Product recommendation based on shared customer's behaviour

Fátima Rodrigues, Bruno Ferreira

Abstract

Today consumers are exposed to an increasing variety of products and information never seen before. This leads to an increasing diversity of consumers' demand, turning into a challenge for a retail store to provide the right products accordingly to customer preferences. Recommender systems are a tool to cope with this challenge, through product recommendation it is possible to fulfill customers' needs and expectations, helping maintaining loyal customers while attracting new customers. However the huge size of transactional databases typical of retail business reduces the efficiency and quality of recommendations. In this paper a hybrid recommendation system that combines content-based, collaborative filtering and data mining techniques is proposed to surpass these difficulties. The recommendation algorithm starts to obtain similar groups of customers using customer lifetime value. Next an association rule mining approach based on similar shopping baskets of customers of the same cluster, in a specific time period is implemented in order to provide more assertive and personalized customer product recommendations. The algorithm was tested with data from a chain of perfumeries. The experimental results show that the proposed algorithm when compared with a base recommendation (made solely on past purchases of customers) can increase the value of the sales without losing recommendation accuracy.

1. Introduction

Consumers are permanently involved in multi-category decision-making. In a retail context, such multi-category decision processes result in the shopping-baskets that comprise the set of items that the consumers purchase on one visit to the store. Both on-line and off-line retailers are traditionally interested in understanding the composition of their customers' market baskets, since valuable insights for designing micro-marketing and/or targeted cross-selling programs can be derived[1]. Recommender systems are technologies that assist businesses to implement such strategies. Schafer[2] presented a detailed taxonomy of recommender systems in e-commerce, and determined how they can increase the probability of cross-selling; establish customer loyalty; and fulfill customer needs by discovering products in which they may be interested. The need to reduce information overload by retrieving the most relevant information and services from a huge amount of data, and also, the development of recommendation approaches and techniques, has determined a rapid proliferation of recommender systems grouped into eight main domains[3]: e-government, e-business, e-commerce/e-shopping, e-library, e-learning, e-tourism, e-resource services and e-group activities. Recommender systems are usually classified based on how recommendations are made[4]. A content-based recommender system is based on similar items to those a given user has liked in the past[5,6]. A collaborative filtering makes recommendations based on items owned by users whose taste is similar to those of the given user[7,8]. Combining content-based and collaborative recommendations originate hybrid approaches[9], which are commonly used, considering that both types of recommendations may complement each other.

In the recommender system here described two sources of information are used. First, it is used clustering to obtain groups of customers with similar interests based on prior purchase patterns. Second, rule association mining is performed on baskets of the same cluster, in order to derive relationships between products. Since these relationships are based on purchases of similar customers that have also purchased, in the same time period, at least one same product, it is expected to identify additional product relationships that are not captured using only the past baskets of the customer. In summary, this recommender system uses collaborative filtering combined with the ideas from content-based filtering.

There are usually quite a lot of products to be considered in a recommender system. It would be very inefficient if every product needs to be considered before making recommendations. Dimensionality reduction techniques have been incorporated to produce quickly quality recommendations for large-scale problems[10,11]. However these systems have some disadvantages, for example, require extra attributes about users or products to group the users into clusters and require the number of clusters be given in advance, which is a big burden on the user. With the proposed approach, clustering is done totally based only on derived attributes about products purchased by costumers, without the necessity of collecting extra attributes about customers and products. Besides, when selecting the baskets for recommending products, we consider only baskets of clients of the same cluster, bought in a specific time period, resulting in a much greater reduction of the number of products to consider. Due to dimensionality reduction on the number of products, the processing time for making recommendations by our approach is much reduced. Experimental results show that the proposed recommender system can enhance the recommendations with a good performance without compromising the recommendation quality.

The remainder of this paper is organized as follows: in section 2 a brief explanation of concepts and algorithms used to implement the recommender system is made. In the following section the hybrid recommendation algorithm is explained. In the next section details of the recommender system are provided. Section 6 presents the experimental results and in last section conclusions and suggestions for future work are disclosed.

2. Background

2.1. Customer Lifetime Value Analysis and RFM Evaluation

Customer lifetime value is typically used to identify profitable customers and to develop strategies to target customers. The RFM (recency, frequency and monetary) model is the most widely used model to characterize customers due to its simplicity and good predictive capabilities. "Recency" represents the time since the last purchase, a lower value corresponding to a higher probability of the customer making a repeat purchase. "Frequency" denotes the number of purchases within a specified time period; higher frequency indicates higher loyalty. "Monetary" means the amount of money spent in this specified time period, a higher value indicating a customer that the company should focus[9]. In fact, these three variables characterize the customer in terms of his behavior and can be used as the segmenting variables by observing customers' attitudes toward the product, brand, benefit, or even loyalty.

2.2. Cluster Analysis

Cluster analysis groups data objects based only on information in data that describes the objects and their relationships. The goal is that the objects within a group be similar to one another and different from the objects in the others groups. The grouping of objects is based on a distance or similarity function, so that clusters can be formed from objects with a high similarity to each other. Several clustering algorithms have been developed, here we will use a partitional algorithm to group customers with similar lifetime value. A partition algorithm initially defines K seed points x_k, one for each cluster, and iteratively update these points to optimize some objective-function. At each iteration, each object x_i is assigned to the most similar seed point. When the attributes are real values, the seed points are referred as centroids if they represent the arithmetic mean of each cluster, or as medoids if the seed points must be objects of the data set (the closest to the clusters' centers). When the features are categorical, the seed points are designated as modes. The most known partitional clustering algorithm is the K-means[10] that (locally) minimizes the sum of squared errors between the cluster centroids and the objects in the corresponding clusters:

$$J_{KM} = \sum_{k=1}^{K} \sum_{x_i \in C_k} \|x_i - \bar{x}_k\|^2 \tag{1}$$

where $\|.\|$ represents the l2-norm. K-means takes as input the number of clusters K, and starts by randomly defining K centroids,$\{\bar{x}_1, \dots, \bar{x}_k\}$. Then, it iterates between two steps: assigning each object to the cluster represented by the closest centroid; and updating the clusters' centroids as the means of each cluster:

$$\bar{x}_k = \frac{\sum_{x_i \in C_k} x_i}{|C_k|} \quad \forall_{k \in \{1,..,K\}} \tag{2}$$

The process repeats until no point changes clusters, or equivalently, until the centroids remain the same.

2.3. Association Rule Mining

Association rule mining is used to identify relationships among a set of items in a database. These relationships are based on the co-occurrence of the data items. Therefore, the main purpose of extracting association rules is to find out synchronous relationships by analyzing the random data and to use this data as reference during decision-making. The association rule is defined as follows[11]:

Let D be a database composed by a collection of items A={a_1, a_2, ..., a_m} and a group of transactions T={t_1, t_2, ..., t_m}, where each transaction $t_i \in$ T is composed by a set of items such that each item set is a non-empty sub-item set of A, $t_i \subseteq$ A. Each item set X holds in T with support Sup, if Sup% of the transactions in T contain X. The support is a measure that evaluates the statistical importance of X in the database D.

The association rule is an implication in the form X→Y in that X,Y ⊆ A and X∩Y=∅. The rule means that if X is purchased, Y can be bought at the same time. The rule X→Y holds in T with confidence Conf, if Conf% of transactions in T that support X also support Y; i.e. Conf (X→Y) = Sup (X∪Y,D) / Sup(X,D). A high confidence ensures the predictability of the rule. The confidence measure doesn't detect independence among the items in a rule, and rules with no correlated items can have high confidence value. This happens because the confidence measures ignores the support of the item set in the rule consequent. One way to address this problem is by applying the lift metric, Lift(X→Y) = Conf(X→Y) / Sup(Y,D), which measures how many times X and Y occur together more than expected, if they were statistically independent.

In this work we will use the Apriori algorithm[11]. Even though Apriori was the first algorithm developed to extract association rules, it is still one of the most widely used algorithms. Ease of implementation, simplicity, efficiency, and empirical success are the main reasons for its popularity.

3. Hybrid Recommendation Algorithm

The cluster K to which a customer C belongs is first identified. Then, the set of all products previously purchased by customer C in a specific time period is selected - PS_{TC}. In order to avoid an explosive rule generation and get customized associations, previous to rule generation it is selected all the customers' transactions from cluster K, in the same time period forming the set of cluster transactions for that period - TS_{TK}. From TS_{TK} are eliminated the baskets with only one item, as it is not possible to generate rules from them.

With this specific TS_{TK} it is calculated the range of support values of its items, and it is selected the minimum value of the support range to be used as minimum support (Sup_{min}) in the Apriori algorithm. The goal of this Sup_{min} is to obtain all possible rules from this specific set of cluster transactions. As confidence is a measure of the rule's strength, in order to avoid items purchased together occasionally, it is defined as minimum confidence, $Conf_{min}$=100%. The Apriori association rule mining algorithm is applied to find the recommendation rules RS_{CK} relate to this subset of transactions TS_{TK}. For each product P from PS_{TC} are selected all the rules from RS_{CK} that contains in his left hand side the product P. To get the items more relate with those purchased by the customer, the RS_{CK} is sorted by the lift measure, and the top-N rules with highest lift are added to the recommendation rule set for customer C, RRS_{TC}. This specific selection, based on items a costumer has bought in the past, and on items other similar costumers of the same cluster have also bought, allows making a hybrid recommendation.

The set of candidate products for recommendation to customer C, RPS_{TC} is the set of all products of the RRS_{TC} minus previously bought products, PS_{TC} (RRS_{TC} - PS_{TC}). Previously bought products are excluded from the recommend list since the recommender is meant to broaden each customer's purchase products.

All candidate products from RPS_{TC} are sorted and ranked according to its support. The N highest ranked candidate products (top sellers) are selected as the top-N recommended products.

4. Recommender System

For the implementation of the proposed methodology, a recommender system was developed for a chain of perfumeries. The goal of the recommender system is to provide periodically relevant personalized item recommendations to loyalty customers, in order to increase the customer's interest in the stores products and consequently increase the sales volume. This company has over 30 years of existence; sells perfumery, cosmetics, make-up and body care products and early bet on customer retention through a loyalty card. The company has already 25000 customers and has more than 11000 items available for sale. The company's management is made through an Enterprise Resource Planning (ERP) system that the company uses to manage all its activities, processes and workflow. The ERP presents a very significant complexity with a database of over 300 tables, corresponding to 10GB of data for the 2012-14 period.

4.1. Data Staging Area

In order to centralize all information relating to customers into a single data source and provide quickly and accurately customer information to the recommender system, a specific data staging area was created. Due to the wide variety and heterogeneity of data in the ERP, collecting and cleaning transactional data are done through an Extraction Transformation Loading (ETL) process. As the name implies this is a three phase process involving: the extraction of data from the ERP system; data transformation such as, the treatment of inconsistencies, mapping data to a single naming convention, handling missing data and errors, integrity faults, etc.; and finally, the loading of data into the staging area. The staging area is a database built on SQL Server not standardized with configurable location, consisting of three tables: customers, products and sales. The mechanisms and routines of the ETL process and the storage management of the staging area are conducted using the SQL language and a generic programming language for performing the management of all surrounding processes.

From the data collected with the ETL process, this study will be focused on the 2012-2014 period. The recommendations will be only to loyalty customers that is, customers who made purchases on the three years 2012-2014. The first two years, 2012 and 2013, will be used as training set, and the first semester of 2014 year as test set, that is, the sales of the two first years will be used to recommend products, that will be checked with the sales of the first semester of 2014 year. The reduced dataset contains 3245 loyalty customers. There aren't any known preferences of customers, only what they bought in the last two years.

4.2. Costumer Segmentation

The system starts to perform customer segmentation based only on customer lifetime value, built based on the R–F–M customer attributes concerning the 2012-2013 purchases period. This segmentation will give groups of customers with identical behaviour shopping in terms of when they buy, how often they buy and how much they buy.

Prior to any analysis that uses distance calculations it is necessary to normalize or to standardize the features to the same scale. Some features can dominate solely because they tend to have larger values than others. Doing normalization this problem is avoided. Data scaling depends on the data distribution. As can be seen in the boxplots of RFM attributes in Fig. 1, these attributes have too many outliers, denoted as circles or dots beyond the whiskers.

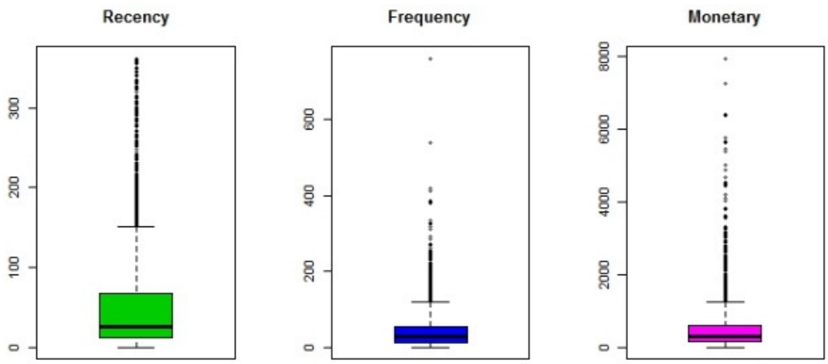

Fig. 1. Boxplot of RFM attributes.

The sigmoidal normalization is a nonlinear transformation that transforms the input data into the range -1 to 1, using a sigmoid function. It is an appropriate approach to capture the very large outlier values while mapping the input data into a range of values.

For the majority of clustering algorithms, including the k-means, it is necessary to specify the number of clusters k in advance. Various cluster evaluation measures can be used to approximately determine the correct or natural number of clusters. A given evaluation measure will work better on some datasets than others, because of that we compute two distinct indices: the total Within Sum of Squares (WSS) and the Calinski-Harabasz index[12] for values of k between 2:10. Fig. 2 shows a plot of the Calinski-Harabasz versus the number of clusters and also a plot of WSS versus the number of clusters. There is a distinct peak in the Calinski-Harabasz and a distinct knee in the WSS when the number of clusters is 3. So, for both indices the best value of k is 3, which means this dataset has 3 clusters of customers that have similar RFM behaviour.

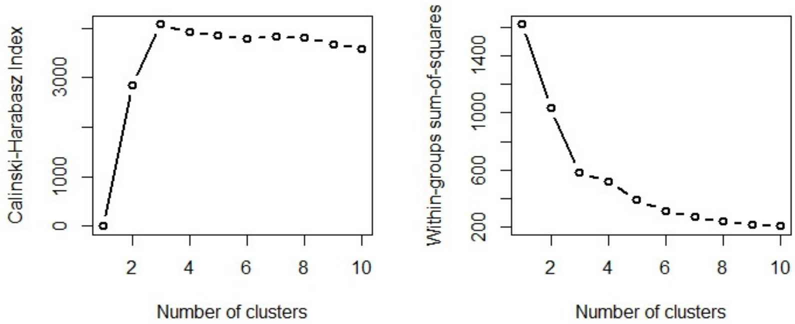

Fig. 2. WSS and Calinski-Harabasz versus number of clusters.

The k-means algorithm is not guaranteed to have a unique stopping point. K-means can be fairly unstable, in that the final clusters depend on the initial cluster centers. It is good practice to run k-means several times with different random starts, and then select the segmentation with the lowest total WSS. For that we run K-means with k=3, 25 random starts, and 100 maximum iterations per run. Table 1 describes quantitatively the three clusters obtained in terms of their average RFM values.

Table 1. Average RFM parameters for each cluster.

Cluster	Number of Customers	Recency	Frequency	Monetary
1	1663	33.51	29.42	251.63
2	842	24.38	96.51	970.99
3	740	181.99	20.33	199.45

To better analyse the clusters, we categorize the R, F, and M parameters in five categories (Very Low, Low, Medium, High, and Very High) accordingly to the quantile values of these parameters (table 2).

Table 2. RFM Categories.

	Very Low	Low	Medium	High	Very High
Recency	0-14	14-33	33-65	65-87	87-362
Frequency	4-17	17-31	31-45	45-56	56-699
Monetary	7-124	124-256	256-426	426-510	510-6165

Comparing the result of RFM parameters' values of each cluster (table 1) with categorical values (table 2), the category of the three parameters is identified to qualitatively characterize each cluster as shown in table 3.

Table 3. Clusters Characterization.

Cluster	Recency	Frequency	Monetary
1	Medium	Low	Low
2	Low	Very High	Very High
3	Very High	Low	Low

The segmentation performed using only the RFM features has discovered well-separated groups of customers with significantly different purchase attraction. Customers in cluster 1 with the pattern R (M) F (L) M (L) are customers who buy few and low value items but, because of their medium recency, they are more likely to make a repeat purchase. Cluster 2 includes high valuable and loyalty customers. They should be treated especially not to lose them. And finally, cluster 3 includes customers who very rarely visit the shops and made very few and cheap purchases. They are valueless customers that the retailer should consider whether or not to make product recommendations.

5. Personalized Product Recommendation

Recommendations should be carefully used, because its misuse, recommendations of no interest to the customer or excessive number of recommendations, could have an opposite desired effect - customer loss. Poor recommendations can cause two types of characteristic errors: false negatives, which are products that are not recommended, though the customer would like them, and false positives, which are products that are recommended, though the customer does not like them. In a recommender system, the most important errors to avoid are false positives, because these errors will lead to unhappy customers and thus they will be unlike to return to the store. Making recommendation only for customers who are likely to buy recommended products could be a solution to avoid the false positives of the poor recommendation. From the segmentation obtained it is clear distinct cluster consumption habits. So, recommendations must be specific to each one of the clusters, and due to the high number of customers in each cluster, specific target customers must be selected.

The selection of target customers for all clusters is made according to the regularity of customer's visits to the stores in the two previous years. The customers that are selected for recommendations in March month for instance are those that have made purchases in March in the last two years. Doing this it is respected the incursion pattern of the customer to the store. The recommendations will be made for periods of two months; this prevents recommending seasonal items at inappropriate times of the year. As the test set includes data for the first half of 2014, it will be evaluated three periods of recommendation for each cluster. Table 4 shows the number of target customers by cluster for each one of the three periods.

Table 4. Number of target customers by period.

Cluster\Period	Jan-Feb	Mar-Apr	May-Jun
1	460	504	555
2	589	607	631
3	68	68	88

The data considered in this study relative to the purchases of the 2012-2014 period consists in 76208 baskets, with 1373 distinct items. The company uses a three-level hierarchical item taxonomy that divides the 1373 items across 71 types and 4 categories: accessories, cosmetics, perfumery and toiletries.

Each cluster has associated a very sparse dataset with few items per basket. Table 5 presents a characterization of the clusters in terms of their baskets.

Table 5. Cluster baskets characterization.

Cluster	Number Baskets	Number Items	Type	Items/basket		
				Min	Mean	Max
1	32982	1365	70	1	1.77	15
2	36546	1371	71	1	2.46	20
3	6680	1246	70	1	1.86	16

The average item included in a single market basket is only 2.11 and the average item purchased by single customer is only 2.04. Baskets with only one item cannot be used to generate association rules therefore they will be removed.

In order to avoid an explosive rule generation and also get customized associations for each cluster, previous to rule generation it is selected the cluster transactions specific to the time period of recommendation. The Apriori association rule mining algorithm is applied to this specific-basket, with a Sup_{min} equal to the minimum support of the items of the basket, and with a $Conf_{min}$=100%. The goal is to find the maximum recommendation rules relate to this subset of transactions - RS_{CK}. Table 6 shows the characterization of the rule set for each cluster, by period of recommendation.

Table 6. Cluster Rule Set characterization by Period.

Cluster	Period	Items	Items/basket		Sup_{min}	Number Rules
			Mean	Max		
1	Jan-Feb	872	2.50	8	0.0007429	1374
1	Mar-Apr	921	2.56	8	0.000645	1358
1	May-Jun	992	2.71	13	0.0006108	2572
2	Jan-Feb	1016	3.13	13	0.000429	2468
2	Mar-Apr	1060	3.16	16	0.0003805	3211
2	May-Jun	1111	3.26	13	0.0003455	2436
3	Jan-Feb	518	2.71	11	0.00268	2072
3	Mar-Apr	567	2.67	11	0.002392	2083
3	May-Jun	590	2.76	8	0.002277	1309

For each target customer it is only selected, from the RS_{CK}, the rules that include in its left hand side at least one item from the set of all products previously purchased (PS_{TC}) by the customer. This rule set is ordered by the lift measure, and the top-N rules with highest lift are added to the recommendation rule set for customer C - RRS_{TC}. The set of candidate products for recommendation to customer C - RPS_{TC} is the set of all products of the $RRS_{TC} – PS_{TC}$, to recommend items, which the customer has not purchased before. All candidate products are sorted and ranked according to their support and then the top-N products are recommended. Due to the fact that the number of recommended products (N) has influence in the recommendation accuracies, and the baskets size have different ranges and characteristics, experiments for each one of the datasets were made with N ranging from 2 to the maximum number of items/basket in that cluster/period, to get the best number of items to recommend. Table 7 presents the best number of items to recommend by cluster/period.

Table 7. Number of items to recommend by cluster/Period.

Cluster\Period	Jan-Feb	Mar-Apr	May-Jun
1	5	6	7
2	7	4	7
3	7	2	2

6. Experimental Results

The proposed system will be used to make individual customer recommendations for specific periods of 2014 year. To measure the potential interest of customers in the recommendation performed, there are two measures extensively adopted in the related literature[13], recall and precision defined in Eqs. 3, 4, respectively.

$$recall = \frac{n(BI \cap RI)}{n(BI)} \tag{3}$$

$$precision = \frac{n(BI \cap RI)}{n(RI)} \tag{4}$$

Where BI represents all items contained in the basket bought by the customer in the specific period of 2014 year, and RI stands for the items recommended to this basket. Recall is defined as the ratio of the number of correctly recommended items (i.e., the number of items recommended really purchased by customers) to the total number of purchased items. Precision is defined as the ratio of the number of correctly recommended items to the total number of recommended items. Increasing the number of recommended items tends to increase the recall and reduce the precision. However, the F_1 metric can be used to balance the trade-off between precision and recall. F_1 metric assigns equal weight to precision and recall.

$$F_1 = \frac{2 \times recall \times precision}{recall + precision} \quad (5)$$

It is important to remember that the recommendation list, by design, will contain no products previously purchased by the customer. Once the assessment will be made with new products not previously purchased by the customer, the algorithm is evaluated by the number of new recommended items purchased by the customer, and for the recommended items not purchased it is checked whether their category match with the category of the items bought by the customer.

In order to quantify the impact of the proposed recommender algorithm, it will be used a control recommendation that works as a "placebo" such as, the list of the past items bought by the customer that we call base recommendation. Base recommendation is a recommendation made only with the past items bought by the customer, that is, for a customer C and a period p, it is recommended the products purchased by C in the same period p, of the two previous years. Another goal of our recommender system is to provide a quantification of the increase in the value of sales achieved by the suggested recommendation. Tables 8 and 9 present the evaluation metrics and the average value of sales obtained with the base recommendation and with proposed hybrid algorithm, for the three first bi-months of 2014 year.

Table 8. Evaluation metrics of Base Recommendation.

Cluster	Period	recall	precision	F1	Avg Value of Sales
1	Jan-Feb	0,204	0,138	0,165	2,323
1	Mar-Apr	0,269	0,14	0,184	2,992
1	May-Jun	0,224	0,133	0,167	2,537
2	Jan-Feb	0,232	0,189	0,208	4,827
2	Mar-Apr	0,146	0,188	0,164	3,676
2	May-Jun	0,193	0,161	0,176	4,911
3	Jan-Feb	0,236	0,12	0,159	1,935
3	Mar-Apr	0,113	0,153	0,13	1,097
3	May-Jun	0,086	0,099	0,092	0,999
Overall				0,161	2,811

A table 9 show that in all cases the recommendation accuracy of the proposed algorithm is higher than that of base recommendation. The maximum F_1 (0,208) of the base recommendation, Cluster = 2, period Jan-Feb, is worse than that of the hybrid recommendation (0,323). The average value of the sales is also significantly larger than that of the base recommendation. The hybrid algorithm increases 96% the average value of the sales when compared with base recommendation. Meanwhile, the overall average F_1 of the hybrid recommendation (0.227) is also slightly better than that of base recommendation (0.161). Therefore, the hybrid algorithm proposed remarkably improves the average value of the sales without decreasing the recommendation accuracy.

Table 9. Evaluation metrics of Hybrid Recommendation.

Cluster	Period	recall	precision	F1	Avg Value of Sales
1	Jan-Feb	0,264	0,188	0,22	3,915
1	Mar-Apr	0,303	0,174	0,221	3,934
1	May-Jun	0,258	0,161	0,198	3,731

2	Jan-Feb	0,342	0,305	0,323	8,967
2	Mar-Apr	0,231	0,289	0,257	9,976
2	May-Jun	0,326	0,294	0,309	9,754
3	Jan-Feb	0,298	0,158	0,207	3,285
3	Mar-Apr	0,158	0,214	0,182	2,434
3	May-Jun	0,116	0,129	0,122	3,539
Overall				0,227	5,504

7. Conclusions

In this study a recommendation algorithm based on customer segmentation, followed by association rule generation to extract the best products to recommend to a target group of customers was described. The segmentation of customers based on costumer consumption behaviour through RFM attributes was adequate, since it has separated distinct groups of customers with different buying habits. Moreover, clustering customers into different groups not only improves the quality of recommendation but also allows selecting baskets of customers with similar buying habits. Also performing recommendations on specific periods of time is advantageous, because it permits to make recommendations specific to the period, which is important in a seasonal business like the perfumeries business. The experimental results show that the proposed algorithm indeed can yield recommendations of higher quality. However, evaluating the performance of a recommender system essentially requires feedback from the user. The success of the deployed system in influencing the customers can be really measured through the change in customer behaviour, such as the number of recommendations that are followed, or the change in revenue.

As future work we intend to monitor future sales and check them with our system recommendations.

Acknowledgements

The authors would like to thank *Mass Perfurmarias* for making available for this study their data sources and information about their customers, products and sales.

References

1. Mild, A., Reutterer, T. *An improved collaborative filtering approach for predicting cross-category purchases based on binary market basket data. Journal of Retailing and Consumer Services*, 10(3), pp. 123-133, 2003.
2. Schafer, J. B., Konstan, J. A., Riedl, J. *E-commerce recommendation applications*. In Applications of Data Mining to Electronic Commerce. Springer US, 115-153, 2001.
3. *Lu J., Wu D., Mao M., Wang W., Zhang G. Recommender system application developments: a survey. Decision Support Systems, 74, 12-32, 2015.*
4. *Park D. H., Kim H. K., Choi I. Y., Kim J. K. A literature review and classification of recommender systems research, Expert Systems with Applications, 39(11), 10059-10072, 2012.*
5. *Liu L., Mehandjiev N., Xu D.-L. Context similarity metric for multidimensional service recommendation, Int. J. Electron. Comm.,* 18 (1), 73 – 104, 2013.
6. *Huang S.-L. Designing utility-based recommender systems for e-commerce: evaluation of preference-elicitation methods Electron. Comm. Res. Appl.*, 10 398 – 407, 2011.
7. *Guo G., Zhang J., Thalmann D., Yorke-Smith N. Leveraging prior ratings for recommender systems in e-commerce Electron. Comm. Res. Appl., 13, 440 –455, 2014.*
8. *Cui H., Zhu M. Collaboration filtering recommendation optimization with user implicit feedback J. Comput. Inf. Syst., 10 (14), 5855 –5862, 2014.*
9. *Bellogin A., Cantador I., Diez F., Castells P., Chavarriaga E. An empirical comparison of social, collaborative filtering, and hybrid recommenders, ACM Transactions on Intelligent Systems and Technology (TIST) 4, 1–29, 2013.*
10. *Ba Q., Li X., Bai Z. Clustering collaborative filtering recommendation system based on SVD algorithm, IEEE Int. Conf. Software Eng. Service Sci., 963–967, 2013*
11. *Cai Y. fung Leung H. Li Q. Min H. Tang J. Li J. Typicality-based collaborative filtering recommendation, IEEE Trans. Knowl. Data Eng., 26 (3), 766–779, 2014*

12. *Gupta, S., Hanssens, D., Hardie, B., Kahn, W., Kumar, V., Lin, N., Sriram, S. Modeling customer lifetime value. Journal of Service Research, 9(2), 139-155, 2006.*
13. *Miglautsch J. Thoughts on RFM scoring, Journal of Database Marketing 8 (1), 67–72, 2000.*
14. *Jain A. K. Data clustering: 50 years beyond K-means. Pattern recognition letters, 31(8), 651-666, 2010.*
15. *Agrawal R., Imielinski T., Swami A. N. Mining association rules between sets of items in large databases. In Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 207-216, 1993.*
16. *Caliński, T., & Harabasz, J. A dendrite method for cluster analysis. Communications in Statistics-theory and Methods, 3(1), 1-27, 1974.*
17. *Gunawardana, A., & Shani, G. A survey of accuracy evaluation metrics of recommendation tasks. The Journal of Machine Learning Research, 10, 2935-2962, 2009*