Store Attribute Weighting for Clustering in Fast Fashion

Ana Cristina Neto Andrade

Master's Dissertation

FEUP supervisor: Prof. Paulo Luís Cardoso Osswald

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

Mestrado Integrado em Engenharia e Gestão Industrial

2019-07-01

Abstract

This project was developed at an IT service company specialized in retail consulting, as a way to facilitate the Store Clustering step of the Assortment Planning process, which the company implements in many different retail clients. Store clustering is the procedure of grouping the different Points of Sale of a retailer according to certain demand criteria in order to then use these clusters to allocate the product mix that better fit each cluster's demand. These store clusters are defined according to store attributes, in a way that, idealy, groups stores that have similar demand patterns together. However, each of the store characteristics has a different impact on the resulting store demand, meaning that it should have a different weight on the clustering process. Currently this attribute weighting is allocated according to expert opinion. This project has the goal of using statistics and artificial intelligence to explore past sales data from the different stores in order to define these weights.

In this work, two different fashion retailers with very different characteristics were considered. Although store clustering is a common process to all retail chains, fashion retailers were chosen for the constant changing nature of their environment, where most products don't last long in the stores and customer preferences develop in a particularly quick way.

In order to calculate the final attribute weights, both statistical and machine learning algorithms are considered. Xie-Beni Weighing is a statistical algorithm that looks into the ability the different attributes have of clustering the data, by dividing stores according to the attribute values and evaluating how well they explain sales data variations.

For the machine learning algorithms, initial clusters were set according to past sales data and the resulting store classification was used in the algorithms as the output to be deduced from the store attributes inputed. The machine learning algorithms considered were Random Forest and Regularized Random Forest. From both of these, variable importance can be extracted directly, meaning attribute weights can be calculated right ahead. In another approach, weights were also measured through the out-of-bag estimate error rate in the Random Forest algorithm, comparing its value when all the attributes are used as input versus when one of them is left out.

Once the weights are defined, it is necessary to understand what number of clusters is able to encompass them better. For that purpose, the number of cluster variable was tuned for each different clustering and its defined set of attribute weights, considering the Silhouette and the Calinski and Harabasz indexes.

Considering the final clusters, and comparing them with actual sales data, it was found that a small number of clusters is desirable. It was also observed that the Xie-Beni Weighting doesn't work well when the number of attribute values varies a lot between attributes and the out-of-bag estimate error rate method causes problems when two variables contain very similar information, resulting in alienating both of them, when only one of them should be given a much lesser weight. Looking into how each method performed for each retailer, there is no evidence that a specific algorithm fitted a specific type of retailer better.

Acknowledgements

Many people contributed to the realization of this dissertation, in both very direct and indirect ways.

First I'd like thank FEUP Professor Vera Miguéis, for her availability to discuss ideas and readiness to provide great insight into the more technical parts of this project.

I'd also like to thank KSR – Retail Consult for allowing me to develop this project. I'd like to thank the whole team for their constant rediness to help with anything they could, with a special mention to Carlos Lima, Nuno Santos, and Paulo Marques, who guided me through the whole process.

Next, Professor Paulo Osswald, my dissertation supervisor, for looking over this whole process, proposing new visions and points of view.

Ana Rita and Maria João, who were also developing their dissertation projects at KSR, and who helped me through the hardest parts of this project, through great discussions about each other's projects that provided a fresh angle to tackle any problem, but also through constant reassurement and support.

Finally, to all my friends and family who all have in one way or another supported me through the development of this project, especially to my parents and sister who are my biggest rock, to my grandparent who are always willing to go above and beyond to make any hard day much brighter, and to Amélia who was always there for me when I needed.

Content Index

Figure Index

Table Index

1 Introduction

1.1 Project framework and motivation

Store attribute weighting for clustering is the evaluation of certain store attributes in their capacity to produce quality store clustering, that will in turn be used in the process of allocating products to groups of stores. A quality store clustering is defined as one that maximizes the probability of a product allocated to a cluster of stores to perform similarly in all those stores.

This step is included within the Assortment Planning process, which deals with understanding which products to sell and where to sell them. Store Clustering happens after the definition of the products that are to be sold and before they're allocated to stores, as a way to smooth this procedure as it isn't feasible to allocate each product to each store individually.

Although this is a common procedure to all retail chains, fast fashion is characterized by its dynamic nature, where new products have to be constantly matched with an ever-changing demand, making it especially important to target the right customers with the right products at the right time, and making, therefore, the quality of these clusters vital to a company's success.

KSR - Retail Consult is a company that works with the Oracle Retail Solutions, implementing them in retailers, while adapting the base software to their needs. This project has the goal of working as a complement to the Assortment Planning solutions regarding store clustering. Currently, the Assortment Planning solution relies on its users to input the attribute weights to form these clusters.

1.2 Project Goals

The process of selecting clustering attributes and their respective weights, as well as the ideal number of clusters is still mainly dependent on the opinion of experts, and, consequently, on their experience. Therefore, the goal of this project is to find and adapt statistical and artificial intelligence methods that can provide good features, weights, and number of clusters for the clustering process, without human opinion or as a decision support system for those who make these calls.

The resulting clusters should provide a good assignment of all the Points of Sale regarding the expected demands of the costumers that visit them. The clusters should reflect the type of sales carried out at the stores, considering the factors that lead a customer to buy at a specific place, which could be physical proximity, store environment, and many others.

With the consideration of two different fashion retailers, with different approaches to fast fashion, it was also thought to understand if there are methods that work better for a certain type of management. Since sales data can be aggregated at many different levels, understanding which level works better for each of the retailers ensures the best trade-off between the information the lower levels provide and the advantages of high aggregation, like lower influence of non-common data.

1.3 Methodology

This project's activities can be divided into 7 steps, as resumed in Figure 1.

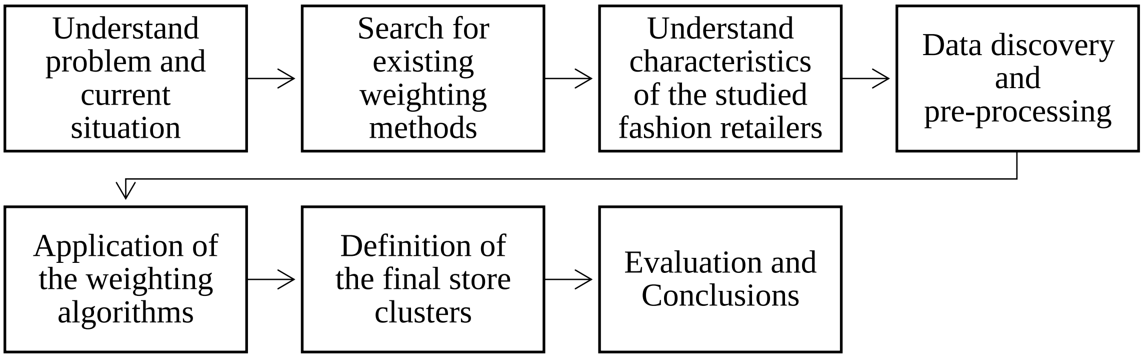

Figure 1 - Methodology.

On a first approach to the problem, and as it is necessary to understand the context in which current state of the art solutions are used, there was a need to review Oracle's Assortment Planning solution, which was taken as an example of good Assortment Planning practices that KSR – Retail Consult has implemented with good results. As store clustering is a step within the Assortment Planning process, it is vital to not only understand how it is being performed, but also what previous actions and decisions will affect it, as well as how the clusters will influence the following procedures.

Based on the knowledge regarding the current assortment planning and store clustering procedures, the next step is to find existing methods that can be used to solve this particular problem. As attribute weighting for store clustering is quite an unexplored topic, variable weighting methods for other type of problems, such as decision making, were also studied. At the end of this step, the different available methods, their conditions, advantages, and disadvantages were analyzed and compiled.

Data from the two retailers was then analyzed, starting with data discovery and understanding, as well as research of the fashion retailers to be analyzed. Data was pre-processed and re-grouped into datasets that fit the different methods to be analyzed. The dataset was partitioned into training data, to be put through the different methods in order to obtain the weights, and testing data used, in the end, to compare the final store clusters to the actual sales values.

After the different weighting methods were applied to the training data, the data matrix of the stores to consider for the clustering, with their respective attributes weighted, was divided into a given number of clusters. This number of clusters also needs to be defined, as different weights for different attributes may require a different number of store groups. After the number of clusters was tuned, the final store clusters were determined.

These final clusters were then evaluated, comparing the different methods for each fashion retailer and analyzing the overall method quality, as well as their fit to each retailer.

1.4 Structure of the Dissertation

After this introduction, Chapter 2,

State of the Art, deals with the current practices regarding the impact of store attributes in store clustering. After this is established, various attribute weighting methods are considered, not only the ones conceived with clustering in mind, but also those more geared towards decision making problems. Both statistical and machine learning algorithms were looked into. Finally, and because to define clusters it is also needed to understand what's the best number of clusters for a specific situation, different methods that tune this parameter were also considered.

Chapter 3, Problem Definition, asserts, at first, where store clustering happens inside the bigger Assortment Planning process, understanding which steps affect and are affected by it, as well as the current practice for this step inside Oracle's Assortment Planning and Advanced Store Clustering solutions.

In this chapter, the Fashion Retailers to consider are defined according to their characteristics, type of available data, and store attributes to consider.

Inside Chapter 4, Proposed Solution, the different methods are explained, taking into account their implementation using R Software, looking into the required preparations, as well as the specific tunings and calculations needed for each one of them. After this, the ways to evaluate the clusters are considered, as it is through the results from this evaluation indexes that the capability of each method to define attribute weights for the clustering process is to be determined. Afterwards, the different results are considered and analyzed, for the data of each Fashion Retailer and all the considered methods, and conclusions are drawn regarding each of them.

Finally, in chapter 5, general conclusions are deduced and prospects of future work are defined.

2 State of the Art

2.1 Store Attributes and Clustering

Attribute weighting in store clustering is used in order to obtain better store clusters for the purpose of product allocation. Product allocation is the process of determining which stores will sell which products, for a given period of time. As big retailers deal with both a big number of stores and a wide range of products, allocating each product to each store individually becomes impossible. To allow a faster, yet still effective, allocation, stores are grouped into clusters, which, ideally, will sell the same products in a similar manner.

These clusters are defined according to the characteristics of the stores within them, regarding their physical attributes, such as type of store and available space, or their customer base, which characterize the average consumer according to what they buy in each store, or, in some cases where this data is available, their demographics. However, these store characteristics have a different impact on the actual sales of the stores, for example the climate of the city usually has a different impact on the products sold when compared with the type of store. As such, the step of weighting all the available store attributes, in order to produce store clusters corresponding to these weights, is of extreme importance to achieve quality store clusters. Within the fast fashion market, characterized by constant change, these clusters are regularly updated as experts try to adapt the weight of each of the attributes to fit current trends. Some of the common attributes that can be consider are climate, which can vary from extreme hot to extreme cold, wet to very dry; geographic region, such as city, country, and continent; type of store, that could be brand-owned, a franchise or a department store; and store concept, used, for example, for beach stores.

The attribute weighting for store clustering is, for the most part, a theoretical and subjective problem where the weight values are obtained through expert opinion and not data analysis. In fact, different authors have studied which clustering algorithms better suit this problem, taking both the attribute weight and the number of clusters as previously defined variables (Kargari and Sepehri 2012; Oner and Oztaysi 2018). Some other focus on understanding what influences these expert decisions and the impact they have on overall company performance (Bahng and Kincade 2014).

Nonetheless, different algorithms have been developed in order to identify the important attributes, as well as their respective weights, for more general clustering processes. A majority of these algorithms (He, Xu, and Deng 2011; Chen and Wang 2006; Modha and Spangler 2003; Huang et al. 2005; Bai et al. 2011) focus on changing attribute weights in each step of the clustering process, based on the way the clusters evolve, which is not the goal of this project. Some other methods defined attribute weights a priori, however worked only under numerical data (Steinley and Brusco 2008), which isn’t viable since most store attributes are categorical.

Looking into algorithms which define attributes a priori and work well with categorical data, these can be divided into two groups, the ones that look into how well the attributes can cluster the rest of the data (Rahman and Islam 2015; Jia et al. 2013) and those who explore the attribute relationship and relevance using prediction models (Houtao Deng and Runger 2012a; Revathy and Lawrance 2017a; Doğan, Arditi, and Murat Günaydin 2008a). This latter type of predictors are based on decision tree and random forest algorithms and mostly used in classification or decision making problems or with a feature selection intent, however they can be adapted in order to obtain feature weights, which can then be used for clustering.

2.2 Current Store Clustering Practices

In order to obtain store clusters, multiple clustering algorithms can be used, however simpler methods, based on k-means, tend to be preferred as they require smaller processing times and computational power, while obtaining similar results, given the type of data usually available. These clusters are defined according to the main store attributes and the weight these should have in the clustering process, the current practice (Goodman 2016; Garro 2011) being that both the selection of the main attributes and the definition of their relative weights are set based on expert's opinions.

Bahng and Kincade (2014) explored the relationship between the relevant attributes experts considered in Assortment Planning and the type of company under study, as well the expert's own demographics. They showed that the choice of relevant attributes changed not only based on the characteristics of the company but also depending on the age, sex and experience of the expert, with, for example, older experts usually opting for more traditional attributes, like store size, while their younger counterparts take into account the available information regarding the customer segment they want to target and their needs. This exposes the intrinsic partiality of the evaluation, showing that the quality of the assortment, and the consequent success of the company, depend on the demographics and experience of the experts involved in the assortment process.

Many articles have undertaken to find out what makes a customer choose a store over another, with focus on both the physical attributes of the store, like layout and general aesthetic, and the type of products it offers, regarding quality, variety, price, etc. Moye (2000) concluded that "Sensory/Layout" and "Music/Aesthetics" accounted for a big part of the reasons customers opt for shopping in a store, with special emphasis on the first, while aspects like the dimension of the space didn't appear to be as important. Thommpson et al (2018), on the other hand, found that "store features", like layout, personnel, and presentation, explained a big part of the reason younger generations chose one store over another. This was also documented in a study involving university students (Makgopa 2018), which named store atmospheric attributes, "environmental cleanliness", as well as store content, as relevant to the customer's choice.

With the constant development of information technology, more and more data is available to the decision makes, regarding customer preferences regarding products, but also points of sale (Şen 2008). However, only with correct interpretation of this data can valuable information be extracted.

2.3 Attribute Weighting

2.3.1 Xie-Beni Index Weighting

Exploring the first type of methods that look into the clustering ability of the store attributes, Rahman and Islam (2015) proposed dividing the dataset according to the different attribute values and evaluating the quality of the clusters each feature provides. In this method the cluster evaluation is done through the Xie-Beni Index (Xie and Beni 1991), given by the formula (2.1), which is based on fuzzy clustering (Ruspini 1969), the distance between categorical records being obtained through a similarity measure (Giggins and Brankovic 2012).

$$XB_a = \frac{\sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^2 \left\|V_i - X_j\right\|^2}{n \min_{i,j}\left\|V_i - V_j\right\|^2} \qquad (2.1)$$

Where:

XB_a is the Xie-Beni index of each attribute a

u_{ij} is the fuzzy membership of point j to the cluster i, given by formula (2.2), and

$\left\|V_i - X_j\right\|$ is the Euclidean distance between the data point X_j and the center V_i.

The fuzzy membership, which defines the level of membership of each point to each cluster is given by:

$$u_{ij} = \frac{\left(\frac{1}{d^2(X_j, V_i)}\right)^{\frac{1}{m-1}}}{\sum_{i=1}^{c}\left(\frac{1}{d^2(X_j, V_i)}\right)^{\frac{1}{m-1}}} \qquad (2.2)$$

Where:

X_j is a data point,

V_i is a cluster center, and

m is a predefined fuzzifier parameter.

Taking the Xie-Beni index, Rahman and Islam (2015) define the weight W_a (2.3) of each attribute based on the normalization of the index values, and where $|A|$ is the total number of attributes.

$$W_a = 1 - \frac{XB_a}{\sum_{a=1}^{|A|} XB_a} \qquad (2.3)$$

Where:

W_a is the weight of each attribute a,

XB_a is the Xie-Beni index of attribute a defined in (2.1), and

$|A|$ is the total number of attributes.

2.3.2 Knowledge Entropy Feature Weighting

The second of this type of methods (Jia et al. 2013) is based in knowledge entropy in rough sets, so as to understand the capability each attribute has of partitioning the dataset without needing to do any previous analysis. In rough set theory, first proposed by Pawlak (1982), knowledge is defined as the ability to classify data objects, considering the universe is a set and knowledge is the capability of creating partitions in such a set (M. Li and Zhang 2005). The relationship between rough sets and knowledge entropy was studied by Miao and Wang (1998), treating knowledge as a variable in rough set, given that the smaller the information entropy, the bigger the information and the smaller the uncertainty within the knowledge (Jia et al. 2013).

Jia et al. (2013) propose that, after due preprocessing, which includes discretizing any continuous numerical attributes, the knowledge entropy of each attribute is calculated, considering the partitions its values create in the universe of the data set. The knowledge entropy caused by this partition is calculated through (2.4).

$$H(a_j) = 1 - \sum_{i=1}^{n} [p(X_i)]^2 \qquad (2.4)$$

Where:

a_j is attribute j,

X_i is class i of attribute a_j, and

$p(X_i)$ is the probability of the class X_i in data set U, defined in (2.5).

$$p(X_i) = \frac{|X_i|}{|U|} \qquad (2.5)$$

Where:

$|X_i|$ is the number of elements in class X_i, and

$|U|$ is the number of elements in data set U.

After the knowledge entropy is obtained, the attribute weight is defined as the normalization of the attribute's knowledge entropy, through (2.6).

$$W_j = \frac{H(a_j)}{\sum_{j=1}^{n} H(a_j)} \qquad (2.6)$$

Where:

W_j is the weight of attribute j, and

$H(a_j)$ is the information entropy of attribute j defined in (2.4).

Jia et al. (2013) found that this weighting method "fully taps the information contained in each attribute", which lead to better clustering results. However, the weights obtained were then used in spectral clustering and not a k-means type algorithm, meaning the results can be quite different.

In both these methods all attributes are considered for the clustering, meaning that all attributes get assigned a weight, which is not always preferable, even if irrelevant attributes are set to have little importance. Having a reduced set of features allows not only for shorter training time and an easier interpretation of the model, but also increases generalization, which will reduce overfitting (Pundir and Amrita 2013a).

2.3.3 Info-DTree

Regarding the second type of attribute weighting, which looks into the attribute's ability to predict a certain outcome, random forest and decision trees were the chosen prediction models, as they are quite simple to understand – especially when dealing with a small number of attributes – and still allow good results. Tree models can handle numerical and categorical data simultaneously, as well as missing values, features with differing scales, non-linearity and synergies between attributes (Houtao Deng and Runger 2012b).

The Info-Dtree method, proposed by Doğan, Arditi, and Murat Günaydin (2008b), exploits the resulting decision tree to take information about the relevance of the attributes, considering their location in the decision tree. The attribute weight is obtained through the sum of information gain each time the attribute appears in the tree multiplied by the proportion of input cases classified by the attribute.

This method is rather simple and intuitive; however, the results obtained the considered article were not of the best quality.

2.3.4 Regularized Trees

Keeping the concept of a tree model, random forest analyzes different decision trees to obtain the best prediction these can provide and was found to be more accurate than single tree models (Breiman 2001a). Both Deng and Runger (2012b) and Revathy and Lawrance (2017b) work with tree-type algorithms to define the relevant features in a dataset.

Different prediction algorithms, such as neural networks, support vector machine, regression models and Naïve Bayes (J. Li et al. 2017), have been used with the objective of feature selection and weighting. However, Random Forest presents the best accuracy in current data mining algorithms, it is able to work well with large unbalanced datasets, all the while not having issues with nominal data and not overfitting (Sahami et al. 1998).

Deng and Runger (2012b) suggest a feature selection algorithm that can be applied to any tree-type prediction algorithm, such as Random Forest (Breiman 2001b), C4.5 (Quinlan 1993), and Boosted Random Forest (Freund and Schapire 1996), and the algorithm was shown to be more successful when implemented along with Random Forest. The proposed framework, Regularized Trees, works in such a way that, at each node of the prediction tree, a new feature – not previously used in this same tree – is only selected if there's a considerable gain in using it instead of an already selected feature, producing a more compact set of attributes that define the dataset. This variable selection process that happens at each node is described mathematically in (2.7).

$$\mathrm{gain}_R(X_j) = \begin{cases} \lambda * \mathrm{gain}(X_j) & X_j \notin F \\ \mathrm{gain}(X_i) & X_i \in F \end{cases} \quad (2.7)$$

Where:

X_j is the new feature,

X_i is a feature already selected for a previous split,

λ is the coefficient that defines the penalty of selecting a previously unused features, and

F is the set of features already used to split the tree in precious nodes.

A new feature will only be selected if $\mathrm{gain}(X_j)$ is considerably larger than $\max_i(\mathrm{gain}(X_i))$. Considering that $\lambda \in [0,1]$, values closer to 0 reflect a greater preference for selecting attributes previously used, while greater values will not penalize new splits as much. Deng and Runger (2012b) proposed working with $\lambda = 0.5$ after finding that the results did not vary significantly with the variation of λ. When compared with other prediction algorithms, Random Forest, with previously selected features defined using Regularized Random Forest (RRF), performed similarly to Support Vector Machine based on recursive feature elimination SVM-RFE (Guyon et al. 2002), but in a more expedite way and without the need to set as much parameters.

Regularized Trees (Houtao Deng and Runger 2012b) has feature selection as objective, and not feature weighting, however the results can be interpreted with the objective of obtaining feature relevance and, therefore, weight, similar to what is done by Pundir and Amrita (2013b). Feature selection algorithms can be classified as filters, where supervised learning is not used, wrappers, in which there's a search for an optimal set of features to be used in a specific learner, or embedded methods, that uses the results of learners to make decisions about feature selection (Houtao Deng and Runger 2012b). Regularized Trees falls under the embedded method classification, as it uses a tree predictor to make conclusions about feature relevance, and not only to find out which features will work better in the specific tree algorithm.

2.3.5 Random Forest Filter

Revathy and Lawrance (2017b) use a Random Forest filter, where both the mean decrease impurity and mean decrease accuracy are used, to define the weight of the features to consider for a future classification. Impurity makes use of the Gini measure (2.8), which is used in order to choose the split with the highest impurity at each node, as through the formula (2.9), given that the average of all decreases in Gini impurity gives the "Gini importance" (Rosario and Thangadurai 2015).

$$\mathrm{GI(t)} = 1 - \sum \mathrm{p(k|t)}^2 \tag{2.8}$$

Where:

$\mathrm{GI(t)}$ is the Gini Index in node t, and

$\mathrm{p(k|t)}$ is the probability of class k being correctly discriminated at node t.

$$\Delta\mathrm{GI(t)} = \mathrm{PtGI(t)} - \mathrm{PLGI(tL)} - \mathrm{PRGI(tR)} \tag{2.9}$$

Where:

$\Delta\mathrm{GI(t)}$ is the Gini impurity in node t,

$\mathrm{PtGI(t)}$ is the Gini Index of node t defined in (2.8),

$\mathrm{PtGI(tL)}$ is the value of the Gini index on the left side of the node t, and

$\mathrm{PtGI(tR)}$ is the value of the Gini index on the right side of the node t.

Differently, the mean decrease accuracy handles "out-of-bag (OOB) errors", which estimates prediction error. It works by permuting feature values and measure the decrease in the classifier accuracy, given that, for unimportant variables, the permutation should not cause a decrease in accuracy. Considering the permuted variables $\mathrm{X_j}$, with $\mathrm{j} \in [1, \mathrm{M}]$, where M is the total number of variables, the disturbed OOB sample is composed by all the $\mathrm{X_j}$ permuted variable along with the non-permuted variables. And the variable importance, given by (2.10) can be considered taking into account the impact the permutations have on the OOB error.

$$\mathrm{VI}(\mathrm{X_j}) = \frac{1}{\mathrm{ntree}} \sum \left(\mathrm{errOOBf} - \mathrm{errOOBf_j}\right) \tag{2.10}$$

Where:

$\mathrm{VI}(X_j)$ is the variable importance of the permuted variable X_j,

ntree is the number of trees in a forest,

errOOBf is the error of the tree f, and

$\mathrm{errOOBf}_j$ is the error of the tree f with the permuted variable X_j.

2.4 Number of Clusters

The number of clusters to consider when performing the clustering algorithm, after defining the attributes to use and their respective weights, is also an important problem to address, as any initially defined parameter influences the quality of the final results (Estiri, Omran, and Murphy 2018).

Regarding store clustering, ebp Global (2018) gives some guidelines about the relationship between the number of stores and the number of clusters to bundle them in, since having too many clusters can lead to worse performing assortments. Assortments with 5 to 6 clusters are recommended for smaller groups of 300 to 500 stores, while for bigger store populations of 3000 to 4000, 8 to 12 store clusters are recommended.

2.4.1 Intra and Inter Cluster Analysis

Doan and Nguyen (2018) take into account intra (2.11) and inter (2.12) cluster analysis coefficients, defined by the distortion within a cluster and the distance between two clusters, respectively.

$$\overline{\alpha} = \frac{\overline{d}_{max}}{\overline{d}_{avr}} = \frac{\max\limits_{q}\left(\sqrt{\sum_{l=1}^{k} w_l^{(x_{il}-x_{jl})^2}}\right)}{\frac{\sum_q\left(\sqrt{\sum_{l=1}^{k} w_l^{(x_{il}-x_{jl})^2}}\right)}{q}} \qquad (2.11)$$

Where:

$\overline{\alpha}$ is the intra cluster coefficient,

q is the number of distances between marginal points of the cluster,

k is the number of attributes to consider when evaluating the cluster,

w_l is the previously defined weight for dimension $l \in [1, k]$, and

$(x_{il} - x_{jl})$ is the difference between two marginal points i and j considering attribute l.

As long as a cluster has more than one object, then $\overline{\alpha} \geq 1$, when a single-object cluster is being analyzed, $\overline{\alpha} = 0$.

$$\overline{\beta} = \frac{\overline{\phi}_{min}}{\overline{\phi}_{avr}} = \frac{\min\limits_{p}\sqrt{\sum_{l=1}^{k} w_l(x_{il}-c_{jl})^2}}{\frac{\sum_p\left(\sqrt{\sum_{l=1}^{k} w_l(x_{il}-c_{jl})^2}\right)}{p}} \qquad (2.12)$$

Where:

$\bar{\beta}$ is the inter cluster coefficient,

p is the number of extremely marginal points,

k is the number of attributes to consider when evaluating the cluster,

w_l is the previously defined weight for dimension $l \in [1, k]$, and

$(x_{il} - c_{jl})$ is the distance between the marginal point i and cluster center j considering attribute l.

Doan and Nguyen (2018) then test different number of clusters in a Fuzzy C-Means-Extended algorithm, evaluating the quality of the clustering by using coefficient $\bar{\gamma}$. The recommended number of clusters is that which minimizes $\bar{\gamma}$, given by (2.13).

$$\bar{\gamma} = \bar{\alpha}_{max} + (1 - \bar{\beta}_{max}) \tag{2.13}$$

Where:

$\bar{\gamma}$ is the coefficient to be minimized,

$\bar{\alpha}_{max}$ is the maximum value of the intra clustering coefficient defined in (2.14), and

$\bar{\beta}_{max}$ is the maximum value of the inter clustering coefficient defined in (2.15).

$$\bar{\alpha}_{max} = \max_{c}(\bar{\alpha}) \tag{2.14}$$

Where:

$\bar{\alpha}_{max}$ is the maximum value of the intra clustering coefficient,

c is the number of clusters, and

$\bar{\alpha}$ is the intra cluster coefficient defined in (2.11).

$$\bar{\beta}_{max} = \max_{c(c-1)}(\bar{\beta}) \tag{2.15}$$

Where:

$\bar{\beta}_{max}$ is the maximum value of the inter clustering coefficient,

c is the number of clusters, and

$\bar{\beta}$ is the inter cluster coefficient defined in (2.12).

2.4.2 Kluster

Looking into more complex algorithms, Estiri, Omran, and Murphy (2018) propose *kluster*, which applies four different methods to random subsets of the dataset, to find each an optimal cluster number. The *kluster* method gives different perspectives on the quality of the clusters, at the same time the sampling approach allows for a more expedite process.

Kluster gives back the mean and mode of the results found by these four methods as a possible optimum number of clusters for the dataset, and can be described in four steps:

1. Retrieve random sample, of size n, from the database. Collected data is given by $(X_1, X_2, \dots, X_n)$.
2. Run the cluster number approximation algorithm w to the retrieved data. The four methods included in w are Bayesian Information Criterion (BIC) (Fraley and Raftery 2002), Calinski and Harabasz index (CAL) (Calinski and Harabasz 1974), Partitioning Around Medoids (PAM) (Kaufman and Rousseeuw 1987), and Affinity Propagation (AP) (Frey and Dueck 2007), such that $w \in (BIC, CAL, PAM, AP)$.

3. Repeat 1 and 2 i times. This results in vector $(k_{w1}, k_{w2}, \dots, k_{wi})$.
4. Finally, using the obtained vector, calculate its mean and mode.

Ünlü and Xanthopoulos (2019) also look into evaluating the number of clusters through multiple methods with "weighted consensus clustering". Here the chosen indices are Silhouette (SH) (Rousseeuw 1987), Calinski and Harabasz (CAL), as well as Davies-Bouldin (DB) (Davies and Bouldin 1979). The ideal number of clusters is such that it maximizes the consensus index given by (2.16).

$$k^* = \underset{k=2,\dots,k_{max}}{\arg\max}\ CI(P_k) \tag{2.16}$$

Where:

k^* is the ideal number of clusters,

k is the number of clusters,

k_{max} is the maximum number of clusters, and

$CI(P_k)$ is the consensus index of the solution P_k, defined in (2.17).

$$CI(P_k) = \sum_{i<j} \psi(P_i, P_j) \tag{2.17}$$

Where:

$CI(P_k)$ is the consensus index of the solution P_k, and

$\psi(P_i, P_j)$ is the similarity between clustering solutions i and j.

Considering that c is both the number of clustering solutions and the number of times to run the algorithms, the solutions are $\mathcal{P} = \{P_k; k = 2, \dots, c + 1\}$. The authors chose Adjusted Rand Index (ARI) (Hubert and Arabic 1985) as a similarity index ψ.

3 Problem Definition

3.1 Assortment Planning and Store Clustering

In retail, the very complex process of deciding which products to sell, and how to sell them, is critical to assure success. Any bad decision can cause unprecedented damage to the company's revenue and image. Assortment planning is the process of selecting and planning products in order to maximize sales and profit, or just general company objectives, for a specified period of time and from which it is expected to get both the size of the starting purchase and the allocation plan for the considered period (Haertle and Albert 2018). It involves decisions regarding the process of buying new products to sell, product retention and elimination of current products (Host and Nilsson 1987).

Oracle's Assortment Planning process, illustrated in Figure 2 and will be further explained ahead, comprises many steps, from selecting which products to sell in a given time frame to the performance analysis of such products during their selling period (Goodman 2016). Cluster Maintenance, which involves both the definition of new clusters and the update of ones already defined is one of the first steps of Assortment Planning, impacting all the following ones.

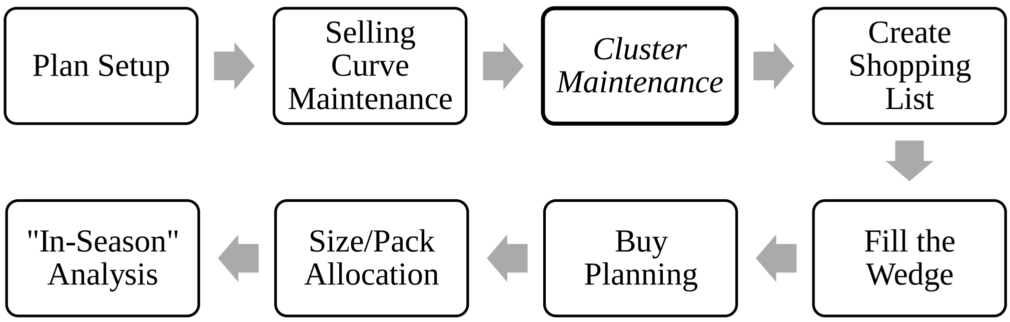

Figure 2 - Assortment Planning process.

In fashion retail, this process is even more important, as new products enter the stores every season, in some cases every week (Garro 2011), and there is no past sales data to analyze. The fashion industry is defined by short product lifecycles, volatile and unpredictable demand patters, high product variety, and long, complex, and low-flexibility supply processes (Şen 2008). Therefore, fashion companies use Assortment Planning to develop a product mix (Haertle and Albert 2018), which specifies when and where a product should be available, at what price and subject to what kind of promotions, in order to meet overall company goals (Goodman 2016).

As explained by Bahng et al (2011), both internal and external components contribute to the decisions made in the assortment planning process. Internal factors include budget, available store space, defined brand image, historic sales, product evaluation – sellability and selling period evaluation -, product cost and markups and inventory; while the demand and characteristics of the target market, available information on fashion trends, competitors' own assortment planning decisions, supplier evaluation, economic conditions and weather information all influence the assortment planning from an external perspective.

Store Clustering represents a vital step in Assortment Planning, as it is part of the process that defines which products will be available, in which quantities, at which time period, and at which price, in each point of commerce (Oracle 2016). It is important to note that Store Clustering is applicable to not only physical stores, but also to any other channel used to sell the products, like a brand website or a multi-brand mobile app (Goodman 2016).

3.2 Oracle's Basic and Advanced Store Clustering

Oracle's Basic Clustering (Goodman 2016) is comprised of four steps, from defining the clustering parameters to approving the final clusters. First, it starts with a revision of the goals defined in the previous Assortment Planning steps, looking into how they relate to the stores, and definition of the attribute weights or the breakpoints that make sense for that specific assortment. These weights can vary within and between periods, which means that, as the market evolves, store clustering can be constantly adapted. After setting which attributes to use in the particular clustering, the clusters are determined. In this step, the user can choose between the two available clustering algorithms – Breakpoint and BaNG (Batch Neural Gas) – and, next, look into the performance of the points-of-commerce to cluster, as well as the performance group they have been assigned to, under the defined clustering method. These performance clusters can then be further divided according to other store attributes. Once the clusters are obtained, further reviews can be done, now looking into how the achieved clusters impact the overall Assortment Plan. Finally, if everything is found to be according to the user's requirements, the point of commerce clusters are approved and the clustering process is done.

Advanced Store Clustering (Oracle 2016) is part of Oracle's Advanced Science Engine Cloud Services solution and it offers the option of looking into what-if scenarios in order to evaluate the performance of different clusters under different circumstances. The cluster recommendation for each scenario is given considering cluster evaluation measures, such as mean square distance. Advanced clustering can take into account the characteristics of the customers, but, in order to provide the best results, it needs data such as point-of-sale data – aggregated by SKU, store and week -, product attributes, location attributes, customer segment profiles, and market hierarchy.

Currently, Oracle's Advanced Store Clustering uses store characteristics as the clustering attributes, given that these characteristics are chosen by experts. The clustering can be done through the Breakpoint or the BaNG algorithms. The Breakpoint algorithm is a hierarchical clustering algorithm, in which sales and profit margins are used to group higher level clusters and, then, user specified attributes further divide these "cluster parent" into lower level clusters. In this algorithm, exemplified in Figure 3, the attributes (for example, climate), and the different values they can take, are the input variables, while the final number of clusters is concluded based on these user-defined breakpoints.

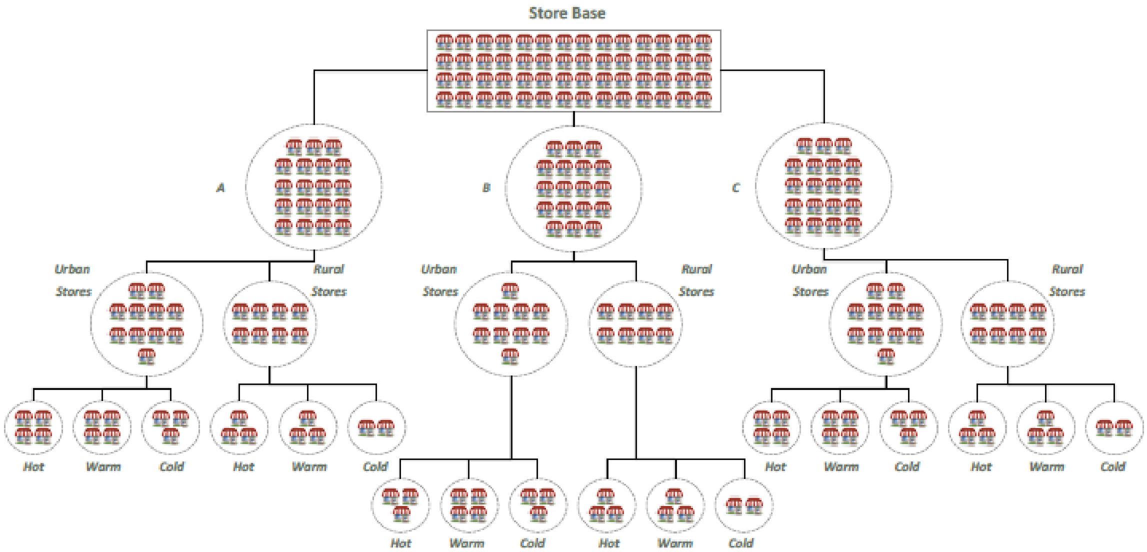

Figure 3 - Hierarchical Clustering through Breakpoint (Oracle 2016).

On the other hand, the BaNG algorithm relies on user specified attributes and their respective weights, as well as the number of clusters. As exemplified in Figure 4, there is only one level of resulting clusters, defined by the attributes to consider, their weight, and their possible values. The quality of the final clusters is, therefore, more influenced by the specialist's inputs. The BaNG algorithm, much like k-means, is related to neural networks and uses crisp assignment of data points to clusters. It has shown, however, to perform better than k-means, converging faster and without initialization problems (Cottrell et al. 2006).

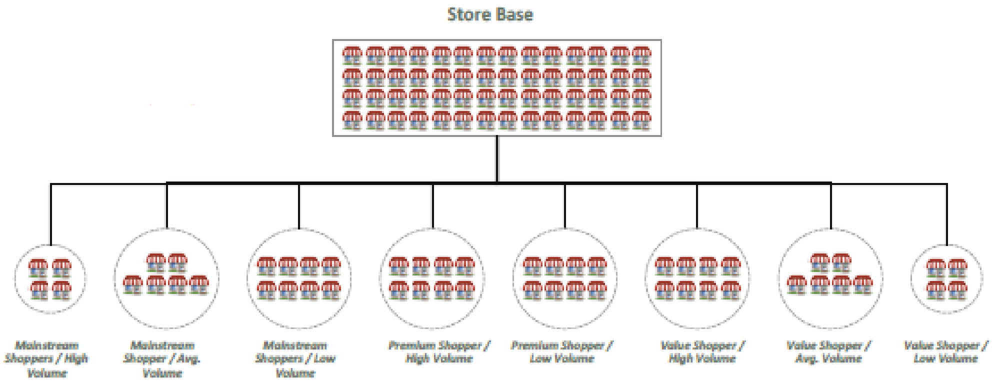

Figure 4 - Weighted Clustering through BaNG (Oracle 2016).

3.3 Influence of Store Clustering

As previously mentioned, the defined clusters will influence the steps that succeed to the Cluster Maintenance. The Fill the Wedge process uses store clusters to assign a specific product assortment to a group of stores, based on the overall assortment goal, as well as store cluster characteristics (Figure 5). Looking into the clusters defined in Figure 4, different products would be assigned to clusters with different types of customers, as it is to be expected that different types of customers will have different needs. At the same time, while some products should be sold by stores with both higher and lower performances, it may be of the interest of the company to only sell some products on high-performance stores.

			Traditional Wedge Store Clusters by Sales Volume			Tailored Wedge Store Clusters by Customer Attributes		
Style	Buy Quantity	Sales Rank	Large Volume	Medium Volume	Small Volume	Urban, High Income	Urban, Middle Income	Suburban, Delivery, Middle Income
Style 1	473	1	X	X	X	X		
Style 5	420	2	X	X	X			X
Style 10	285	3	X	X	X		X	X
Style 3	277	4	X	X	X	X	X	X
Style 6	261	5	X	X	X	X	X	X
Style 2	249	6	X	X	X	X		
Style 9	230	7	X	X		X		
Style 11	220	8	X	X		X	X	
Style 4	179	9	X	X		X	X	X
Style 7	151	10	X				X	X
Style 12	130	11	X			X		
Style 8	102	12	X			X	X	X

Figure 5 - Fill the Wedge (Oracle 2015).

In a traditional wedge the assortment is made considering sales volume, so that stores with more sales would sell all products, while stores with less sales would sell only the higher ranking products, meaning the ones which are expected to sell more. Oracle's Tailored Wedge, however, takes into account characteristics like, for example, customer attributes, so it is important that the clusters that enter this process take into account the needs of their customers, and not necessarily the level of sales the stores have.

The quality of the obtained clusters also influences the Buying Plan and the subsequent Size/Pack Allocation processes, as these take the assortment defined by store cluster in the Fill the Wedge step and define a time phased plan of inventory, receipts and distribution.

Finally, the "In-Season" Analysis looks into the difference between the planned and actual sales, considering data like preferences of each customer segment or how much each channel or store is selling. This means that stores will be analyzed within their defined cluster. Besides being used to make in-season decisions, this process also gives insights on the quality of the overall Assortment Planning, as big deviations from the plan with no explanation expose a poor planning process.

3.4 Characterization of the retailers

Fashion Retailer 1 (FR1) is an international retailer, present in 4 continents and 32 different countries, which distributes through their own stores, franchisers, department stores, and online stores. Although their products are also sold in different online retailers, these sales aren't the focus of the analysis, as each of the online retailers are responsible for all of their distribution and stock control decisions. FR1 has well defined seasons, with spring/summer starting in late November, and its regular season ending in the beginning on May, while the regular fall/winter season run through the rest of the year.

As clearance is used as a way to sell excess stock at the end of the season, its sales depend on the products already available at the stores, and, as such, clearance sales data should not be considered when defining clusters that are expected to reflect the normal demand of a store. Besides, only sales regarding new products are considered for each season, as products from past seasons could be sold as either carry-on items, which would maintain the same price range, and whose predicted demand could be deducted from previous seasons, or as clearance items that were kept in store even after the regular clearance period from its own season. Furthermore, and although these can diverge from store to store and happen all throughout the regular season, promotion sales are considered as a part of regular sales, as these are usually regular occurrences for which the retailer prepares for, and are also recorded together with regular sales in this specific fashion retailer.

The attributes used to define the cluster are the ones predefined by the fashion retailer, in the case of FR1, the available data classifies stores regarding their distribution “channel”, the “country” and “continent” in which they’re located, and their available “space” for each subclass.

Table 1 - Examples of FR1 store attributes.

Store	Channel	Country	Continent	Space Subclass 1 (m^2)	Space Subclass … (m^2)	Space Subclass n (m^2)
store_1	Department Store	Portugal	Europe	0,001	…	0,023
store_2	Franchising	Japan	Asia	1,03	…	0,58

Fashion Retailer 2 (FR2) sells products from different brands, and, currently, has stores only within one country. These stores can either sell products from only one brand, or a combination of them, being that these last stores can either be characterized as general stores, which sell products from many different brands, or combo stores, which only sell two or three brands. The focus of the analysis is one of these specific brands, Brand 1 (B1), which is targeted at a young fashionable segment, meaning it is much more susceptible to trends, and is sold in all the types of stores previously mentioned.

Unlike FR1, FR2 doesn’t have well defined seasons, with many products being sold throughout the whole year, or for specific occasions with no regards to season. This could be explained by the fact that FR2 stores, although all within the same country, serve cities with a large variation of weather conditions between them, some of which with very small local variation: for example, some beach shops sell summer clothing all year round. Fashion Retailer 2 separates its promotional, clearance, and regular sales data, therefore only the regular data will be considered. As products often sell through different seasons without interruption, all regular sales of the product are to be considered.

The available store classification for FR2 is defined as the “region” of the country where it is located, “type of store”, “concept”, “climate”, and “zone”.

The definition of the store clusters is made for each of the product classes, as it is expected that the sales variation of different classes behave differently, for example while jeans may sell similarly in different climates, heavy coats are not expected to have that pattern of demand. Inside product classes, the comparison between stores can be made on different levels, considering the subclass, the product group, the product, or the product and its size.

The consideration of size, although interesting in an environment where the stores serve a huge diversity of customers, has to consider the issue of size consistency. Sizes often vary between products, seasons, and even suppliers. Taking these expected variations into account, as well as the small size variations observed between stores (Figure 6), comparing sales at the size level is not justifiable.

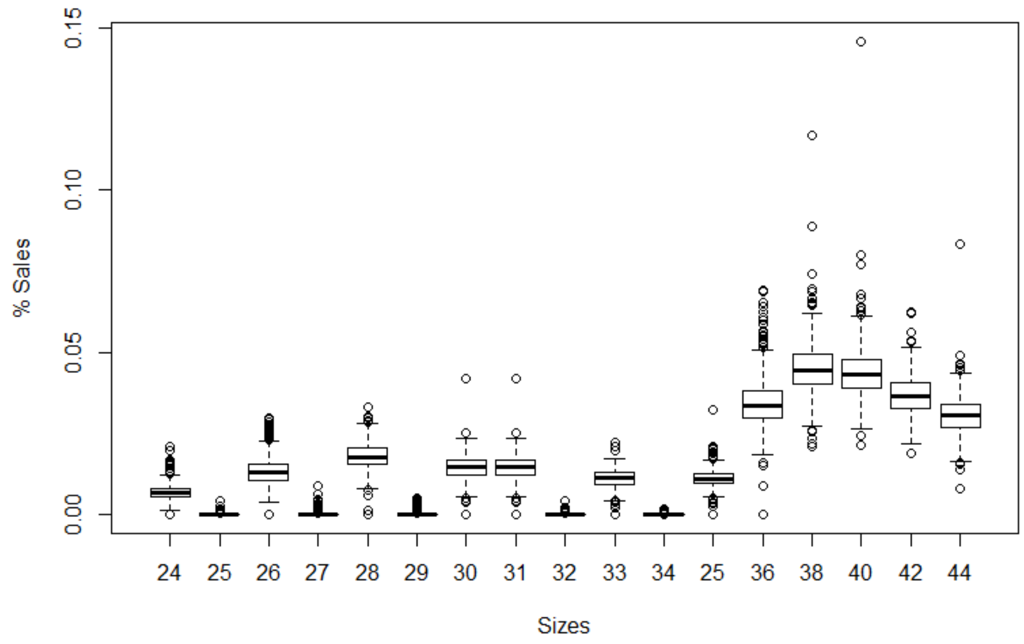

Figure 6 - Boxplots of the variation of the percentage of sales from FR1 stores each size represents.

For both the retailers in analysis, comparing sales units at the product level is, at first glance, the best option, as it takes into account the preference of customers regarding color, fabric, style, and price. Nonetheless, an analysis on the subclass level could also be interesting, as it looks into the way a type of product usually sells, not taking into account the trends that usually affect the choice of specific style. However, information about the buying power and style preferences are lost, as products of very different price ranges, with different styles and colors are aggregated. Regarding sales by product price range, while in FR1 most stores sell products with a similar price range distribution (Figure 7), with some outliers to consider, in FR2 (Figure 8) the percentage of products of each price range varies quite a lot showing that this, unlike size, price range is a variable relevant for store separation.

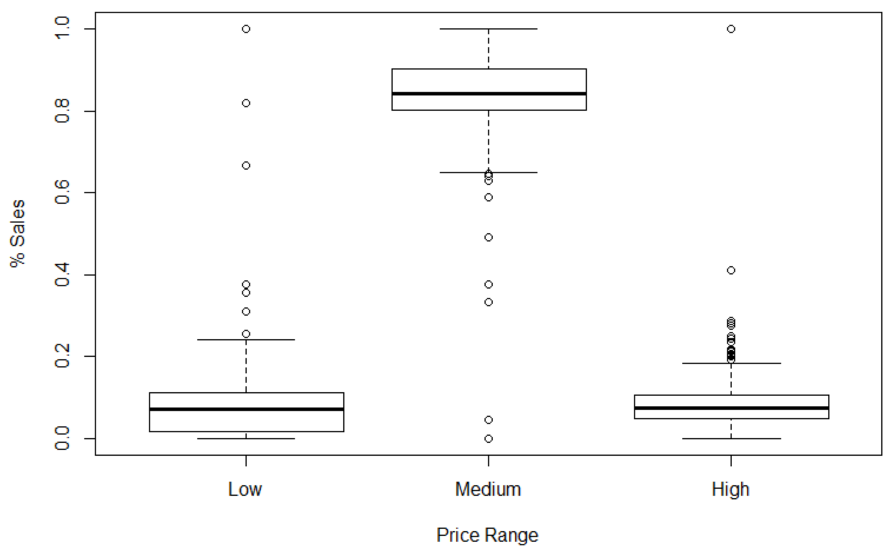

Figure 7 - Boxplots of the variation of the percentage of sales from FR1 stores each Price Range represents.

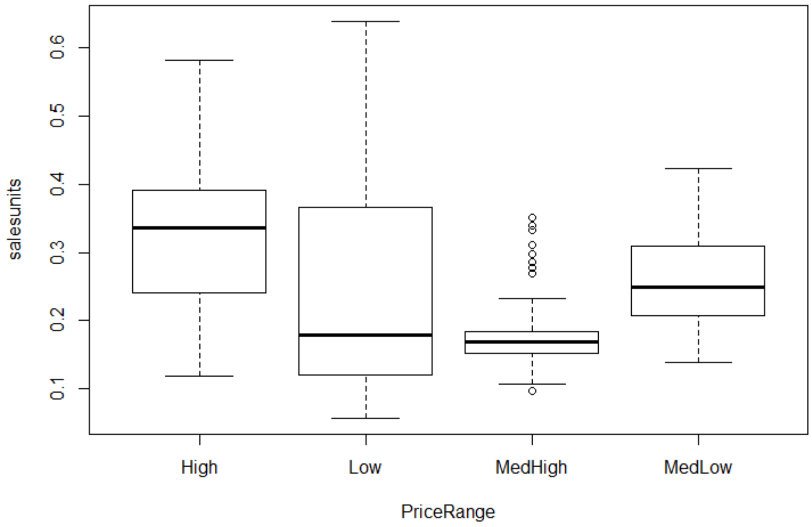

Figure 8 - Boxplots of the variation of the percentage of sales from FR2 stores each Price Range represents.

Considering the different product aggregation levels available (Figure 9), the intermediate level between the product (SKUP) and the "Subclass" is “SKUG”, a group of products, which takes into account variables like style but not color, and, sometimes, may vary in price, but not significantly. The best relationship between detecting interesting differences between stores and their demand, and the advantages of working on a higher product level, such as lower influence of outliers, must be analyzed, taking into account that different retailers could benefit from working at different aggregation levels.

Therefore, sales data is to be analyzed according to three different levels of aggregation: “SKUP”, “SKUG”, and “Subclass”. These levels are all above the “SKU” level that takes size into account, which doesn’t vary widely between stores, and below the “Class” level, for which the clusters are created. The “Family” level, between “Class” and “Subclass”, often provides a similar classification to “Class” with only one additional differentiator, most often considering the “Class” of a product along with its season. Because the season variable is to be analyzed separately, by either considering different clusters for different seasons or by not differentiating between season, this “Family” product level wasn’t considered.

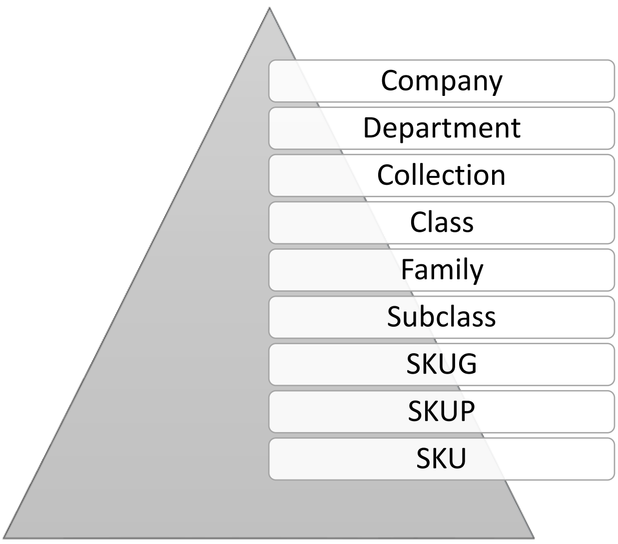

Figure 9 - Pyramid of product levels.

4 Proposed Solution

4.1 Algorithm implementation

In order to test the algorithms, a couple of product classes from each fashion retailer were selected. In the case of FR1, women's t-shirts and dresses were chosen, as these collectively represent over 40% of total regular sales. The available data spans 3 full years and the beginning of a fourth year, relative to the spring/summer season, which is to be used as the final testing data.

In the case of FR2, only data of one of the multiple brands was considered, choosing, again, two different classes, long sleeved shirts and trousers. As the available data is relative to 3 full years, the seconds half of the third year was used as the final test data, using the initial 2,5 years to calculate the attribute weights.

For this analysis, MySQL software is used for data aggregation and simple calculations, while R software is used for deeper analysis and visualization. The data is to be analyzed in sales units by store, for each of the levels of aggregation that is being analyzed. In FR1, the Subclass, SKUG, and SKUP levels are considered, while for FR2 only Subclass and SKUP are analyzed as most product groups (SKUG) are only composed of one product (SKUP). For FR1, as there's a distinction between seasons, this data is also grouped by season for each store.

Before proceeding with the analysis, it is necessary to make sure the data complies with the chosen weighting algorithms. As one of the FR1 store attributes is continuous, "space", and the algorithms deal with categorical data, it is necessary to discretize this variable. For this, and as the variable's values varied quite a lot, from less than 0,5 m^2 to more than 120 m^2 of space available for each class at each store, a clustering of the available space values was performed, using the "NbClust" function, of the "NbClust" R package, which clusters data by first defining the best number of clusters within a range, according to different clustering methods. The number of clusters range was set as [2,10] as it is not desirable to define too many space tiers, and the applied method was k-means, with Euclidian distance.

After this, different preparations are necessary for the Xie-Beni Index Weighting and for the tree-based algorithms (Regularized Trees and Random Forest Filter).

The Xie-Beni Index Weighting algorithm is used to find out the clustering power of different attributes. The data allocated to the attribute weight definition, regarding older sales, is divided into different cluster structures, one for each store attribute being considered, and each cluster is composed of stores with the same attribute value. For example, in the cluster structure used to evaluate the clustering power of the attribute "country", each store is allocated to the cluster of its country, resulting on a cluster of stores for each country.

To evaluate these cluster structures, according to their past sales, the Xie-Beni Index (2.1) of each one is calculated. For this, it is necessary to choose a fuzzifier parameter m, which can vary from 1 to ∞, being that a value of m=2 was chosen, as values between 2 and 3.5 are more stable, and a lower value reflects a higher weight for the distances between data points and cluster centers. Function “cls.scatt.data”, from R package “clv” was used to define each of the cluster centers.

This weighting method results in values between 0 and 1 for each attribute, however they sum is not equal to one. To make sure this happens, the final weight is the relative weight of each attribute, in comparison to all the others.

In order to apply the forest based algorithms, first it is necessary to create initial clusters, taking into account past sales, as these clusters are to be used as output to these learning algorithms. To create these clusters, R package “NbClust” was used, as it evaluates the quality of the clusters for different number of partitions, according to multiple indexes and distance measures. The k-means method was chosen, and the range of the number of clusters was set at [2:20], as having too many clusters would go against the final goal of having only a few store grouping to allocate products to.

With the clusters defined, 3 tree based methods are employed, all of them classification algorithms that take the store attributes as input values, and the previously defined cluster as output.

The Regularized Trees, described in 2.3.4, are obtained through the “RRF” function of the “RRF” R package, where both the input and output variables are selected, and the “mtry” parameter is tuned, with the expected out-of-bag error as the comparison metric, which is to be minimized. The “mtry” parameter corresponds to the number of attributes randomly sampled as candidates at each tree split, and ranges from 1 to the number of attributes. The number of trees in each forest was set to 1000, as it is seen as a sufficient number to train the algorithm, while also maintaining an acceptable computational time. The weight of each attribute is the result of the mean increase of the Gini Index of each variable divided by the sum of all mean increased of the Gini indexes, which can be directly extracted from the “RRF” function.

For the Random Forest Filter, attribute importance can be directly extracted from the “randomForest” function in R, from the R package of the same name. Here, much like for Regularized Trees, the “mtry” parameter has to be tuned, and the number of trees to consider in each forest was set to 1000, for the previously explained reasons. Attribute weight is obtained by normalizing the resulting variable importance, i.e. by dividing each importance value by the sum of all of them.

Finally, attribute weight was also calculated from a sensitivity analysis point of view, taking into account the out-of-bag estimate error result from the “randomForest” function. Attributes were excluded from the input variables one by one, considering that the gain in estimate error when an attribute is not included is expected to be related to this attribute’s importance.

Once weights were defined, they were be used to cluster the stores for which products are going to be allocated in the following season. The clusters were obtained through the neural gas method, using the convex cluster function “cclust”, from the R package with the same name, considering the maximum iterations as 10000. To tune the number of clusters to use for each of the weight groups, the Calinski and Harabasz and the Silhouette (Thinsungnoen et al. 2015) indexes, also used in the *kluster* method described in 2.4.2, were used to measure the quality of the clusters. Both of these indexes are calculated through the function “cluster.stats”, present in the “fcp” R package, and evaluate the relationship between inter and intra cluster variation, with higher values being desirable.

All of these algorithms are implemented using free R Software, making use of free R packages. While the computational time varies between methods all of these can be performed in a matter of hours. Although, for example, the out-of-bag estimate error rate method takes “a”+1 times longer to compute than Random Forest Filter, as it has to run the Random Forest algorithm for each of the “a” store attributes to consider, these calculations can still be performed within a day. This, in addition to the fact the store clustering process is usually performed only once at the beginning of each season, can justify higher computational time costs if it is the case that these more expensive algorithms prove to work better.

4.2 Algorithm evaluation

In order to evaluate the quality of the store clusters resulting from the above described algorithms, they were associated with the real sales data not considered to calculate the weight. By evaluating the clusters taking into account real sales, it is possible to understand if the stores are associated with the right clusters.

The Silhouette index looks exactly at this parameter, with higher values of average silhouette width, which varies between -1 and 1, representing more compact clusters, and less likelihood that a store should belong to a different cluster.

In addition, the Within Sum of Squares (WSS) (Thinsungnoen et al. 2015) gives insights about the dispersion of a cluster, the objective being to choose clusters of stores that sell as similarly as possible. Because these values can only be compared against each other and don't give an account of the quality cluster on their own, they'll be used as a secondary evaluation measure, and only considered to compare methods within each class and aggregation level.

4.3 Results

4.3.1 Fashion Retailer 1

Table 2 shows the evaluation of the clustering results with the weights calculated by each of the models for the "dress" class according to the Silhouette and WSS indexes.

Table 2 - Evaluation of the clusters resulting from each weighting method for class "dress" of FR1.

	SKUP Level		SKUG Level		Subclass Level	
	Silhouette	WSS (*10^3)	Silhouette	WSS (*10^3)	Silhouette	WSS (*10^3)
Random Forest Filter	-0,1778	645	-0,1280	632	-0,0705	<u>8151</u>
OOB estimate error rate	**0,1551**	624	-0,1171	<u>585</u>	**0,2301**	8312
Regularized Trees	-0,0909	656	-0,1731	695	-0,0525	8295
Xie-Beni Weighting	0,0279	<u>584</u>	-0,1133	595	0,0473	8200

The out-of-bag estimate error rate method obtains the highest Silhouette values at both the SKUP and Subclass level, the fact that both these values are positive also indicate an acceptable quality of clustering, as the intra cluster variations are smaller than the variations between clusters. The Xie-Beni Weighting method also produced positive results, although not as high, being that, at the SKUP level, its WSS value is the lowest, showing that the clustering may not be as coherent, but the stores within each cluster are generally more similar to each other.

Clusters on the SKUG level got overall worse Silhouette results across all methods, which indicates grouping at this level is not indicated for the considered class, "dress". On the other hand, the Subclass level resulted in better clusters for all the methods. The higher quality of the Subclass clusters can be explained due to being done at a more aggregated level, where trends don't have such a big influence. The advantage of clustering at the SKUP level, instead of SKUG, could be due to the fact that SKUP takes into account stylistic elements such as color, which might be related to some of the store attributes that reflect different cultures, like the country and continent.

On the SKUP level, the attribute weights (Table 3) derived through the out-of-bag estimate error rate show a great importance given to the "channel" attribute, with around 53,8%, as well as the "country" the store is located in (44,1%), with "continent" explaining the rest of the variance (2,1%).

The presence of both "country" and "continent", which combined make up more than 50% of the total weight, seems to indicate that cultural differences explain the different products people buy from the same product groups.

The high weight of the "channel" attribute could be due to most people preferring a certain type of store when buying this specific class of products, "dress", for example, people who usually make their purchases through e-tailers are likely to continue making their purchases there, instead of going to a department store. At the same time, different groups of people may prefer to shop at different types of store, as it is expected that someone who is specifically looking to buy clothing from this fashion retailer's own brand is more likely to buy at a FR1 own retail store instead of a multi-brand department store.

Table 3 - Attribute weights according to each weighting method for class "dress" of FR1 aggregated at the SKUP level.

	Channel	Country	Continent	Space
Random Forest Filter	31,6%	44,1%	5,0%	19,4%
OOB estimate error rate	**53,8%**	**44,1%**	**2,1%**	**0,0%**
Regularized Trees	25,1%	49,2%	0,0%	25,7%
Xie-Beni Weighting	49,9%	0,0%	50,0%	0,1%

Comparing the weights obtained at the SKUP level, the ones obtained through the SKUG level's best attribute weight method – Xie-Beni weighting –, the main difference is on the value attributed to the "country" and "continent" attributes, where "country" is given a significance of 0% and "continent" represents almost 50% of the clustering division (Table 4). This means the cultural differences are being evaluated at a much higher level, which doesn't translate to such good clustering at the subclass level. These weight values are, however, very similar to the ones obtained through the same Xie-Beni index method for the SKUP level of aggregation, which has an acceptable silhouette value, combined with the lowest WSS, meaning that, again, some information is being lost in comparison to the out-of-bag estimate error rate method, but, at the same time, in this particular case, an aggregation at the SKUP level is more valuable.

Table 4 - Attribute weights according to each weighting method for class "dress" of FR1 aggregated at the SKUG level.

	Channel	Country	Continent	Space
Random Forest Filter	31,3%	44,3%	5,0%	19,4%
OOB estimate error rate	44,9%	24,3%	12,8%	0,0%
Regularized Trees	24,9%	49,2%	0,0%	25,9%
Xie-Beni Weighting	**49,9%**	**0,0%**	**50,0%**	**0,1%**

Although the results for the Xie-Beni weighting method were acceptable, it is important to note that this method doesn't deal well with attributes which have considerably more attribute values, giving them weights closer to zero in all situations. This happens in this case, as the attribute “country” has many more (32) attribute values than the other attributes, which number values range from 4 to 7, and, therefore, is given a weight of 0,0% at all aggregation levels by the Xie-Beni weighting method (Table 3, Table 4, Table 5).

At the subclass level (Table 5), the out-of-bag estimate error rate method resulted in all the weight being put on the “channel” attribute, meaning that the type of store explains the variation in the sales of the subclass of dresses, which is in this specific retailer differentiated by fabric. The fact that variables like style, color, and price range are not taken into account may explain why variables that somehow explain the demography of the store’s customers were not taken into account.

Table 5 - Attribute weights according to each weighting method for class “dress” of FR1 aggregated at the Subclass level.

	Channel	Country	Continent	Space
Random Forest Filter	50,0%	32,1%	2,7%	15,2%
OOB estimate error rate	**100%**	**0,0%**	**0,0%**	**0,0%**
Regularized Trees	47,2%	34,2%	1,3%	17,3%
Xie-Beni Weighting	50,0%	0,0%	49,9%	0,1%

On both SKUP and Subclass level of aggregation, the best attribute weighting gave no importance to the variable “space”, which represents the size of the space available for the product class. This could be because although two stores may have similar spaces designated to sell dresses, the number of dresses from each specific subclass or from each product may vary widely between stores.

For both the best clustering solutions, according to the silhouette index, at the SKUP and Subclass levels, 4 clusters were found to be ideal, while the best clustering result for the SKUG level had stores aggregated into 10 clusters, meaning that it is possible to work with good quality clusters even when working with a lower number of them. Additionally, the Xie-Beni weighting method at the SKUP level, which also produced acceptable results, did so in 9 clusters, a much higher number than the one obtained through the out-of-bag estimate error rate. This could indicate that a smaller number of clusters makes it more likely that each store is allocated to the right cluster, however a bigger number of clusters results in clusters with stores more similar to each other.

It is important to note that a lower number of clusters means a quicker allocation of products to stores, but also a smaller variation between offers at each type of stores, so, if the retailer intends on bigger diversification between types of stores, a different range for possible number of clusters should be defined up front, as some algorithms may result in better suited weights for a higher number of clusters.

As for the t-shirts (Table 6), very distinct results were obtained, which could be explained by the fact that the algorithms work differently with the data available for this class, however the best clustering structures, with the higher Silhouette index value, continue to be at the Subclass level, followed by the SKUP level, and with the SKUG sales producing the worst clusters, this time with a positive, although still low, value.

While for the dresses the out-of-bag estimate error produced the best results, here the Random Forest variable importance method showed good results for all aggregation levels, with positive silhouette values and good WWS results when compared to other methods.

Table 6 - Evaluation of the clusters resulting from each weighting method for class “t-shirt” of FR1.

	SKUP Level		SKUG Level		Subclass Level	
	Silhouette	WSS (*10^3)	Silhouette	WSS (*10^3)	Silhouette	WSS (*10^3)
Random Forest Filter	**0,0909**	1061	0,0314	<u>1078</u>	0,0274	<u>25021</u>
OOB estimate error rate	-0,1555	1014	**0,0639**	1113	-0,1814	27215
Regularized Trees	-0,1052	<u>1000</u>	-0,1435	1079	**0,1326**	27363
Xie-Beni Weighting	-0,1517	1187	-0,1865	1238	-0,4494	37597

At the SKUP level (Table 7), looking at the weights attributed through the random forest method, a higher importance is given to the “channel” attribute yet again (57,6%), with “country” in second place (26,1%), followed by the “space” available (11,1%) and, finally, the “continent” (5,1%). The high “channel” and “country” weights can be explained in a similar manner as has been done for similarly high values found in the “dress” class. Here, however, “space” is given a rather relevant weight, which could be due to different allocation strategies, where, for example, bigger stores get a bigger variety of t-shirts.

Table 7 - Attribute weights according to each weighting method for class “t-shirt” of FR1 aggregated at the SKUP level.

	Channel	Country	Continent	Space
Random Forest Filter	**57,6%**	**26,1%**	**5,1%**	**11,1%**
OOB estimate error rate	93,1%	0,0%	6,9%	0,0%
Regularized Trees	20,9%	56,9%	0,0%	22,2%
Xie-Beni Weighting	0,0%	0,0%	0,0%	100%

Looking into the weights at the SKUG level of aggregation (Table 8) defined through the out-of-bag estimate error rate, no weight is given to the “space” attribute yet again, with most of the importance being given to the store “channel” at 74,7%, followed by “country” (15,4%), as it happens at the SKUP level. The Random Forest Filter method, however, does attribute weight to the “space” attribute at this same level, at 15,4%. This makes it clear how differently both these methods allocate weight, even if both of them are based on the random forest algorithm. Both of them do give more importance to the “channel” and “country” variables, with 46,4% and 35,7% assigned to them, respectively, through the Random Forest Filter method.

Table 8 - Attribute weights according to each weighting method for class “t-shirt” of FR1 aggregated at the SKUG level.

	Channel	Country	Continent	Space
Random Forest Filter	46,3%	35,7%	2,6%	15,4%
OOB estimate error rate	**74,7%**	**15,4%**	**9,9%**	**0,0%**
Regularized Trees	21,5%	58,5%	0,0%	20,0%
Xie-Beni Weighting	0,0%	0,0%	50,0%	50,0%

Finally, at the Subclass level of aggregation (Table 9), the regularized trees method resulted in the highest Silhouette value, with the highest weights being allocated to the "channel" and "country" attributes yet again. Unlike previous methods, however, the "continent" the store is located in was assigned a weight of 0%, while the available "space" was given an importance of 15,8%. The Random Forest Filter algorithm allocated similar weights to the attributes at this level of aggregation, however the "continent" of the store was attributed some weight, at 3,9%. These small differences, however, resulted in different optimal number of clusters, 6 for the Random Forest Filter method and 4 for the regularized trees. As observed previously, now working with similar value weights, higher numbers of clusters lead to the smallest WWS values, but also to lower silhouettes.

Table 9 - Attribute weights according to each weighting method for class "t-shirt" of FR1 aggregated at the Subclass level.

	Channel	Country	Continent	Space
Random Forest Filter	55,5%	28,3%	3,9%	12,3%
OOB estimate error rate	73,1%	14,7%	10,6%	1,6%
Regularized Trees	**47,6%**	**36,6%**	**0,0%**	**15,8%**
Xie-Beni Weighting	0,1%	0,0%	49,9%	50,0%

4.3.2 Fashion Retailer 2

One of the classes chosen to analyze the methods in FR2 was "long sleeved blouses". Table 10 shows the evaluation indexes of the different weighting methods for sales data aggregated at the SKUP and Subclass level. The SKUG level wasn't taken into account as these are almost always composed by only one product.

Table 10 - Evaluation of the clusters resulting from each weighting method for class "long sleeved blouses" of FR2.

	SKUP Level		Subclass Level	
	Silhouette	WSS (*10^3)	Silhouette	WSS (*10^3)
Random Forest Filter	-0,0422	2332	-0,1117	54819
OOB estimate error rate	-0,1288	<u>2198</u>	-0,1109	55565
Regularized Trees	-0,0718	2258	-0,1955	<u>54608</u>
Xie-Beni Weighting	-0,0087	2390	-0,1921	59038

It is clear that none of the method proved to be successful, as none of the Silhouette values are positive. Some of them, specifically the Xie-Beni weighting and the Regularized Trees methods at the SKUP aggregation level. At the Subclass level, however, all the Silhouette levels are quite low, showing none of the methods provided were able to assign quality weights.

Considering a SKUP aggregation level, the weights obtained (Table 11) from the Xie-Beni method show that almost no importance is given to the "weather" attribute (0,0013%), while all the other attributes, "region", "type" of store, store "concept", and "zone", share similar weights, at about 25%. The Random Forest Filter method, second best according to the Silhouette index, but with a smaller WSS value, attributes a relevant level of importance to all the store attributes, being that "weather" has the 2nd highest weight (23,1%), only beaten by "region", at 25,8%. The lowest weight value was assigned to "concept" (11,2%).

While it created problems with FR1 by allocating values close to 0 for attributes with many possible attribute values, the Xie-Beni weighting method did not have this drawback with this fashion retailer. Although "weather" is given a very small value, number of different values per attribute for FR2 has much smaller variation than those for FR1, with the number of attribute values ranging between 3 and 8, with "weather" having 7 possible attribute values.

Table 11 - Attribute weights according to each weighting method for class "long sleeved blouses" of FR2 aggregated at the SKUP level.

	Region	Type	Concept	Weather	Zone
Random Forest Filter	25,8%	17,7%	11,2%	23,1%	22,2%
OOB estimate error rate	0,0%	0,0%	0,0%	100%	0,0%
Regularized Trees	29,1%	17,3%	0,0%	28,0%	25,6%
Xie-Beni Weighting	**25,0%**	**25,0%**	**25,0%**	**0,0%**	**25,0%**

Because 3 was the number of clusters found to work better with the weights allocated through these methods, the variation of the Silhouette and WSS indexes depends only on the weights attributed.

The "trousers" class was also selected to evaluate the methods for FR2 (Table 12), and once again none of the methods was found to produce reliable clusters. In this class, however, the Random Forest Filter produced the best clusters, followed by the Xie-Beni index weighting, showing that these two algorithms produce the better quality clusters for this retailer.

Here, however, the number of clusters found to work better for each of them is quite different, with 4 clusters for the Random Forest Filter method, and 8 for the Xie-Beni weighting. Although the number of clusters is double for the latter, the WSS value doesn't change dramatically, meaning that using a method that makes use of a higher number of clusters doesn't bring much reward in this particular class. This could be explained by the fact that, unlike the previously considered items, sales levels of most models of "trousers" are not so dependent on fast changing trends or environment characteristics, like weather.

Table 12 - Evaluation of the clusters resulting from each weighting method for class "trousers" of FR2.

	SKUP Level		Subclass Level	
	Silhouette	WSS (*10^6)	Silhouette	WSS (*10^6)
Random Forest Filter	-0,0927	677	-0,2101	2127
OOB estimate error rate	-0,1808	653	-0,4486	<u>1931</u>
Regularized Trees	-0,4309	<u>577</u>	-0,0972	2231
Xie-Beni Weighting	-0,1502	665	-0,1348	2277

Similarly to what happened with the "long sleeved blouses" class, the Random Forest Filter method resulted in relevant weights for all variables (Table 13), although with some big differences, for example, on the "concept" attribute, where its importance dropped to 4,5%. At the same time, "weather" and "region" were, again, attributed the highest weights, with 27,2% each, followed by "zone" (24,3%) and "type" of store (16,7%).

The Xie-Beni Weighting also presents similar weights to those found in the previous class, with "weather" having even littler impact on the clustering with a weight of 0,0002%, while all the other variables have weights close to 25%.

On the other hand, the Silhouette values at the Subclass level of aggregation were slightly better for the "trousers" subclass, using the Regularized Trees algorithm. This algorithm favors attributes such as “weather”, with a weight of 32%, and “region” (27,2%), while the store “concept” seems is given a very low importance (0,1%). Although it has a higher value of the Silhouette index, this clustering also has a high WSS, meaning that clustering at this level is not beneficial.

Table 13 - Attribute weights according to each weighting method for class “trousers” of FR2 aggregated at the SKUP level.

	Region	Type	Concept	Weather	Zone
Random Forest Filter	27,2%	16,7%	4,5%	27,2%	24,3%
OOB estimate error rate	0,0%	7,1%	14,3%	78,6%	0,0%
Regularized Trees	27,3%	17,3%	0,0%	31,0%	24,6%
Xie-Beni Weighting	**25,0%**	**25,0%**	**25,0%**	**0,0%**	**25,0%**

For both the considered classes, the out-of-bag estimate error obtained fairly bad results, contrary to what happened for FR1. The weights attributed by this method to the FR2 classes often had multiple attributes be given 0 importance, which didn’t happen in FR1. This method looks at the way eliminating an attribute from the input values affects the correct identification of the store’s cluster, meaning attributes whose information about the store’s sales can be obtained through another attribute, or a combination of them, are often discarded. If two attributes give very similar information regarding store sales, then they’ll both be dismissed, even if they should be accounted for in the clustering process. Taking this into account, it is fair to say that the information within the attribute “region”, which got weighted at 0% for all the out-of-bag estimate error algorithm attempts for this retailer, can also be explained by other attributes, however these were also being given weight of 0% (store “type” and “zone”).

This resemblance between the attribute weights attributed to classes at the SKUP level seem to indicate that these classes sell similarly according to the defined attributes. This doesn’t mean, however that the same stores sell these products in a similar way, just that their sales are impacted on a similar scale by each of the attributes.

5 Conclusion and future work

5.1 Conclusions

For FR1, it is clear that, at all levels, the attribute “channel”, whether it is a regular brand-owned retail store, an e-tailer, a department store, or a franchise, is the most important attribute, followed by the “country” the store is located in. This would be expected, as these two variables can be used to break down the demography of the different types of customers the brand serves.

Different algorithms were found to work better for different product classes. This could be because the types of sales these classes entail are quite different. Both classes, however, presented some similarities regarding the quality of the clusters at the different levels. The Subclass level aggregation resulted in better quality clusters, however it is important to take into account this level of aggregation doesn’t regard for variables like style and price range. The SKUP level, however, represents the biggest amount of information, and produced quality clusters nonetheless, especially when compared with the SKUG aggregation level, with less information, that resulted in worst clusters. Therefore, for this fashion retailer, working at the SKUP level of aggregation, taking into account customer preferences for fabrics, styles, colors, and price ranges, is the better option.

In the case of FR2, the store attribute “region” was found to be the most relevant and, although the obtained clusters for this retailer were of lower quality, the Xie-Beni weighting and Random Forest Filter methods obtained the best results for the considered classes. It is important to note, however, that these algorithms evaluated the weight for these classes in a very similar manner, meaning that it could be the case that these two classes are just similar enough in their variation according to the considered attributes.

The different weights each of these algorithms allocates to the same attributes, taking into account the same product class, could be explained through the fact that some attributes may qualify customer groups in a similar manner, for example, the attribute “concept”, which includes values like “beach” stores, may give similar information about the expected sales as “weather”. This overlap in the variation explained by each of the attributes can result in one being favored over the other in certain algorithms.

In this retailer, the SKUP level of aggregation also proved to be the best option, with rather poor quality clustering coming from a Subclass aggregation.

As an algorithm wasn’t found to be clearly better when applied to FR1’s data and the similarities between the results found for each of FR2’s product classes could be due to these being coincidentally similar, the ideal scenario should consider to have all algorithms be tested for all the classes, settling for the one which produces the more cohesive clusters, given a certain number cluster range the retailer intends to work with.

As the clustering process usually only happens once every season, it is viable to always adapt the algorithms to both the new company needs and the most recent sales data.

Nonetheless, the Xie-Beni weighting method shouldn't be considered when the number of attribute values varies widely between attributes, like it is the case in Fashion Retailer 1, where the attribute "country" always has a low weight, due to its big number of possible values when compared with other attributes.

Besides this, the OOB estimate error rate doesn't work very well when two or more attributes reflect a very similar information gain, as it tends to give both of them very low weights, even if they are relevant to the clustering process separately.

This results in the Random Forest Filter and Regularized Trees being the most reliable methods, which could fit any type of retailer. However, OOB estimate error rate did work very well for the FR1, probably due to it not considering attributes that reflect on very similar variations on the data.

Finally, regarding the number of clusters, for both retailers, a higher number of clusters means a smaller Silhouette index, where it is more likely some stores are allocated to the wrong cluster, but also a smaller WSS, meaning the stores within a cluster are more similar to each other.

Given the results obtained in this project, across all considered aggregation levels, a smaller number of clusters – ranging from 3 to 6 – is recommended, however retailers may want to offer very different things in many stores, and for this diversification goal, a higher number of clusters has to be set. Thus, it should remain a managerial decision to understand the ideal range of number of clusters to consider, being that this range should be inputted up front, so as to take it into account when defining the initial clusters from which the weights are to be deduced from.

5.2 Future Work

As the final number of clusters can vary between weighting methods and has a big impact on the final results, future work could be focused on understanding the number of clusters that would better work for certain managerial goals, such as sales growth or targeting new customers through a higher variety of products across stores.

Another issue is to realize if, instead of different retailers, different classes of products can be better clustered according to different methods. Different typologies of products can have very different types of data, ranging from a small number of products, spread throughout all stores, that sell a large number of units, to a big number of products, each of them available in a small number of stores, which sell very few units.

In order to take into account recent developments in buying patterns, considering past seasons to have a differently weighted relevance in the initial clustering that will lead to the attribute weights could also lead to interesting results, as it is allows for some sales patterns to emerge, as stores' markets may shift.

Also looking into the constant development of shopping trends, and as mentioned in many decision making variable selection process studies, the inclusion of an expert opinion variable in the weight allocation methodology may be beneficial. Instead of the current practice of having the whole process rely on experts, their opinion could just count as an add-on to the statistical and machine learning analysis of previous data.

Besides this, all the calculations made throughout this project rely on sales data, meaning they're dependent on previous managerial decisions, as two stores can serve customers with rather similar needs, however if the same products aren't allocated to these stores, this connection is harder to find and the weight of the attributes they have in common isn't going to be altered. Taking into consideration as an input to the attribute weighting process the previous decisions that lead to the current sales data, as well as how sales shifted as clusters were altered, could also lead to better results.

Finally, and because allocating products to stores isn’t only a fast fashion problem, these methods could be applied to other retail areas, either ones subject to constant quick changes similar to what is felt in the fashion industry, such as the electronics market, or others that seem to change in slower ways and face different challenges, like grocers.

References

Bahng, Youngjin, and Doris H. Kincade. 2014. "Retail Buyer Segmentation Based on the Use of Assortment Decision Factors." *Journal of Retailing and Consumer Services* 21 (4): 643–52. https://doi.org/10.1016/j.jretconser.2013.12.004.

Bahng, Youngjin, Doris H Kincade, Luann R Gaskill, Joann M Emmel, and Patti J Fisher. 2011. "Developing a Retail Buying Model Based on the Use of Assortment Decision Factors." Virginia Polytechnic Institute and State University. http://hdl.handle.net/10919/77107.

Bai, Liang, Jiye Liang, Chuangyin Dang, and Fuyuan Cao. 2011. "A Novel Attribute Weighting Algorithm for Clustering High-Dimensional Categorical Data." *Pattern Recognition* 44 (12): 2843–61. https://doi.org/10.1016/j.patcog.2011.04.024.

Breiman, Leo. 2001a. "Random Forest." *Machine Learning* 45 (1): 5–32. https://doi.org/10.1023/A:1010933404324.

———. 2001b. "Random Forest." *Machine Learning* 45 (1): 5–32. https://doi.org/10.1023/A:1010933404324.

Calinski, T., and J. Harabasz. 1974. "A Dendrite Method for Cluster Analysis." *Communications in Statistics - Theory and Methods* 3 (1): 1–27. https://doi.org/10.1080/03610927408827101.

Chen, Chun-Bao, and Li-Ya Wang. 2006. "Rough Set-Based Clustering with Refinement Using Shannon's Entropy Theory." *Computers & Mathematics with Applications* 52 (10–11): 1563–76. https://doi.org/10.1016/j.camwa.2006.03.033.

Cottrell, Marie, Barbara Hammer, Alexander Hasenfuß, and Thomas Villmann. 2006. "Batch and Median Neural Gas." *Neural Networks* 19 (6–7): 762–71. https://doi.org/10.1016/j.neunet.2006.05.018.

Davies, David L., and Donald W. Bouldin. 1979. "A Cluster Separation Measure." *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-1 (2): 224–27. https://doi.org/10.1109/TPAMI.1979.4766909.

Doan, Huan, and Dinh Nguyen. 2018. "A Method for Finding the Appropriate Number of Clusters." *The International Arab Journal of Information Technology* 15 (4): 675–82. https://iajit.org/PDF/July 2018, No. 4/9993.pdf.

Doğan, Sevgi Zeynep, David Arditi, and H Murat Günaydin. 2008a. "Using Decision Trees for Determining Attribute Weights in a Case-Based Model of Early Cost Prediction." *Journal of Construction Engineering and Management* 134 (2): 146–52. https://doi.org/10.1061/(ASCE)0733-9364(2008)134:2(146).

———. 2008b. "Using Decision Trees for Determining Attribute Weights in a Case-Based Model of Early Cost Prediction." *Journal of Construction Engineering and Management* 134 (2): 146–52. https://doi.org/10.1061/(ASCE)0733-9364(2008)134:2(146).

ebp Global. 2018. “Machine Learning Applied to Store and Account Clustering.” 2018. https://ebp-global.com/?p=1511.

Estiri, Hossein, Behzad Abounia Omran, and Shawn N. Murphy. 2018. “Kluster: An Efficient Scalable Procedure for Approximating the Number of Clusters in Unsupervised Learning.” *Big Data Research* 13 (September): 38–51. https://doi.org/10.1016/j.bdr.2018.05.003.

Fraley, Chris, and Adrian E Raftery. 2002. “Model-Based Clustering, Discriminant Analysis, and Density Estimation.” *Journal of the American Statistical Association* 97 (458): 611–31. https://doi.org/10.1198/016214502760047131.

Freund, Yoav, and Robert E. Schapire. 1996. “Experiments with a New Boosting Algorithm.” In *ICML’96 Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, 148–56. Morgan Kaufmann Publishers. https://dl.acm.org/citation.cfm?id=3091715.

Frey, Brendan J, and Delbert Dueck. 2007. “Clustering by Passing Messages Between Data Points.” *Science* 315 (5814): 972–76. https://doi.org/10.1126/science.1136800.

Garro, Andres. 2011. “New Product Demand Forecasting and Distribution Optimization: A Case Study at Zara.” http://hdl.handle.net/1721.1/66072.

Giggins, Helen, and Ljiljana Brankovic. 2012. “VICUS - A Noise Addition Technique for Categorical Data.” In *Proceedings of the Tenth Australasian Data Mining Conference*, 134:139–48. https://doi.org/10.1016/j.chemosphere.2014.04.074.

Goodman, Bernadette. 2016. “Oracle® Retail Assortment Planning User Guide for the RPAS Fusion Client.” *Oracle AP User Guide*.

Guyon, Isabelle, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. 2002. “Gene Selection for Cancer Classification Using Support Vector Machines.” *Machine Learning* 46: 389–422. https://doi.org/10.1023/A:1012487302797.

Haertle, Anne, and Courtney Albert. 2018. “Fashion vs. Basic Assortment Planning | Retail Consultants, Retail Strategy, Retail Thought Leadership.” 2018. http://www.parkeravery.com/pov_Fashion_vs_Basic_Assortment_Planning.html.

He, Zengyou, Xiaofei Xu, and Shengchun Deng. 2011. “Attribute Value Weighting in K-Modes Clustering.” *Expert Systems with Applications* 38 (12): 15365–69. https://doi.org/10.1016/j.eswa.2011.06.027.

Host, Viggo, and Jerker Nilsson. 1987. *Reseller Assortment Decision Criteria*. Aarhus University Press.

Houtao Deng, and George Runger. 2012a. “Feature Selection via Regularized Trees.” In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE. https://doi.org/10.1109/IJCNN.2012.6252640.

———. 2012b. “Feature Selection via Regularized Trees.” In *The 2012 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE. https://doi.org/10.1109/IJCNN.2012.6252640.

Huang, J.Z., M.K. Ng, Hongqiang Rong, and Zichen Li. 2005. “Automated Variable Weighting in K-Means Type Clustering.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27 (5): 657–68. https://doi.org/10.1109/TPAMI.2005.95.

Hubert, Lawrence, and Phipps Arabic. 1985. “Comparing Partitions.” *Journal of Classification*. Vol. 2. https://link.springer.com/content/pdf/10.1007%2FBF01908075.pdf.

Jia, Hongjie, Shifei Ding, Hong Zhu, Fulin Wu, and Lina Bao. 2013. “A Feature Weighted Spectral Clustering Algorithm Based on Knowledge Entropy.” *Journal of Software* 8 (5):

1101–8. https://doi.org/10.4304/jsw.8.5.1101-1108.

Kargari, Mehrdad, and Mohammad Mehdi Sepehri. 2012. “Stores Clustering Using a Data Mining Approach for Distributing Automotive Spare-Parts to Reduce Transportation Costs.” *Expert Systems with Applications* 39 (5): 4740–48. https://doi.org/10.1016/j.eswa.2011.09.121.

Kaufman, L, and P J Rousseeuw. 1987. “Clustering by Means of Medoids.” *Statistical Data Analysis Based on the L 1-Norm and Related Methods*, 405–16. https://wis.kuleuven.be/stat/robust/papers/publications-1987/kaufmanrousseeuw-clusteringbymedoids-l1norm-1987.pdf.

Li, Jundong, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P Trevino, Jiliang Tang, and Huan Liu. 2017. “Feature Selection.” *ACM Computing Surveys* 50 (6): 1–45. https://doi.org/10.1145/3136625.

Li, Ming, and Xiao-Feng Zhang. 2005. “Knowledge Entropy in Rough Set Theory.” In *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No.04EX826)*, 4:1408–12. IEEE. https://doi.org/10.1109/ICMLC.2004.1381994.

Makgopa, Sipho Selatole. 2018. “The Importance of Store Attributes on University Students‘ Clothing Store Selection.” *Journal of Business & Retail Management Research* 12 (04): 134–42. https://doi.org/10.24052/JBRMR/V12IS04/ART-14.

Miao, D. Q., and J. Wang. 1998. “On the Relationships Between Information Entropy and Roughness of Knowledge in Rough Set Theory.” *Pattern Recognition and Artificial Intelligence* 11 (1): 34–40.

Modha, Dharmendra S., and W. Scott Spangler. 2003. “Feature Weighting on K-Means Clustering.” *Machine Learning* 52 (3): 217–37. https://doi.org/10.1023/A:1024016609528.

Moye, Letecia Nicole. 2000. “Influence of Shopping Orientations, Selected Environmental Dimensions with Apparel Shopping Scenarios, and Attitude on Store Patronage for Female Consumers.” Virginia Polytechnic Institute and State University. http://hdl.handle.net/10919/11249.

Oner, Sultan Ceren, and Başar Oztaysi. 2018. “An Interval Type 2 Hesitant Fuzzy MCDM Approach and a Fuzzy c Means Clustering for Retailer Clustering.” *Soft Computing* 22 (15): 4971–87. https://doi.org/10.1007/s00500-018-3191-0.

Oracle. 2015. “Build the Wedge.” Oracle® Retail Assortment Planning User Guide for the RPAS Fusion Client. 2015. https://docs.oracle.com/cd/E75762_01/assortplan/pdf/141/html/assortplan_fcug/Output/ap-ug-assortment-wedge.htm.

———. 2016. “Advanced Store Clustering.” Oracle® Retail Advanced Science Engine Cloud Services User Guide Release 15.0. 2016. https://docs.oracle.com/cd/E76441_01/orase/pdf/orase/150/html/user_guide/clustering.htm.

Pawlak, Zdzis?aw. 1982. “Rough Sets.” *International Journal of Computer & Information Sciences* 11 (5): 341–56. https://doi.org/10.1007/BF01001956.

Pundir, Sneh Lata, and Amrita. 2013a. “Feature Selection Using Random Forest in Intrusion Detection.” *International Journal of Advances in Engineering & Technology* 6 (3): 1319–24.

———. 2013b. “Feature Selection Using Random Forest in Intrusion Detection.” *International*

Journal of Advances in Engineering & Technology 6 (3): 1319–24.

Quinlan, John Ross. 1993. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers. https://dl.acm.org/citation.cfm?id=152181.

Rahman, Md Anisur, and Md Zahidul Islam. 2015. "AWST: A Novel Attribute Weight Selection Technique for Data Clustering." In *Proceedings of the Thirteenth Australasian Data Mining Conference (AusDM 15)*, 51–58. Sidney, Australia: CRPIT. http://crpit.com/confpapers/CRPITV168Rahman.pdf.

Revathy, R, and R. Lawrance. 2017a. "Classifying Crop Pest Data Using C4.5 Algorithm." In *2017 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, 1–6. IEEE. https://doi.org/10.1109/ITCOSP.2017.8303122.

———. 2017b. "Classifying Crop Pest Data Using C4.5 Algorithm." In *2017 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)*, 1–6. IEEE. https://doi.org/10.1109/ITCOSP.2017.8303122.

Rosario, F. S., and K. Thangadurai. 2015. "RELIEF: Feature Selection Approach." *International Journal of Innovative Research & Development* 4 (11).

Rousseeuw, Peter J. 1987. "Silhouettes: A Graphical Aid to the Interpretation and Validation of Cluster Analysis." *Journal of Computational and Applied Mathematics* 20 (November): 53–65. https://doi.org/10.1016/0377-0427(87)90125-7.

Ruspini, Enrique H. 1969. "A New Approach to Clustering." *Information and Control* 15 (1): 22–32. https://doi.org/10.1016/S0019-9958(69)90591-9.

Sahami, Mehran, Susan Dumais, David Heckerman, and Eric Horvitz. 1998. "A Bayesian Approach to Filtering Junk E-Mail." www.aaai.org.

Şen, Alper. 2008. "The US Fashion Industry: A Supply Chain Review." *International Journal of Production Economics* 114 (2): 571–93. https://doi.org/10.1016/j.ijpe.2007.05.022.

Steinley, Douglas, and Michael J Brusco. 2008. "A New Variable Weighting and Selection Procedure for K -Means Cluster Analysis." *Multivariate Behavioral Research* 43 (1): 77–108. https://doi.org/10.1080/00273170701836695.

Thinsungnoen, Tippaya, Nuntawut Kaoungku, Pongsakorn Durongdumronchai, Kittisak Kerdprasop, and Nittaya Kerdprasop. 2015. "The Clustering Validity with Silhouette and Sum of Squared Errors." In , 44–51. https://doi.org/10.12792/iciae2015.012.

Thompson, Kim Helen, Debbie Ellis, Sanjay Soni, and Samantha Paterson. 2018. "Attributes Influencing Clothing Store Choice for an Emerging Market's Generation Y Twixter Customers." *The International Review of Retail, Distribution and Consumer Research* 28 (2): 157–73. https://doi.org/10.1080/09593969.2017.1357647.

Ünlü, Ramazan, and Petros Xanthopoulos. 2019. "Estimating the Number of Clusters in a Dataset via Consensus Clustering." *Expert Systems with Applications* 125 (July): 33–39. https://doi.org/10.1016/j.eswa.2019.01.074.

Xie, X.L., and G. Beni. 1991. "A Validity Measure for Fuzzy Clustering." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13 (8): 841–47. https://doi.org/10.1109/34.85677.