Building clusters for CRM startegies by mining airlines customer data

Helena Sofia Miranda
Lisbon, Portugal
helenasofiamiranda@hotmail.com

Roberto Henriques
ISEGI, Universidade Nova de Lisboa Campolide, 1070-312
Lisbon, Portugal
roberto@isegi.unl.pt

Abstract— **As airlines strive to gain market share and sustain profitability in today's economically challenging environment, they should develop new ways to optimize their frequent flyer programs while increase revenues. Aware of the challenges, airlines want to implement a customer relationship management (CRM) strategy based on customer analytics and data mining techniques to support marketing decisions. So, to achieve this goal, we have to apply clustering techniques to the company customer databases and develop a single view of customer across their demographic and behavioral characteristics as well as their value for the company. This will enable the company to identify the most profitable customers and run marketing campaigns more efficiently.**

Keywords- Cluster analysis; airlines; data mining; decision support; customer relationship management

I. INTRODUCTION

The airlines industry reached a crossroad. The effects of worldwide economic slump and the rise of the fuel costs have severely impacted airlines economics and viability. Competition is forcing management to constantly cut costs and raise revenues, which demands for an approach to marketing that is more accountable, efficient and effective. Thus, to gain and keep market share, companies have to consider customer-level information [1], to target personalized marketing strategies to their needs and achieve a higher return on investment. Most companies in the airline industry are facing declining revenue per seat and increasing competitive pressure because of the deregulation and unfavorable economic conditions. At the same time, airlines product offering are nearly indistinguishable from another. Fares came under enormous pressure with pricing data proliferating on the web. Low costs carriers are opening up new segments, attracting new customers and taking market share from the establish airlines. Airlines companies know that competitive advantage in the long run will be based in large part on solid differentiated customer relationships. Therefore, deliver a consistent and distinctive customer experience and maintain low operating costs requires customer databases exploitation. But, how can we analyze more than one million customers and understand their differences to run campaigns more efficiently? To answer this challenge we have to use computational techniques such as data mining. If it is true that marketing and business users have long used data to segment customers, today's volume of customer data imposes more complexity to this task. There are several algorithms that can be used to segment customers; here we intent to evaluate the performance of three different algorithms, *k*-means, SOM and Hierarchical SOM and identify the most efficient for an airline company customer data set.

II. CLUSTERING METHODS

Even though there are several methods that can be classified as non-hierarchical clustering, we will cover k-means, Self-Organizing Map and Hierarchical SOM. Each of these three methods were chosen for one reason: k-means is one of the most widely used techniques for clustering analysis [2]; SOM has been pointed out to be less prone to local optima than k-means which allows the search space to be better explored and guarantees better results [3-4]; and, HSOM provides a new perspective on data by grouping it by themes.

A. *k-means algorithm*

The *k*-means may be one of the oldest and most widely used clustering algorithms among data miners [5]. The *k*-means algorithm is popular because it is easy to implement and has linear time complexity in the size of the dataset. It also presents the capacity to handle large databases [2]. This algorithm uses an iterative procedure, to set cluster centers. These centroids are the vectors of mean characteristics across the clusters members.

So, given the n points that become the initial cluster centers, each of the remaining points is assigned to the closest k cluster center according to its Euclidean distance. Once all points are grouped into k clusters, new clusters centers are calculated. This interactive process will stop when no more reassignments occur or the squared error ceases to decrease significantly.

This algorithm tend to produce equal-sized clusters because it implicitly assumes spherical shaped clusters a common error variance [6] and is not suitable for discovering clusters with convex shapes or very different sizes [7]. Due to the use of Euclidean distance, *k*-means is especially effective dealing with normal distributions. According to [2] a major problem with this algorithm is that it is sensitive to the selection of the initial partition and may converge to a local minimum of the criterion function value if the initial partition is not properly chosen. Also, *k*-means method can only be applied when the means of clusters are defined and do not perform well with qualitative attributes. This algorithm is very sensitive to noise and outlier data [7].

B. Self-organizing map

Self-Organizing Map (SOM) was first proposed by Tuevo Kohonen in 1982 [8] and was originally used for analyzing images. SOM basic idea is to map high-dimensional data onto one, two or three dimensions, maintaining the topological relations between data patterns. SOM "extract and illustrate" the essential structures in a dataset, through a map resulting from an unsupervised learning process [9-10]. SOM involve iterative procedures for associating a finite number of inputs (object vectors) with a finite number of representational points in such a way that proximity relationships between the inputs are respected by these representational points. The algorithm performs a non-linear mapping from a high dimensional data space to a low dimensional space, typically two-dimensional, rectangular grid [11] which allows the presentation of a multidimensional data in two dimensions. To do this, SOM uses an input layer and an output layer. Each unit in the output layer is connected to inputs (or attributes) in the input layer and the strength of each connection is measured by a weight. The weights between the input and the output layer are iteratively changed (this is called learning) until a termination criterion is satisfied. Further, SOM's convergence is controlled by various parameters such as the learning rate and the neighborhood of the winning layer input node in which learning takes place. Due to this competitive learning, similar patterns are automatically grouped by a single unit (neuron) based on data correlation. The output is said stable if no pattern in the training data changes its category after a finite number of learning interactions. To reach stability, the learning rate should be decreased to zero as iterations progress which affects the ability of the algorithm to adapt to new data [1]. One of the interesting properties of SOM is the capability of detecting small differences between the objects and its efficiency in finding multivariate data outliers [12-13]. Nevertheless, SOM as *k*-means, may generate a suboptimal partition if the initial weights are not chosen properly. Depending on the initial parameterization, the SOM can produce different results. In fact, there are multiple choices that have significant consequences on the final result, such as: the size of the map, the output space dimension, the initialization and the neighborhood function.

C. Hierarchical SOM

As we have already referred, the airline's customer information increases every day, placing additional pressure on the existing analysis tools and addressing concerns of efficiency, high-dimensionality feature extraction and data projection [14].Therefore, clustering may benefit from HSOM multilayer structure to gain efficiency throughout the stratification and exploitation of the databases from different thematic perspectives.

Traditional clustering methods, in which self-organizing maps [11] are included, are very sensitive to divergent variables. We understand by divergent variables those that present significant value differences to the general tendency.

To avoid this problem we propose the use of a hierarchical structure to explore and cluster customer information. With HSOM, variables are grouped in topics, where each topic will be independently clustered. These partial clusters are then used to create a global partition. By performing the clustering task in two stages, based on individual topics and only then globally, HSOM is less sensitive to divergent variables than SOM and other traditional clustering methods because divergent variables will merely have a direct impact on their own topic. In fact, this approach ensembles two main advantages: it reduces the dimensionality of the inputs and the number of units in each SOM granting HSOM less computational effort than a standard SOM [17] and allow HSOM to fit better due to its hierarchical structure, less sensitive to outliers and which may also provide an easier interpretation of the results.

In HSOM, the first level of SOM filters which data patterns are sent to the second level SOM by moving forward the index of the best matching unit, the quantization error, the coordinates of the best matching unit and all activation values for all units of the first level or any other type of data [17]. This information which is passed to the second level SOM is used to train it. A specific output of one SOM layer could be the original or an empty data pattern. However, many different arrangements are feasible for HSOM. These arrangements can vary in the number of layers used, the different methods connections are made and also in the information which is sent through each connection.

There are different possible taxonomies for Hierarchical SOMs. They can be classified as agglomerative or divisive [19]. The level of data abstraction in the agglomerative HSOM increases as the hierarchy goes up and the main goal is to create clusters which will be more general and provide an easier way to understand the data. Divisive HSOM is mostly less precise in the first level and is likely to be more exact as the levels of HSOM [17] go down. In the second level, the agglomerative HSOMs can be arranged by specific subjects about the clusters whilst divisive HSOMs can be arranged into static or dynamic. Here, we will focus on thematic agglomerative hierarchical SOM, and refer to it simply as HSOM.

The main advantage of HSOM or SOM clustering algorithm with respect to *k*-means is the adaptive distance measure.

III. BUILDING CLUSTERS

A. Dataset used in this problem

In this project, we use an airline customer database to evaluate the performance of *k*-means, SOM and HSOM. This dataset is a random sample of flight active member's original database. This data contains information of 20.000 customers and describes customer's age, gender, country of residence, number of years has a client, top routes, top brand booking, the number of months since last flight, as well as member's flight miles, promotion miles and redeemed miles, and the number of flown segments.

B. Implementing clustering procedures

We started by training automatic *k*-means in order to have an idea of the numbers of clusters and then we built an SOM and HSOM. For *k*-means we have used SAS Enterprise Miner

Tools, while in the case of SOM we have used two tools: SAS Enterprise Miner which is one of the most widely used software and GeoSOM Suite [17] which uses the original SOM Kohonen algorithm. HSOM have been calculated in GeoSOM Suite tools. Therefore we will refer to SOM calculated in Enterprise Miner as SOM EM and to SOM calculated in GeoSOM as SOM Toolbox.

To identify the optimal number of clusters for *k*-means we have applied the basic *k*-means algorithm randomly choosing the initial clusters centers for a maximum number of 20 clusters and considering the min-max as internal standardization criterion and the Ward Clustering Method to guarantee low variance within the clusters. Cubic Clustering Criterion suggests 5 clusters. Then, we have tested the results suggested by CCC, running k-means several times.

To test the results suggested by *k*-means, we have run Kohonen SOM in SAS Miner several times, with random initialization and min-max for 3x1, 4x1, 5x1, 6x1 and 7x1neurons.

We have also tested the optima solution for SOM Toolbox. The SOM method was implemented with a 15x10 regular SOM lattice. Data have been normalized according to the min-max method and the neurons have been random initialized. SOM Toolbox algorithm train was sequential. We have trained the algorithm for 200 epochs, a learning rate of 0.3 and the radius is set to 8 in the rough train and finish using in the finetune of 400 epochs, a learning rate of 0.1 and a radius of 4 neurons. U-matrix may suggest two or three clusters but this is not clear. Although, we decided to implement several SOM with the following sizes: 3x1, 4x1, 5x1, 6x1and 7x1.

HSOM was implemented in the GeoSOM Suite [18]. This tool presents an interface where the user can choose the HSOM inputs, based on the SOMs created before. Thus, we have created a structure that combines two levels of SOMs. The lowest level has two SOM one for the customer purchase behavior and the other for the purchase frequency. The top level is composed by one SOM that receives as input the U-matrices coordinates from the two lowest levels SOMs.

All SOMs were trained using the sequential algorithm. We have started by training the algorithms for 200 epochs, a learning rate of 0.3 and the radius is set to 8 in the rough train and finish using in the finetune of 400 epochs, a learning rate of 0.1 and a radius of 4 neurons. In Figure 1 we present HSOM U-matrix results.

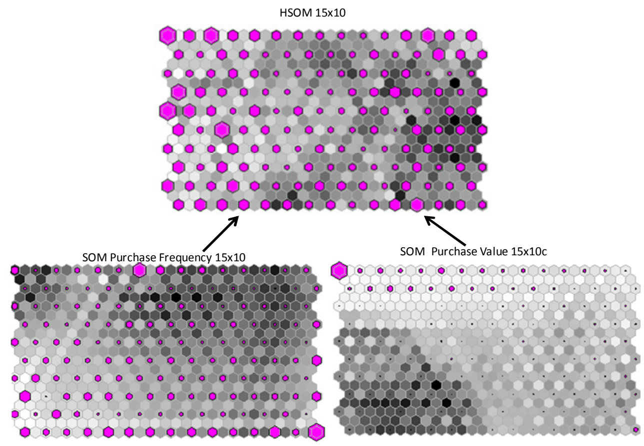

Figure 1. Aglomerative HSOM example

C. Comparing cluster algorithms and number of clusters

In this section we present the comparison of the most relevant statistics. A general analysis of the distances comparison shows a tendency for *k*-means to outperform SOM and HSOM (Figure 2). The sum of distances between the observation and the cluster's seeds is always is smaller for *k*-means except for *k*=3 where SOM EM achieves the smallest value. SOM Toolbox achieves the second best results in terms of distance.

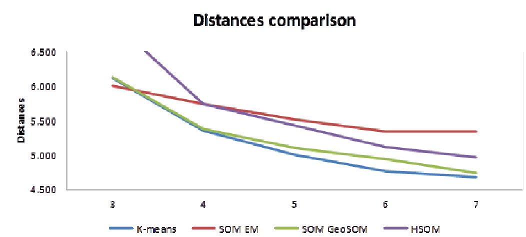

Figure 2. Distances comparasion between algorithms

The comparison the coefficient of determination attests the results suggested by the distances comparison. Most of the times, *k*-means achieves higher values for R^2 than SOM EM, SOM Toolbox and HSOM. The second best solution is SOM Toolbox solution.

TABLE I. COEFFICIENT OF DETERMINATION COMPARISON

Number of clusters	R^2 K-means	R^2 SOM EM	R^2 SOM GeoSOM	R^2 HSOM
3	0,398553	0,403805	0,371793	0,226544
4	0,547713	0,449904	0,502558	0,429136
5	0,603092	0,483723	0,549319	0,479097
6	0,629810	0,526087	0,570988	0,543490
7	0,657312	0,538599	0,601959	0,569999

Finally, we have compared the results achieved through Pseudo F statistics and we attest that *k*-means has better results. After *k*-means, SOM Toolbox is the solution with better results in terms of Pseudo F statistics. The worst result is obtained with HSOM algorithm.

TABLE II. COEFFICIENT OF DETERMINATION COMPARISON

Number of Clusters	Pseudo F K-means	Pseudo F SOM EM	Pseudo F SOM Toolbox	Pseudo F HSOM
3	6619.95	6766.27	5912.41	2926.05
4	8064.75	5446.71	6728.16	5006.28
5	7589.02	4679.57	6087.62	4593.67
6	6797.44	4435.26	5317.64	4756.65
7	6386.01	3886.37	5034.97	4413.28

All statistics analyzed here are related with within-class variance and *k*-means procedure appears to give partitions that are reasonably efficient in terms of within class variance [19]. Here we will opt for k=4 as an optimal cluster solution due to the results of the three statistics.

Finally, in order to compare the algorithms used and the cluster's distribution produced by each one, we have mapped them in a 15x10 SOM U-matrix.

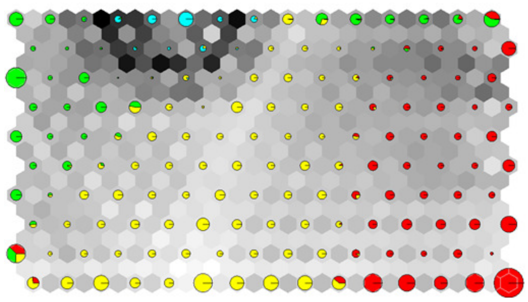

Figure 3. *k*-means clusters mapped on the 15x10 SOM U-Matrix

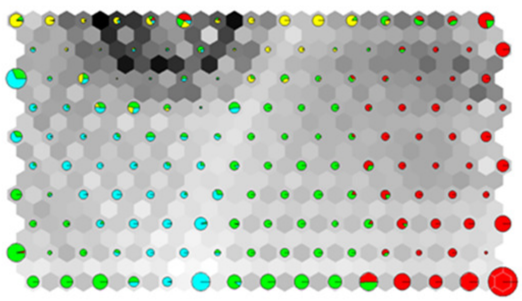

Figure 4. SOM EM clusters mapped on the 15x10 SOM U-Matrix

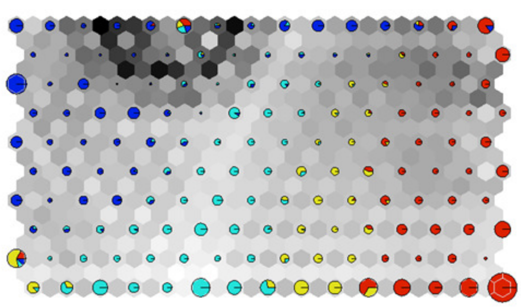

Figure 5. SOM Toolbox clusters mapped on the 15x10 SOM U-Matrix

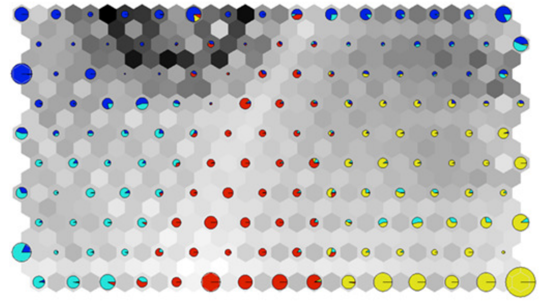

Figure 6. HSOM clusters mapped on the 15x10 SOM U-Matrix

Figures 3 to 6 confirm that SOM Toolbox has a distribution in space similar to *k*-means. In terms of the reliability of the interpretation and cluster profile we found *k*-means more intuitive. This result is compatible with the fact that the distances inter clusters are higher with *k*-means as shown in the table III. *k*-means inter-clusters higher distances may result in an easier interpretation because the clusters are more dissimilar.

TABLE III. COEFFICIENT OF DETERMINATION COMPARISON

Algorithm	Inter clusters distances for K=4
K-means	20,274.27
SOM EM	17,278.44
SOM Toolbox	16,798.53
HSOM	15,796.95

IV. CONCLUSIONS

In this paper we compared *k*-means, SOM and H-SOM to cluster an airline customer database. We conclude that *k*-means and SOM present similar results, although *k*-means is statistically superior to SOM. *k*-means clusters appears to be more intuitive in terms of cluster's profile due to its higher inter-cluster distances. In the presence of these findings we believe that will be easier for the company to define targeted marketing strategies.

REFERENCES

[1] Kumar, V. and J. A. Petersen (2005). "Using a customer-level marketing strategy to enhance firm performance: A review of theoretical and empirical evidence." Journal of the Academy of Marketing Science 33(4): 504-519.

[2] Jain, A. K., M. N. Murty, et al. (1999). "Data clustering: A review." Acm Computing Surveys 31(3): 264-323.

[3] Bodt, E. d., M. Cottrell, et al. (1999). "Using the Kohonen Algorithm for Quick Initialization of Simple Competitive Learning Algorithms." ESANN'1999 Bruges.

[4] Bacao, F., V. Lobo, et al. (2005). Self-organizing maps as substitutes for k-means clustering. Computational Science - Iccs 2005, Pt 3. V. S. Sunderam, G. D. VanAlbada, P. M. A. Sloot and J. J. Dongarra. Berlin, Springer-Verlag Berlin. 3516: 476-483.

[5] Steinbach, M., V. Kumar, et al. (2006). Introduction to data mining.

[6] Everitt, S. B., S. Landau, et al. (2011). Cluster Analysis, 5th Edition.

[7] Han, J. and M. Kamber (2006). Data mining: concepts and techniques, Elsevier.

[8] Kohonen, T. (1982). "Self-organized formation of topologically correct feature maps." Biological Cybernetics 43(1): 59-69.

[9] Kaski, S. and T. Kohonen (1996). "Exploratory data analysis by the self-organizing map: structures of welfare and poverty in the world." Neural Networks in Financial Engineering. N. Apostolos-Paul, Yaser Refenes, Yaser Abu-Mostafa, John Moody and A. Weigend. Singapore, World Scientific: 498-507.

[10] Kaski, S., J. Nikkilä, et al. (1998). "Methods for interpreting a self-organized map in data analysis." Proceedings of ESANN'98 6th European Symposium on Artificial Neural Networks, Bruges, Belgium, D-Facto.

[11] Kohonen, T. (2001). Self-Organizing Maps, Berlin, Springer-Verlag.

[12] Munoz, A. and J. Muruzabal (1998). "Self-organizing maps for outlier detection." Neurocomputing 18(1-3): 33-60.

[13] Nag, A. K., A. Mitra, et al. (2005). "Multiple outlier detection in multivariate data using self-organizing maps title." Computational Statistics 20(2): 245-264.

[14] Mao, J. and A. K. Jain (1995). "Artificial Neural Networks for Feature Extraction and Multivariate Data Projection." IEEE Trans Neural Networks 6: 296-317.

[15] Mukkulainen, R. (1990). "Script Recognition with Hierarchical Feature Maps." Connection Science 2(1-2): 83-101.

[16] Luttrell, S. P. (1989). "Hierarchical Vector Quantization." Iee Proceedings-I Communications Speech and Vision 136(6): 405-413.

[17] Henriques, R. (2010). Artificial Intelligence in Geospatial Analysis: applications of Self-Organizing Maps in the context of Geographic Information Science, Universidade Nova de Lisboa – ISEGI.

[18] Henriques, R., Bacao, F., and V. Lobo (2012). "Exploratory geospatial data analysis using the GeoSOM suite." Computers, Environment and Urban Systems 36(3): 218-232.

[19] Ding, C. and X. He (2002). "Cluster merging and splitting in hierarchical clustering algorithms." IEEE Computer Society Proceedings of the 2002 IEEE International Conference on Data Mining.

[20] MacQueen, J. (1967). Some methods for classification and analysis of multivariate observation. 5th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press.