# DEASort: assigning items with Data Envelopment Analysis in ABC classes

DOI: 10.1016/j.ijpe.2018.02.007

Link: http://linkinghub.elsevier.com/retrieve/pii/S0925527318300963

## Abstract

Multi-criteria inventory classification groups similar items in order to facilitate their management. Data envelopment analysis (DEA) and its many variants have been used extensively for this purpose. However, DEA provides only a ranking and classes are often constructed arbitrarily with percentages. This paper introduces DEASort, a variant of DEA aimed at sorting problems. In order to avoid unrealistic classification, expertise of decision-makers is incorporated, providing typical examples of items for each class and giving the weights of the criteria with Analytic Hierarchy Process (AHP). This information bounds the possible weights and is added as a constraint in the model. DEASort application is illustrated using a real case study of a company managing warehouses stocking spare parts.

**Keywords:** Inventory, Data Envelopment Analysis, DEA, AHP, Sorting

## 1. Introduction

In an organisation, even one of moderate size, there may be thousands of inventory stock keeping units (SKUs) that have to be held in a warehouse. As the size of the inventory increases, controlling the items requires time and additional expenditure, and thus the use of optimised inventory management would lead to significant savings (van Kampen et al., 2012). As production and inventory policies are influenced by the characteristics of the product, items or SKUs, can be ordered according to their importance (Mohamadghasemi and Hadi-Vencheh, 2011), which enables companies to make decisions on production strategies, inventory management and customer service for the whole class instead for each item separately. Inventory classification using ABC analysis is widely applied by organisations as it is simple to understand, easy to use, and often based on only one criterion. However, a single classifying criterion such as annual usage value cannot generally represent the whole criticality of an item (see Section 2). In addition to this criterion, others such as lead time, criticality, commonality, obsolescence, durability, perishability and inventory cost should also be considered (Rezaei and Dowlatshahi, 2010). To solve this multi-criteria inventory classification problem, items are first ranked by importance using a multi-criteria ranking method such as the analytic hierarchy process (AHP) (Saaty, 1980), ELECTRE (Roy, 1978) or data envelopment analysis (DEA) (Charnes et al., 1978), with the Pareto principle (Dickie, 1951) then applied to assign items into classes. The latter classification method, which is based only on a percentage, can be misleading, as an item can be assigned to, for instance, class A only to satisfy the proportionality of 20%. This issue has led to the recent emergence of studies applying multi-criteria sorting methods that define classes a priori (Chen et al., 2008; Lolli et al., 2014). However, DEA does not yet have an associated a priori sorting technique. This paper presents DEASort, an extension of DEA aimed at sorting items into ordered classes. This method makes use of information provided by managers; AHP is used to elicit the weights of the criteria, with the possible range defined by the group of experts added as a constraint in the model.

The remainder of the paper is organised as follows: Section 2 briefly reviews the main contributions in the literature regarding ABC clustering. Section 3 presents the new DEASort methodology. Section 4 illustrates the application and feasibility of DEASort using a real case study. It also measures the DEASort robustness by varying the number of reference items. It compares the DEASort with other classification approaches and the DEASort and the ABC classification from a cost perspective. Section 5 concludes the paper.

## 2. Literature review

Traditionally, ABC analysis divides items into three classes: A (very important), B (moderately) and C (least important), based on the Pareto principle (Dickie, 1951). Class A items are very few in number (10%), but constitute a relatively large amount of annual usage value (70%) and must be controlled tightly and monitored closely. In contrast, class B inventory items represent 20% of a company's business and account for around 20% of inventory; finally, class C items are relatively large in number (70%) but constitute a relatively small amount of annual usage value (10%).

Many authors agree that in addition to annual usage value, other criteria are needed for classification. In this regard, two main streams have been developed: methods based on multi-criteria decision analysis and those based on DEA.

### 2.1 MCDA-based approaches

Flores and Whybark (1986) were the first to propose a bi-criteria matrix approach, wherein annual dollar usage is combined with another criterion in a joint-criteria matrix.

Although this approach was a first step towards the multi-criteria inventory control (MCIC), issues of complexity arise when extended to more criteria in representing a multi-dimensional matrix. Furthermore, the weights of all criteria are considered to have the same weight, which is not very realistic. Therefore, multi-criteria decision making methods have subsequently been developed for MCIC. Flores et al. (1992) applied a weighted sum, where the weights of the criteria were calculated with AHP and the scores of each criterion (lead time, costs, durability) were simply normalised. Since then, several versions of AHP (Ishizaka and Labib, 2011; Saaty, 1980) have been applied (Cakir and Canbolat, 2008; Hadi-Vencheh and Mohamadghasemi, 2011; Kadziński et al., 2015; Partovi and Burton, 1993; Partovi and Hopton, 1994), as well as other MCDA methods such as ELECTRE III (Mendola and Volo, 2017) and TOPSIS (Bhattacharya et al., 2007).
Although research into the multi-criteria ranking of items has evolved rapidly, the multi-criteria sorting of items into classes is in its infancy, with the Pareto principle (Grosfeld-Nir et al., 2007) still the most widely used method for classifying items. However, the main problem with this sorting rule is that two items with the same or nearly the same score may be assigned to two different classes in order to satisfy the Pareto proportions. Moreover, products with a high priority could be assigned to class C just because the predetermined percentages of classes A and B are already satisfied, while the reverse is also true, with low priority items classified as important just to satisfy the percentage of class A. As a result, recent research has been undertaken aimed at avoiding these problems.

In sorting techniques, the classes must be defined a priori. For this purpose, the decision-maker assigns a number of (real or fictitious) reference items to each class, with the thresholds of the classes and other parameters (e.g. criteria weights) then inferred using a mathematical program (Chen et al., 2008). Soylu and Akyol (2014) employed UTilités Additives DIScriminantes (UTADIS), which is based on the same idea of inferring thresholds via a mathematical program. If the decision-maker is unable to classify certain reference items, automatic classification can be used, such as the K-means algorithm employed by (Lolli et al., 2014).

## 2.1. DEA-based approaches

Ramanathan (2006) proposed the first weighted linear optimisation model aimed at addressing the MCIC problem. This method, known as the R-model, aims to offset the impact of the subjectivity of MCDA, with weights generated endogenously. This approach is particularly useful for a new database of items, where information regarding the importance of each criterion may not be available due to a lack of history. It is worth of pointing out the similarities of the R-model with a class of linear programming model used in DEA, since an output maximising multiplier DEA model with many outputs and a constant input will reduce to the R-model. However, as this model is fully non-compensatory, an item may be inappropriately classified in class A if it is the best rated in at least one criterion, even if this criterion is of very low importance. To address this shortcoming, constraints must therefore be applied to the linear optimisation. Ng (2007) proposed to ask decision-makers for ordinal ranking of weights. In the model developed by (Hadi-Vencheh, 2010), the squared sum of the weights is normalised as a constraint. As a result, the distance between the weights increases and thus the likelihood that low scores for one criterion are ignored decreases. Another way in which to decrease the problem of the non-compensatory effect was proposed by (Zhou and Fan, 2007), who calculate the most and least favourable weights for each item. Based on these weights, good and bad indexes are created, with both indexes then combined in a weighted sum, where the decision-maker subjectively defines the weight of the indexes. However, Chen (2011) has criticised this approach because only two extreme cases are considered and each item has its own set of weights, which makes them less comparable. Furthermore, a particular criterion might be neglected by receiving a weight of zero, especially if the number of criteria increases. Therefore, he proposed calculating weights for all items and using them to evaluate the efficiency of other items; this approach is thus referred to as peer-evaluation or cross-evaluation rather than self-evaluation. A second objective is to maximise the cross-efficiency of other items. This means that cross-efficiency has the advantage of preventing unrealistic weights (i.e. all but one criteria weights are equal to zero) because they are diluted due to peer-estimation. Ladhari et al. (2016) combined the approaches of Zhou and Fan (2007) and Ng (2007), adopting the ordering of weights employed in the latter study and adding these weight constraints to the model developed in the former.

In DEA-based approaches, there is no model available with which to sort items, with assignment to classes still performed according to the Pareto principle. In the present paper, we introduce an adapted version of DEA aimed at sorting problems and apply it to MCIC. In the following section, the new method DEASort is described in detail.

## 3. Methodology

In our approach, DEASort is combined with AHP, the output of which is used to take into account the expertise of the decision-makers in calculating the weights. This weight constraint is then added to the DEASort model. The method classifies $I$ items based on $J$ criteria by $K$ decision-makers, using the 6 steps described in the following paragraphs:

*Step 1: Normalisation of item scores*

The measured score $v_{i,j}$ of each item $i$ for each criterion $j$ (e.g. frequency of issue, annual usage value, etc) is normalised on a 0-1 scale to make them comparable via the following expression:

$$(1)\ v^*_{i,j} = \frac{v_{i,j} - \min_{i=1,\dots,I} v_{i,j}}{\max_{i=1,\dots,I} v_{i,j} - \min_{i=1,\dots,I} v_{i,j}} \quad \forall i = 1, \dots, I$$

*Step 2: Criteria weight evaluation*

Criteria weights are evaluated separately via AHP by $K$ decision-makers. For this purpose, the $J$ criteria are pairwise compared in a matrix on a 1-9 scale, where 1 indicates equal importance and 9 extreme importance (Ishizaka and Labib, 2011). Weights are found by calculating the eigenvector (Saaty, 1980).

$$(2)\ A_k w_k = \lambda_{max_k} w_k$$

Where $A_k$ is the comparison matrix for decision-maker $k$

$\lambda_{max_k}$ is the principal eigenvalue for decision-maker $k$

$w_k$ is the vector of weights for decision-maker $k$.

As $A_k$ has redundancy of information, the consistency of the entered judgments by the decision-maker can be tested using the consistency ratio (CR).

$$(3)\ CR_k = \frac{CI_k}{RI}$$

Where $CI_k = (\lambda_{max_k} - n)/(n-1)$ is the consistency index for decision-maker $k$

$n$ is the dimension of the comparison matrix

RI is the ratio index.

The ratio index (RI) is the average of the consistency index of 500 randomly filled matrixes. Saaty (1980) considers that a consistency ratio exceeding 10% may indicate a set of judgments that are too inconsistent to be reliable and therefore recommends revising the evaluations.

*Step 3: Weight bounding*

In order to limit the range of possible weights, we define a lower and upper bound for each weight.

The lower bound of the weight for criterion $j$ is given by the minimum evaluation score among $K$ experts:

(4) $w_{LBj} = \min_{k=1,\dots,K}\{w_{j,k}\}$

The upper bound of the weight for criterion $j$ is given by the maximum evaluation score among $K$ experts:

(5) $w_{UBj} = \max_{k=1,\dots,K}\{w_{j,k}\}$

*Step 4: Calculation of the item priority*

For each specific item $o$ under evaluation, the mathematical programme (6) inspired by DEA is solved. This method improves on previous models (Section 2.2) by introducing the weight constraints, corresponding to the last line of (6), calculated in c).

(6) $\max P_o = \sum_{j=1}^{J} w_{o,j} v_{o,j}$

s.t. $\sum_{j=1}^{J} w_{oj} v_{i,j} \leq 1 \qquad i = 1, \dots, I$

$w_{oj} \geq 0 \qquad i = 1, \dots, I$

$w_{LB,j} < w_{oj} < w_{UB,j}$

It is to note that a weight bounding in the model may result in their infeasibility, lead to zero or negative priorities. This phenomenon indicates that the weight bounding need to be reassessed (Podinovski and Bouzdine-Chameeva, 2013).

*Step 5: Definition of classes*

The number of classes must be set and the classes defined. In general, three classes $C_c$ corresponding to $C_1 = A$, $C_2 = B$ and $C_3 = C$ are chosen. In order to define these classes each expert $k$ is asked to select $L$ reference items that (s)he knows very well and that belong to each class. The item priority $P_{ckl}$ is then calculated for each reference item.

A decision tree (Bishop, 2006) is trained on the reference items, the inputs being the item priorities $P_{ckl}$ and the outputs being their relative classes. The decision tree uses the Gini's diversity index as a splitting criterion. The number of thresholds is equal to the number of classes minus one. The classification tree is able to work with multiple reference items and is robust to misclassified reference items. The use of machine learning methodologies into inventory management is a recent new area of research (Lolli et al., 2017a; Lolli et al., 2017b).

*Step 6: Sorting into classes*

Item $z$ is assigned to class $C_I$ that has its threshold $th_I$ just below the item priority $P_z$.

$P_z \geq th_1 \quad \Rightarrow \quad k \in C_1$

$th_2 \leq P_z \leq th_1 \quad \Rightarrow \quad k \in C_2$

...

$P_z < th_{n-1} \quad \Rightarrow \quad k \in C_n$

It is to note that step 2 and 3 are optional. If no a priori information on the weights is known, then step 4 can be directly used after step 1 and the last line of (6) removed for the calculation of the priority items.

# 4. Case study

## 4.1. Introduction

A real-life case study was carried out using the British firm Entec Global Group. This company leads the international arena in providing total supply-chain management for Maintenance, Repair and Overhaul (MRO) by evaluating, designing and implementing both procurement and supply-chain management solutions. Among their core competencies, they manage warehouses stocking spare parts for many factories in several countries all over the world. Currently, the SKUs are sorted into classes $C_A$, $C_B$, $C_C$ based only on the single criterion of annual usage value, but the managers of the company have recognised that classification based on one single criterion is not realistic.

In the following sub-sections, the implementation of the steps required by DEASort is detailed (Section 4.2). Section 4.3 reports the robustness analysis of our proposal through a large experimentation by changing the number of reference items. Section 4.4 is devoted to compare the classification achieved by DEASort with those achieved both by the standard ABC classification on usage value and by the DEA-based approach proposed by Ramanathan (2006). Finally, Section 4.5 aims to investigate the inventory cost effects that DEASort does exhibit when a specific inventory control system is adopted.

## 4.2. Classification with DEASort

The three following criteria were considered in classifying 200 SKUs in a pilot study:

- Annual Usage Value (AUV): as given by the product of the unitary purchasing cost and the annual demand;
- Frequency Of Issue per year (FOI): as the number of issues per year. Each issue can contain several SKUs;
- Current Stock Value (CSV): as given by the quantity in stock multiplied by the unitary purchasing cost.

Based on these new criteria, the methodology described in Section 3 was applied as follows:

*Step 1: Normalisation of item scores*

The values of the three criteria AUV, FOI and CSV were normalised for each SKU using eq. (1), providing $v^*_{i,AUV}$, $v^*_{i,FOI}$ and $v^*_{i,CSV}$, respectively.

*Step 2: Criteria weight evaluation*

In order to weight the criteria, a questionnaire was submitted to the two spare parts managers of Entec Global Group. They were asked to pairwise compare the importance of the three criteria, with the weights derived using the eigenvalue method (2). The results of this process are given in Table 1. As this table shows, the ordering of weight importance is identical for both managers and the difference in weight values is small. Frequency of issue is the most weighted criterion, with its weight more than double that of annual usage value. This indicates that the company’s previous classification method based solely on this criterion was lacking in precision.

The most important criterion was found to be the frequency of issue per year, followed by the annual usage value and finally the current stock value.

| Criteria | Spare part manager 1 | Spare part manager 2 | Difference |
| --- | --- | --- | --- |
| Frequency of issue | 0.637 | 0.722 | 0.085 |
| Annual usage value | 0.258 | 0.227 | 0.031 |
| Current stock value | 0.105 | 0.051 | 0.054 |

Table 1: Criteria weights estimated by the two spare parts managers

*Step 3: Weight bounding*

The range of weights permissible in DEA for each criterion was obtained by setting the lowest (4) and highest (5) values from Table 1. The result of this procedure is displayed in Table 2.

| Criteria | Lower bound | Upper bound |
| --- | --- | --- |
| Frequency of issue | $w_{LB,FOI} = 0.637$ | $w_{UB,FOI} = 0.722$ |
| Annual usage value | $w_{LB,AUV} = 0.227$ | $w_{UB,AUV} = 0.258$ |
| Current stock value | $w_{LB,CSV} = 0.051$ | $w_{UB,CSV} = 0.105$ |

Table 2: Range of permissible weights

*Step 4: Calculation of the item priorities*

Algorithm (6) was implemented in R and the item priority $P_k$ was calculated for each item. Item priority values were not revealed to the spare parts managers at this point.

*Step 5: Definition of classes*

The two spare parts managers selected a typical item for each class; their item priorities are listed in Table 3.

| Criteria | Spare part manager 1 | Spare part manager 2 |
| --- | --- | --- |
| Class A | $P_6 = 0.655$ | $P_{104} = 0.249$ |
| Class B | $P_{173} = 0.055$ | $P_{98} = 0.072$ |
| Class C | $P_{78} = 0.036$ | $P_{116} = 0.026$ |

Table 3: Typical items and their item priorities

This training set (Table 3) was fed into the classification tree algorithm in MATLAB to train the decision tree described in Figure 1.

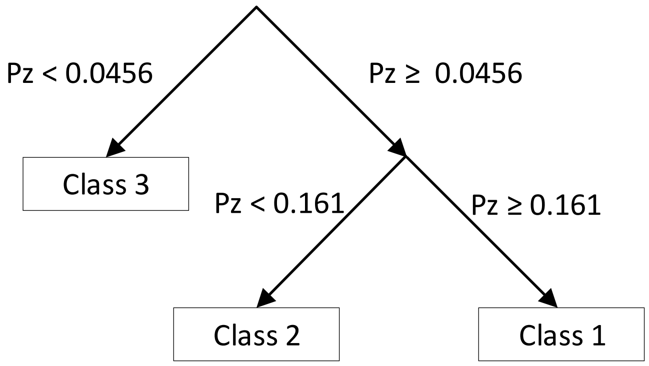


Figure 1: trained decision tree.

*Step 6: Assignment to classes*

Table 4 lists the items assigned to classes A and B, with all remaining items assigned to class C (see table in the supplementary materials). Twenty-three items were found to have a score above the limiting profile of class A and were therefore assigned to this class. Only fifteen items were assigned to class B, with their scores falling between the limiting profiles of class A and class B. One hundred and sixty-two items scored below the limiting profile of class B and were therefore assigned to class C.

| A | | B | |
|---|---|---|---|
| Item | score | item | score |
| 109 | 0,7298 | 93 | 0,1360 |
| **6** | **0,6555** | 97 | 0,1110 |
| 13 | 0,5560 | 184 | 0,0922 |
| 1 | 0,4821 | 161 | 0,0885 |
| 4 | 0,4312 | 31 | 0,0852 |
| 2 | 0,4270 | **98** | **0,0724** |
| 3 | 0,4266 | 186 | 0,0668 |
| 8 | 0,4245 | 117 | 0,0650 |
| 5 | 0,3905 | 87 | 0,0640 |
| 7 | 0,3835 | 121 | 0,0605 |
| 9 | 0,3801 | 159 | 0,0579 |
| 10 | 0,3745 | **173** | **0,0554** |
| 11 | 0,3664 | 103 | 0,0478 |
| 18 | 0,3572 | 89 | 0,0471 |
| 12 | 0,3277 | 164 | 0,0463 |
| 14 | 0,3117 | | |
| 19 | 0,2902 | | |

| | | | |
|---|---|---|---|
| 20 | 0,2872 | | |
| 16 | 0,2870 | | |
| 15 | 0,2775 | | |
| 17 | 0,2765 | | |
| **104** | **0,2497** | | |
| 166 | 0,2143 | | |

Table 4: Items assigned to classes A and B. Reference items are indicated in bold

## 4.3. Classification robustness analysis

In order to evaluate the robustness of the classification, several decisions trees are trained with different reference items. The classification of items in the supplementary materials is considered as the control set. In total six groups of simulations are carried out with 10,000 individual classifications per group.

Each simulation group uses a fixed number of reference items from 1 to 6 per class. The reference items are randomly chosen from the subset of items with different priorities. The resulting classification is compared with the control classification in the supplementary materials, Step 5 reference items are not used for training or comparison.

The results of each simulation group are used to calculate precision and recall performance measures with 95% confidence intervals. Given a class, the precision is computed as the total number of correctly classified items divided by the total number of items assigned to that class in the simulation. The recall is calculated per class as the number of correctly classified items in the class divided by the total number of items correctly belonging to the class. The confidence intervals in both cases are computed with the Clopper-Pearson method. Figures 2, 3 and 4 present the precision performance for class A, B and C respectively while Figures 5, 6 and 7 present the recall performance for class A, B and C respectively. The solid lines in the figures outline the average performance while the dotted lines represent the confidence intervals.

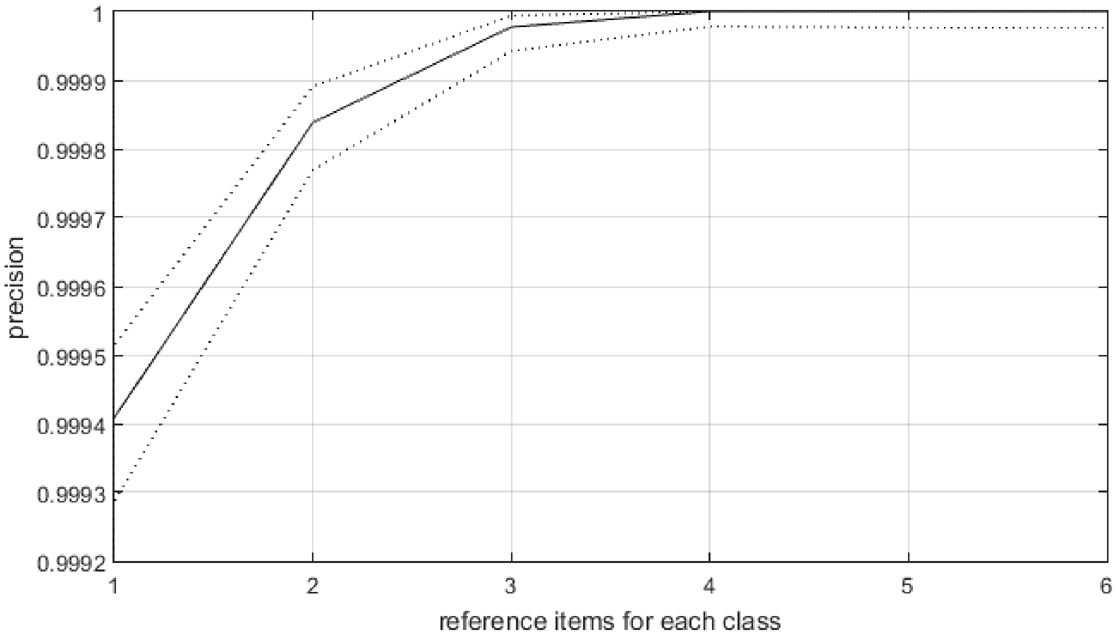


Figure 2: Class A precision according to the number of reference items.

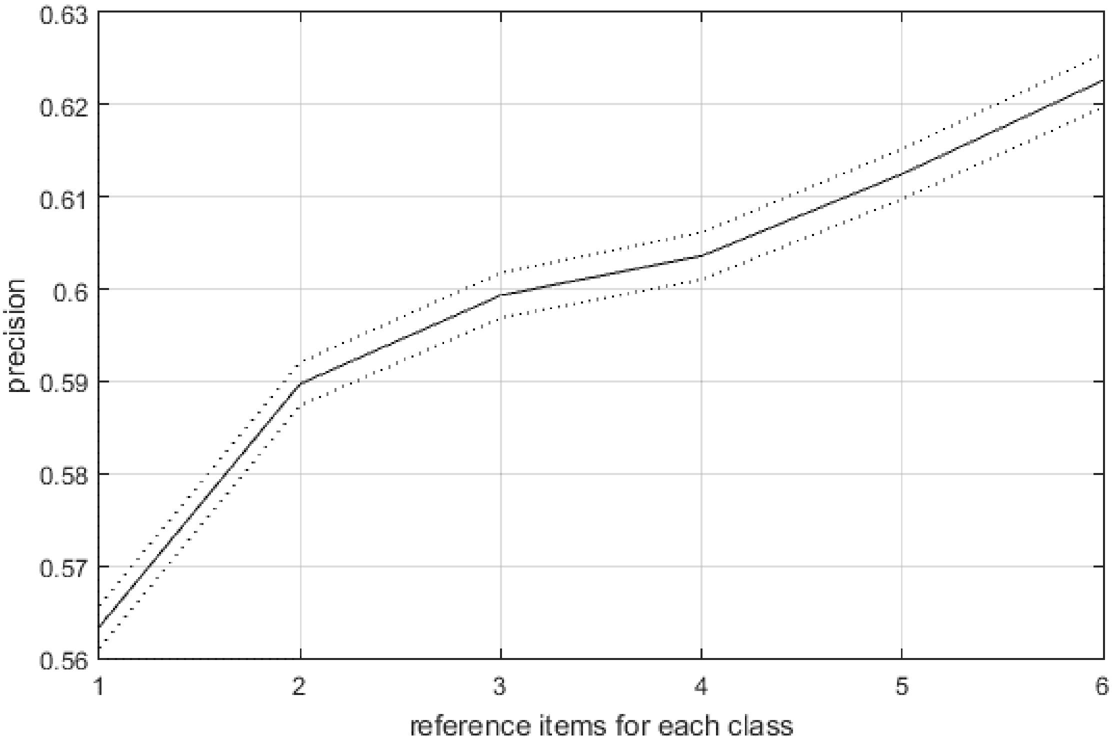


Figure 3: Class B precision according to the number of reference items.

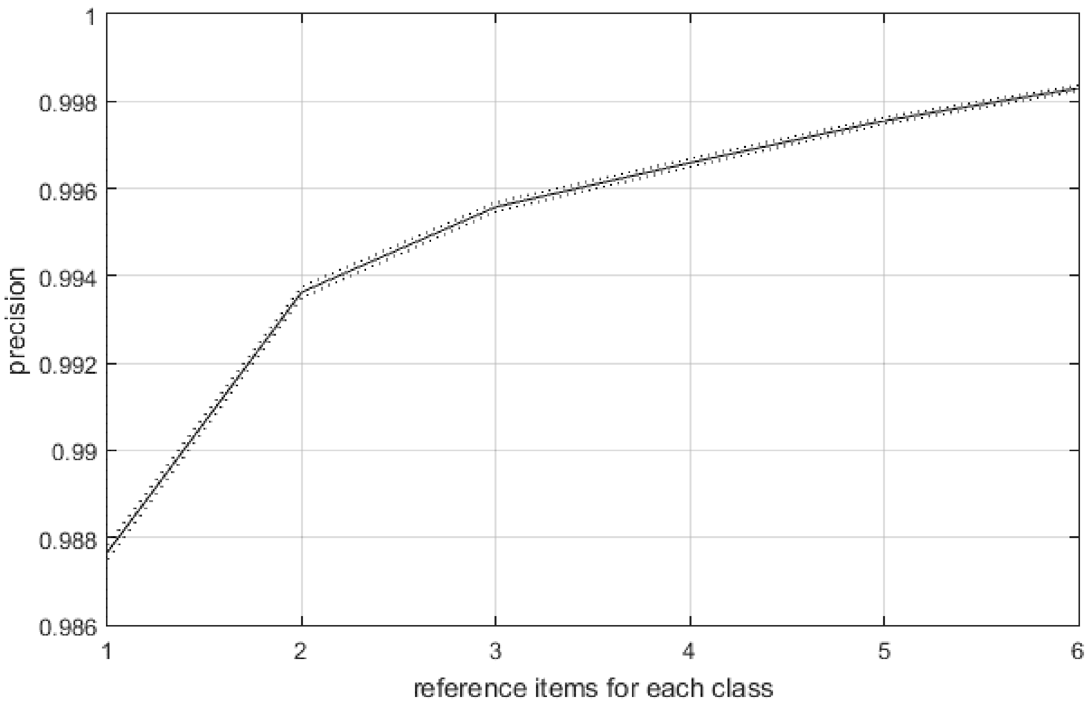


Figure 4: Class C precision according to the number of reference items.

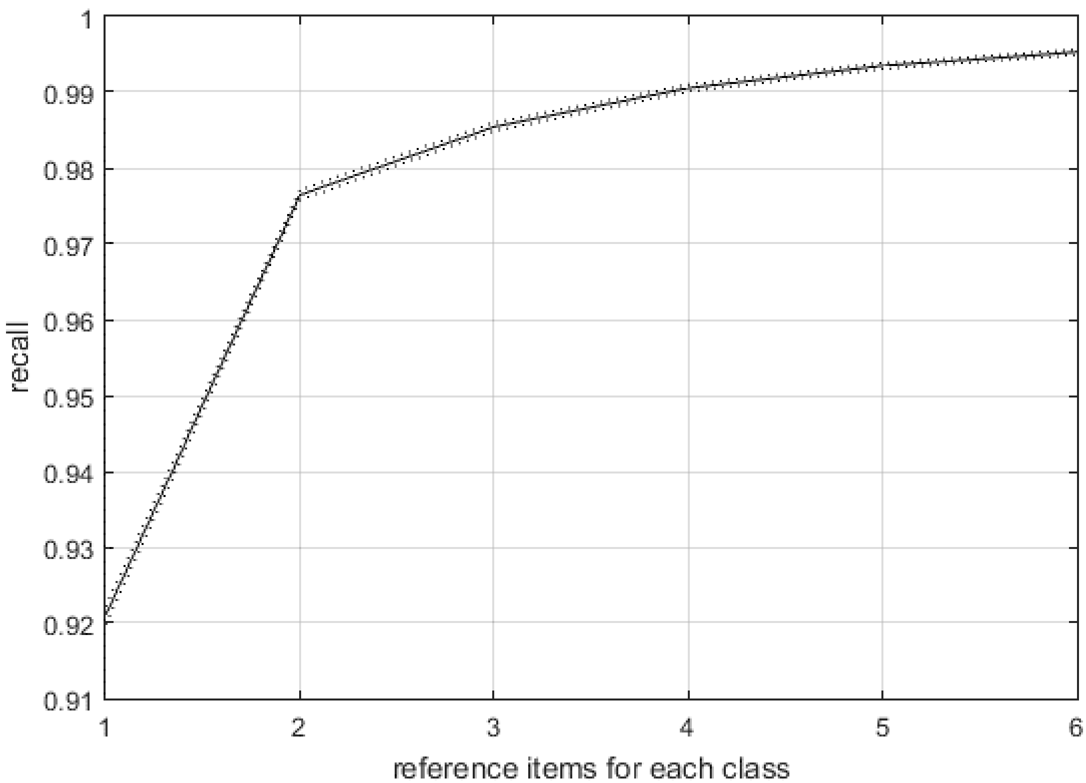


Figure 5: Class A recall according to the number of reference items.

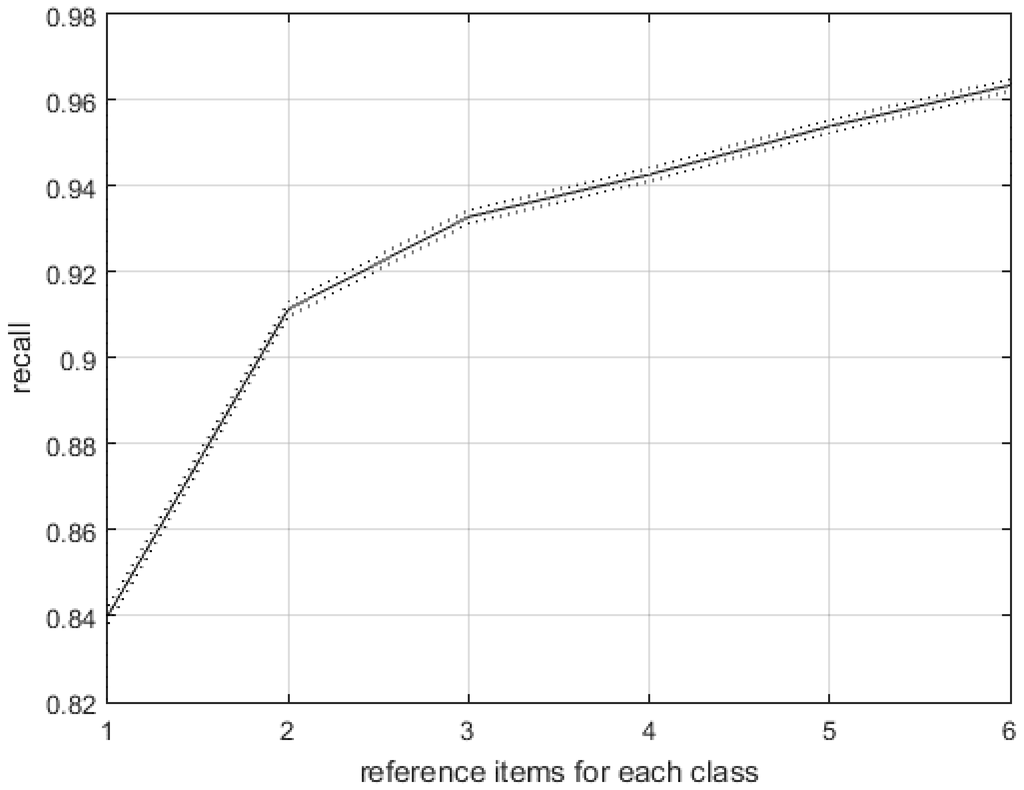


Figure 6: Class B recall according to the number of reference items.

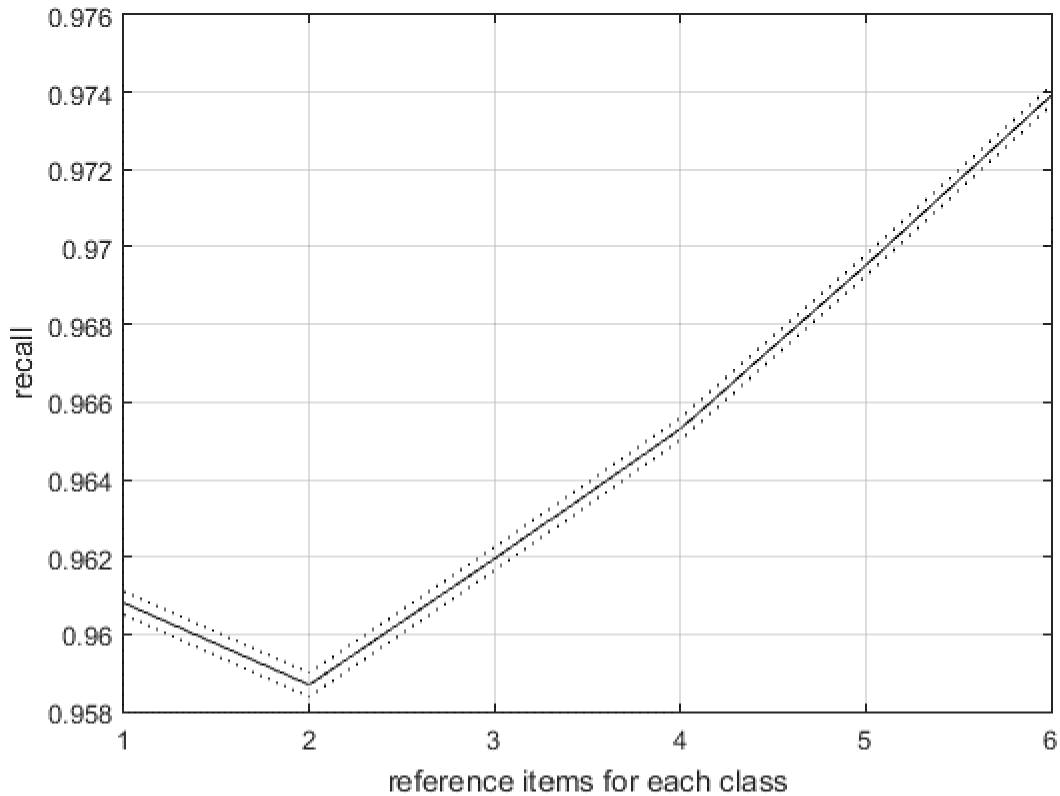


Figure 7: Class C recall according to the number of reference items.

The decision trees achieve both high precision and high recall even with a small number of reference items. Their performance increases as the number of reference items per class increases. Class B is the only case with a relative low performance, i.e. less than 0.7 precision for any number of reference items. This case can be explained by the small number of elements in class B compared to class C and by the proximity of class B to class C in the item priority space. Figure 8 shows an example of this phenomenon.

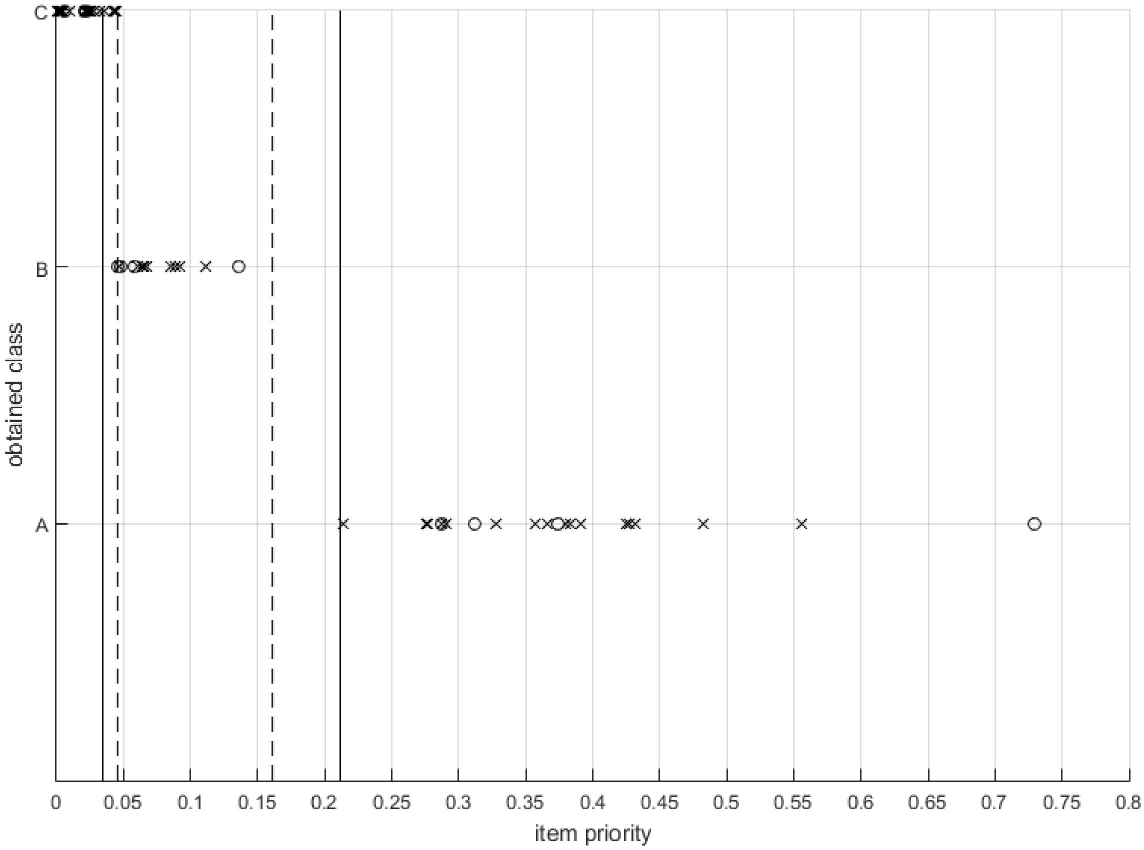


Figure 8: On example of simulation with four reference items.

In Figure 8, the dotted lines are the original reference classification boundaries calculated by the decision tree (Figure 1). The obtained classification is represented by the crosses, the solid lines are the obtained classification boundaries and the circles are the reference items. The obtained division line between class B and C is slightly tilted towards class C. Since class B and C are close and class C is densely packed, this shift moves 13 items from class C to class B. The relative impact of this error is different for the two classes: class C is large and only 8.3% of its items end up in class B, while class B is small and ends up with 40.9% of the items classified in B that actually belongs to class C.

Figure 9 summarizes the phenomenon previously discussed. The precision of both class B and C is computed in different scenarios by moving the decision tree cut point from its control state. As expected minor shifts have a far greater impact on class B precision than on class C.

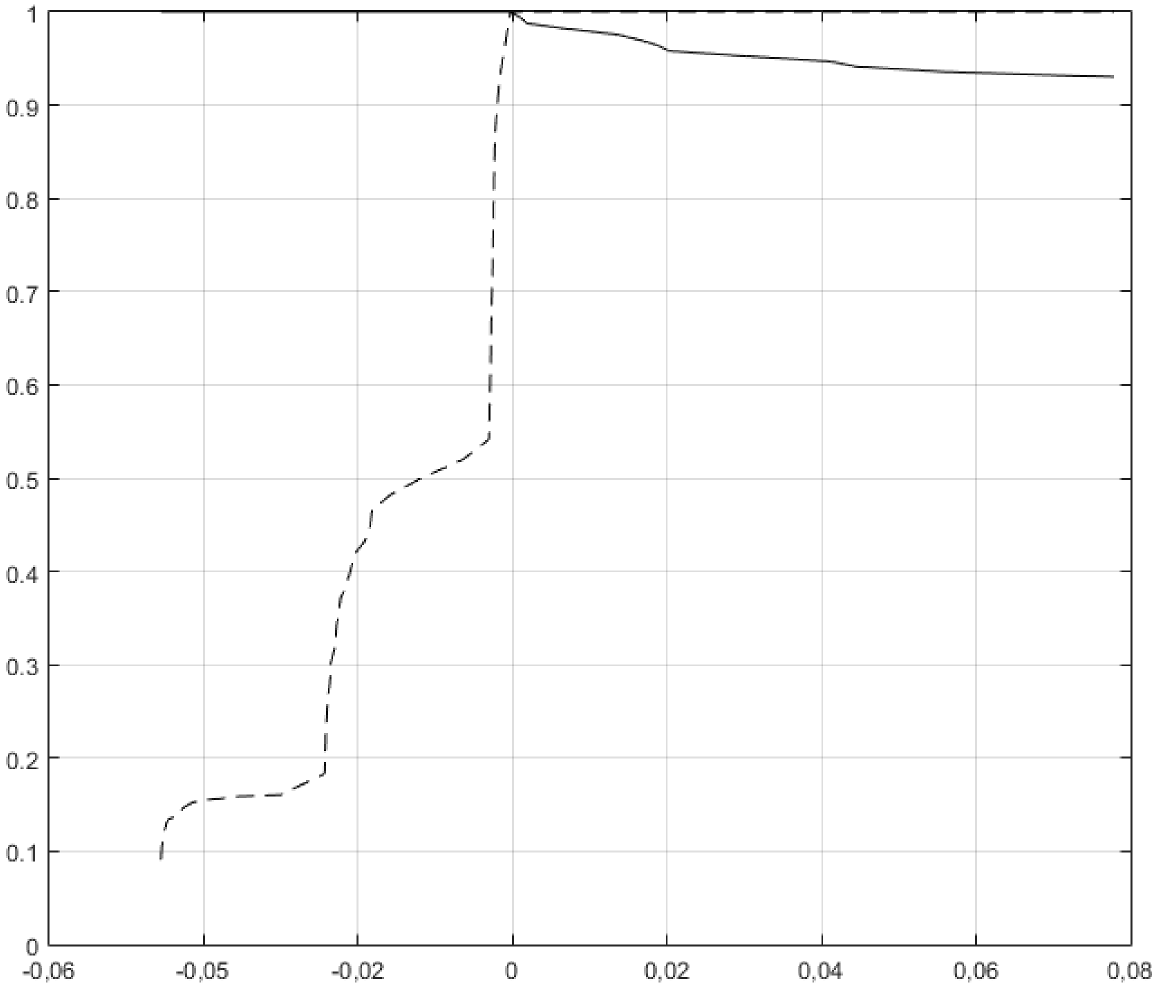


Figure 9: Class B (dotted line) and C (solid line) precision by varying the thresholds.

## 4.4. Comparison with other classifiers

In this section, the results obtained using DEASort (Table 4) are compared with both the original ABC classification used by Entec Global Group (i.e. based solely on the annual usage value) (Table 5) and the classification proposed by (Ramanathan, 2006) (Table 6).

| | | ABC classification | | | |
|---|---|---|---|---|---|
| | | A | B | C | Sum |
| DEASort | A | 9 | 8 | 6 | 23 (11.5%) |
| | B | 3 | 3 | 9 | 15 (7.5%) |

| | | | | | |
|---|---|---|---|---|---|
| | C | 0 | 3 | 159 | 162 (81%) |
| | Sum | 12 (6%) | 14 (7%) | 174 (87%) | |

Table 5: Comparison between DEASort and ABC

| | | (Ramanathan, 2006) | | | |
|---|---|---|---|---|---|
| | | A | B | C | Sum |
| DEASort | A | 19 | 4 | 0 | 23 (11.5%) |
| | B | 0 | 15 | 0 | 15 (7.5%) |
| | C | 1 | 21 | 140 | 162 (81%) |
| | Sum | 20 (10%) | 40 (20%) | 140 (70%) | |

Table 6: Comparison between DEASort and (Ramanathan, 2006) classifier

Both the traditional ABC and Ramanathan method have a predefined increasing percentage of items in each class. By following this exogenous rule, it can have unexpected consequences, i.e. more than 20% of the items are critical, some of the critical items end up assigned to class B and, as a consequence, some of the class B items are pushed to class C. Classification methods based on fixed percentages enhance the classification errors if said percentages are not representative of the actual criticality ratios. In contrast, DEASort uses a justifiable rule to assign items, and the resulting classes do not necessarily produce an increasing item percentage.

In the DEASort case class A is more populated than class B and class C remains the largest, with an item percentage lying between that of the ABC and Ramanathan methods. Six items that were assigned to class A using ABC analysis were assigned to class C by DEASort. This difference in the two classes is due to the fact that the new approach considers more criteria than ABC analysis. One item classified as belonging in class A by the Ramanathan method was assigned to class C by DEASort (Table 6). This is due to the fact that the Ramanathan method allows the use of any weights for the criteria, which permits an item to achieve the highest possible score. In contrast, DEASort constrains weights to a range given by experts.

## 4.5. Cost-oriented comparison

As outlined in Table 5, DEASort and ABC classify 85.5% of the items the same way. A cost-oriented comparison then is developed to gauge the potential impact of this change. The analysis implements a continuous review reorder point policy (s, Q) see (Silver et al., 1998)) and measures the mean relative safety stock holding cost and the fill-rate difference for each class. Table 7 summarizes the model assumptions. Only the items with positive monthly demand (22%) are analysed since it is not possible to implement a data-driven reorder policy for those with null demand. This analysis aims at measuring the potential savings obtained by adopting a DEASort approach where such savings can be quantified.

| Measure | | Value |
|---|---|---|
| Demand relative standard deviation $\frac{\sigma}{\mu}$ | | 0.6 |
| Lead-time $LT$ | | 7 $[day]$ |
| Order cost $c_o$ | | 2.5 [€] |
| Yearly relative holding cost $c_{hr}$ | | 0.2 $[year^{-1}]$ |
| Cycle service level $csl$ | Class A | 0.99 |
| | Class B | 0.95 |
| | Class C | 0.90 |

Table 7: Model assumptions.

For each item the model calculates the parameters of a Gamma distributed lead-time demand (since $\frac{\sigma}{\mu} > 0.5$ the Gamma is preferred over a Normal distribution):

(7) $a_i = \frac{\mu_i^2}{\sigma_i} LT_i$

(8) $b_i = \frac{\sigma_i}{\mu_i}$

The inverse Gamma computes the reorder point for each item and classification (DEASort or ABC):

(9) $s_{i,ABC} = \Gamma\left(csl_{i,ABC}, a_i, b_i\right)^{-1}$

The safety stock is calculated as the reorder point minus the average demand during the lead-time:

(10) $ss_{i,ABC} = s_{i,ABC} - a_i b_i$

The yearly relative holding cost is transformed into a daily holding cost by changing the unit measure and multiplying for the item value $p_i$:

(11) $c_{i,h} = \frac{c_{hr} p_i}{365}$

The daily safety stock cost is obtained from the safety stock and the daily holding cost:

$$(12)\ c_{i,ABC,ss} = c_{i,h} ss_{i,ABC}$$

In order to calculate the fill-rate, i.e. the fraction of demand (measured in items) not in backorder during a replenishment cycle, the economic order quantity must be computed first as follows:

$$(13)\ eoq_i = \sqrt{\frac{2c_o\mu_i}{c_{i,h}}}$$

The fill-rate is then calculated as:

$$(14)\ fr_{i,ABC} = 1 - \frac{a_i b_i \left(1 - \Gamma\left(csl_{i,ABC}, a_i + 1, b_i\right)^{-1}\right) - rp_{i,ABC}\left(1 - \Gamma\left(csl_{i,ABC}, a_i, b_i\right)^{-1}\right)}{eoq_i}$$

At the end of these calculations, the safety stock holding cost and the fill-rate are available for each item and classification. These measures are separated by class (A, B or C) and classification (DEASort or ABC) and their class average value is computed. For instance, for class A they are respectively computed by:

$$(15)\ c_{A,ABC,ss} = \frac{\sum_{i\in A} c_{i,ABC,ss}}{|A|}$$

$$(16)\ fr_{A,ABC} = \frac{\sum_{i\in A} fr_{i,ABC}}{|A|}$$

Finally, for each class, the relative difference between the DEASort and ABC measures is calculated. For class A such a relative difference is give by:

$$(17)\ fr_{A,\Delta} = \frac{fr_{A,DEASort} - fr_{A,ABC}}{fr_{A,ABC}}$$

The results are summarized in Table 8. The DEASort classification reduces the safety stock holding cost by more than 40% in each class while its impact on the fill-rate is negligible.

| | $fr_{\Delta}$ | $c_{\Delta,ss}$ |
|---|---|---|
| Class A | $8.087 \cdot 10^{-5}$ | $-0{,}507$ |
| Class B | $2.260 \cdot 10^{-4}$ | $-0{,}467$ |
| Class C | $1.107 \cdot 10^{-4}$ | $-0{,}411$ |

Table 8: Mean relative safety stock holding cost and fill-rate difference for each class

## 5. Conclusions

The Multi-Criteria Inventory Classification problem has been receiving increased attention from experts, as traditional ABC analysis based on a single criterion may be effective but it is not necessarily efficient. This paper addresses the ABC inventory classification problem through the MCIC approach and proposes the DEASort methodology to classify a large number of items in three classes: A, B and C. Although most studies thus far have evaluated the problem within a very general framework, specific industry characteristics may impact on the resulting classification. Thus, the new methodology was applied to a real-life case study involving the British procurement and logistics firm Entec Global. DEASort, inspired by DEA, enables the judgments of spare parts managers and decision-makers to be taken into account in different phases, such as in the weighting of criteria and the choice of reference items. Moreover, the method is highly useful and effective in solving complex MCIC problems involving a large number of criteria. Whereas traditional Data Envelopment Analysis has total weight flexibility that many DMUs can take advantage of by assigning to some criteria a zero weight, the new methodology constrains the weights within a certain range, ensuring that all criteria are considered for each item. Furthermore, DEASort avoids fixing the classes percentages. Fixed classes percentages could lead to enhanced classification errors if said percentages are not representative of the actual criticality ratios.

The application of an inventory system to the classes obtained by means of DEASort has led to relevant holding cost savings in comparison with the standard ABC classification, i.e. more than 40% in each class. Actually, these results are case-sensitive but enforce our beliefs about the effectiveness of DEASort in real settings.

Finally, it is worth to remark that DEASort is a generic classification method, and thus it can be easily applied to other sorting problems.

**Acknowledgment**

We are thankful to Entec Global Group, which provided us the data for the case study.

## References

Bhattacharya, A., Sarkar, B., Mukherjee, S., 2007. Distance-based consensus method for ABC analysis. International Journal of Production Research 45, 3405-3420.

Bishop, C., 2006. Pattern Recognition and Machine Learning. Springer, New York.

Cakir, O., Canbolat, M., 2008. A web-based decision support system for multi-criteria inventory classification using fuzzy AHP methodology. Expert Systems with Applications 35, 1367-1378.
Charnes, A., Cooper, W., Rhodes, E., 1978. Measuring the efficiency of decision making units. European Journal of Operational Research 2, 429-444.
Chen, J.-X., 2011. Peer-estimation for multiple criteria ABC inventory classification. Computers & Operations Research 38, 1784-1791.
Chen, Y., Li, K., Kilgour, M., Hipel, K., 2008. A case-based distance model for multiple criteria ABC analysis. Computers & Operations Research 35, 776-796.
Dickie, H., 1951. ABC Inventory Analysis Shoots for Dollars, not Pennies. Factory Management and Maintenance 109, 92-94.
Flores, B., Olson, D., Dorai, V., 1992. Management of multicriteria inventory classification. Mathematical and Computer Modelling 16, 71-82.
Flores, B., Whybark, C., 1986. Multiple Criteria ABC Analysis. International Journal of Operations & Production Management 6, 38 - 46.
Grosfeld-Nir, A., Ronen, B., Kozlovsky, N., 2007. The Pareto managerial principle: when does it apply? International Journal of Production Research 45, 2317-2325.
Hadi-Vencheh, A., 2010. An improvement to multiple criteria ABC inventory classification. European Journal of Operational Research 201, 962-965.
Hadi-Vencheh, A., Mohamadghasemi, A., 2011. A fuzzy AHP-DEA approach for multiple criteria ABC inventory classification. Expert Systems with Applications 38, 3346-3352.
Ishizaka, A., Labib, A., 2011. Review of the main developments in the analytic hierarchy process. Expert Systems with Applications 38, 14336-14345.
Kadziński, M., Tervonen, T., Figueira, J., 2015. Robust multi-criteria sorting with the outranking preference model and characteristic profiles. Omega 55, 126-140.
Ladhari, T., Babai, Z., Lajili, I., 2016. Multi-criteria inventory classification: new consensual procedures. IMA Journal of Management Mathematics 27, 335-351.
Lolli, F., Gamberini, R., Regattieri, A., Balugani, E., Gatos, T., Gucci, S., 2017a. Single-hidden layer neural networks for forecasting intermittent demand. International Journal of Production Economics 183, 116-128.
Lolli, F., Ishizaka, A., Gamberini, R., 2014. New AHP-based approaches for multi-criteria inventory classification. International Journal of Production Economics 156, 62-74.
Lolli, F., Ishizaka, A., Gamberini, R., Balugani, E., Rimini, B., 2017b. Decision Trees for Supervised Multi-criteria Inventory Classification. Procedia Manufacturing 11, 1871-1881.
Mendola, D., Volo, S., 2017. Building composite indicators in tourism studies: Measurements and applications in tourism destination competitiveness. Tourism Management 59, 541-553.
Mohamadghasemi, A., Hadi-Vencheh, A., 2011. Determining the ordering policies of inventory items in class B using If–Then rules base. Expert Systems with Applications 38, 3891-3901.
Ng, W., 2007. A simple classifier for multiple criteria ABC analysis. European Journal of Operational Research 177, 344-353.
Partovi, F., Burton, J., 1993. Using the Analytic Hierarchy Process for ABC Analysis. International Journal of Operations & Production Management 13, 29-44.
Partovi, F., Hopton, W., 1994. The Analytic Hierarchy as Applied to Two Types of Inventory Problems. Production and Inventory Management Journal 35, 13-19.
Podinovski, V., Bouzdine-Chameeva, T., 2013. Weight restrictions and free production in Data Envelopment Analysis. Operations Research 61, 426-437.
Ramanathan, R., 2006. ABC inventory classification with multiple-criteria using weighted linear optimization. Computers & Operations Research 33, 695-700.

Rezaei, J., Dowlatshahi, S., 2010. A rule-based multi-criteria approach to inventory classification. International Journal of Production Research 48, 7107-7126.
Roy, B., 1978. ELECTRE III: algorithme de classement base sur une présentation floue des préférences en présence de critères multiples. Cahiers du CERO 20, 3-24.
Saaty, T., 1980. The Analytic Hierarchy Process. McGraw-Hill, New York.
Silver, E., Pyke, D., Peterson, R., 1998. Inventory management and production planning and scheduling, 3rd ed. Wiley, New York.
Soylu, B., Akyol, B., 2014. Multi-criteria inventory classification with reference items. Computers & Industrial Engineering 69, 12-20.
van Kampen, T., Akkerman, R., Pieter van Donk, D., 2012. SKU classification: a literature review and conceptual framework. International Journal of Operations & Production Management 32, 850-876.
Zhou, P., Fan, L., 2007. A note on multi-criteria ABC inventory classification using weighted linear optimization. European Journal of Operational Research 182, 1488-1491.