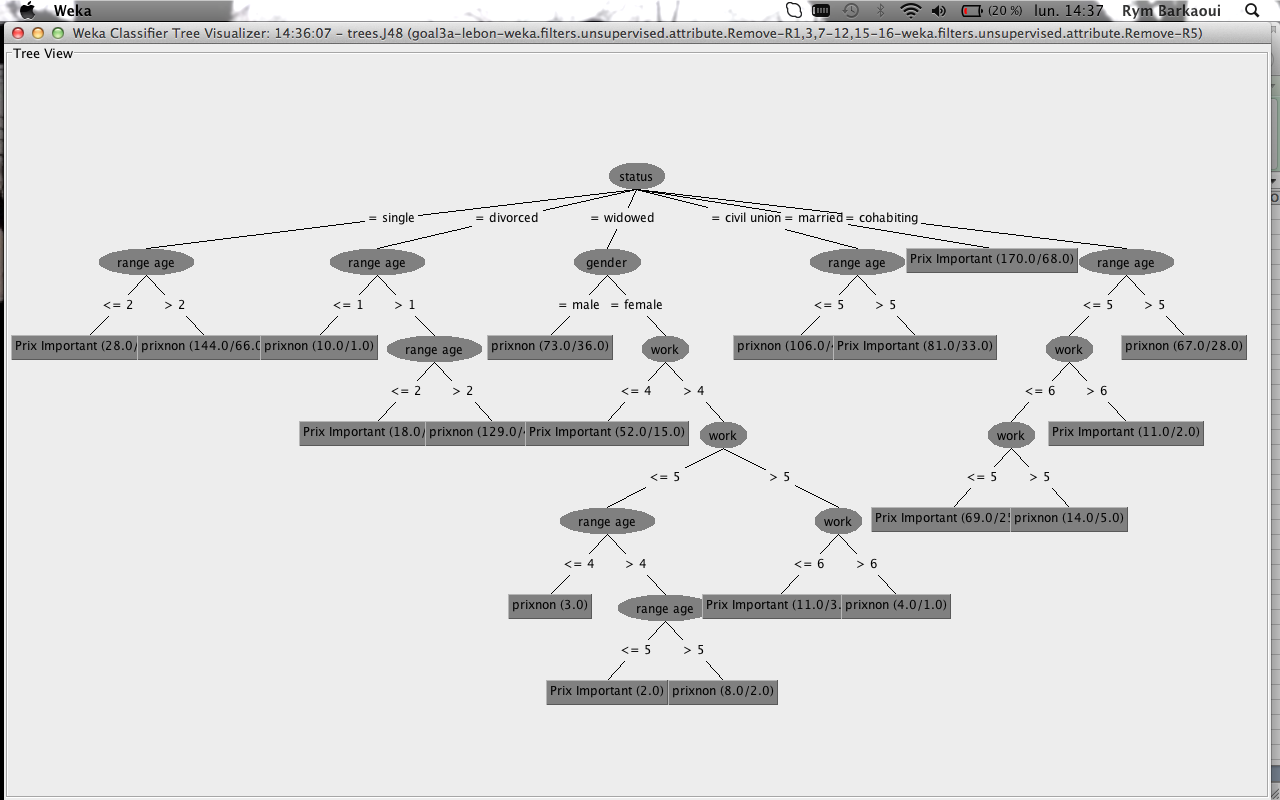


By this way, we can figure out that the profile of these customers is people who

* Change their mind very rarely,
* Less than 35 years old,
* Employees, workers or students.

# Price importance:

For some customers, the most important factor is the price. We consider for this case the age, the status, the work situation and the gender to know whether a customer would buy a product more expensive or not. Indeed, depending on the work situation if a customer is a student most of the time parents will pay that is why they will focus more on the technical configuration than the price whereas a married customer will give importance to the price (we assume that he has a family to care about…)

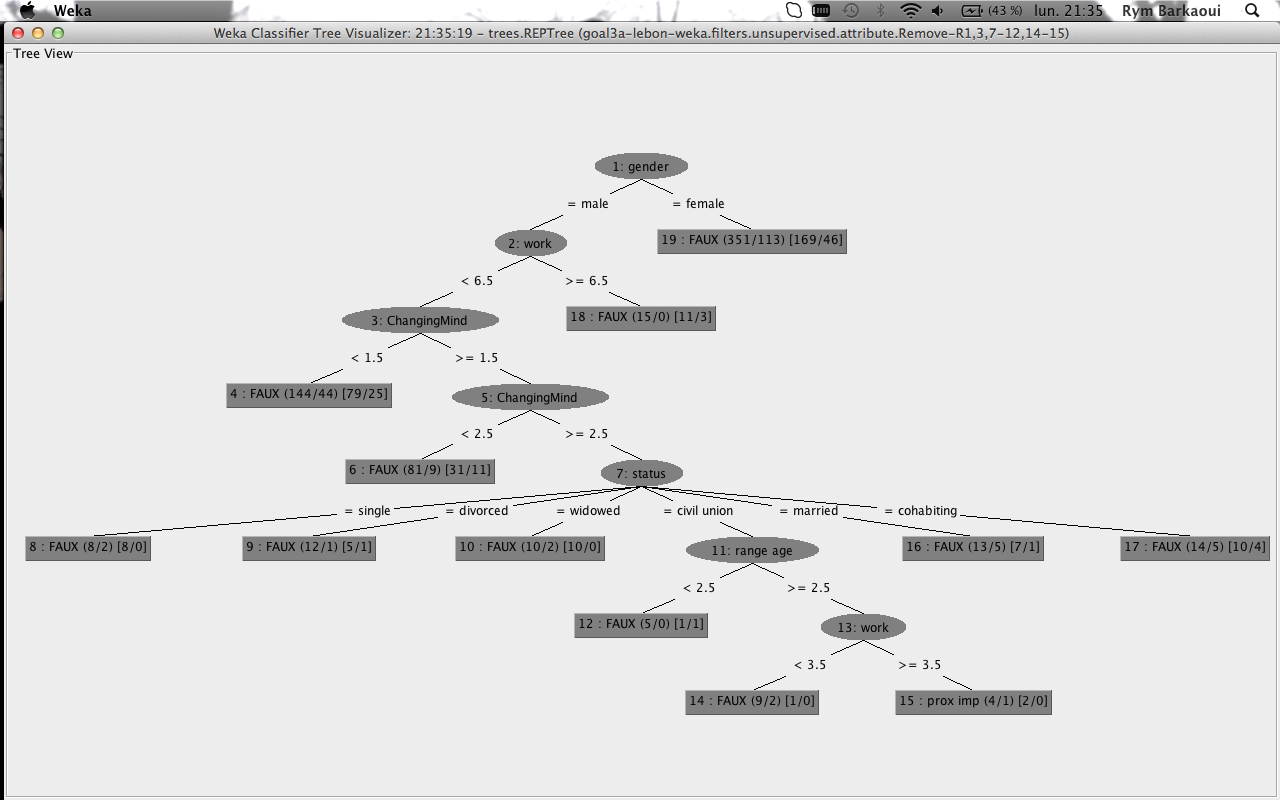


From that tree we can establish the profile of all the customers who care the most about the price:

|  |  |  |  |
| --- | --- | --- | --- |
| Status | Range Age | Work | Gender |
| Single | <35 | All | All |
| Divorced | <35 | All | All |
| Widowed | All | 1. Farmer  2. Artisan, shopper, CEO  3. White collar and grey matter  4. Intermediary professions | Female |
| Widowed | <55 | 5. Employees  6. Workers  7. Students | Female |
| Widowed | All | 5. Employees  6. Workers | Female |
| Civil union | <55 | All | All |
| Married | All | All | All |
| Cohabiting | <55 | Student | All |
| Cohabiting | <55 | 1. Farmer  2. Artisan, shopper, CEO  3. White collar and grey matter  4. Intermediary professions  5. Employees | All |

# Proximity shops importance:

We can notice that for some customers, the distance between their home/office and the shop is very important and can make them choose a product over another. That is why we consider that the age is not a important attribute whereas the gender, the age, the status and the work are highly interesting in this case.



We figure out that there is only one profile for who the proximity of the shop is more important. Indeed this category profile is:

* male
* workers or employees or doing intermediary professions,
* changing their mind very frequently,
* living in a civil union,
* more than 40 years old.