Available online at www.sciencedirect.com

ScienceDirect

Procedia Manufacturing 42 (2020) 62–69

Procedia MANUFACTURING

www.elsevier.com/locate/procedia

International Conference on Industry 4.0 and Smart Manufacturing (ISM 2019)

Machine learning methods to improve the operations of 3PL logistics

Alessandro Tufano[a,*], Riccardo Accorsi[a], Riccardo Manzini[a]

[a]Department of Industrial Engineering (DIN), Alma Mater Studiorum – University of Bologna, Viale del Risorgimento, 2, 40136 – Bologna - Italy

* Corresponding author. E-mail address: alessandro.tufano2@unibo.it

Abstract

Nowadays, the variety in the product mix, unpredictable customer demand and the need for a high level of service are crucial challenges in the management of a supply chain. Flexible processes are needed to gain competitive advantage and economic edges. This paper presents a data-driven application of unsupervised machine learning clustering algorithms to a real-world case study in the automotive industry. The clustering input dataset collects the data available to a third-party logistics (3PL) provider. Clustering algorithms are used to define product families for the assignment of the workload to the processing resources. Several clustering algorithms (k-means, Gaussian mixture models and hierarchical clustering) define different product families scenarios using different tuning parameters. The impact of each clustering scenario on the operations is assessed via a dashboard of logistics KPIs to identify the best performing clustering algorithm. The performance of each clustering is, then, compared to a logistic benchmark given by a capacitated clustering to identify the best compromise between a logistic-constrained algorithm with a long runtime and fast data-driven uncapacitated algorithm.

Keywords: Automotive; 3PL; machine learning; clustering; family grouping; logistics

1. Introduction & Literature review

In the last decades, the logistics market becomes challenging, and logistics providers strive to satisfy their customers due to new global trends in the customers' demand [1]. They require high customisation of the products and very short shipping times. These expectations lead to a new organisation of the supply chain processes, including modular products, form postponement, high storage levels to satisfy the unpredictable market demand.
These characteristics are highly recognisable in business-to-consumer (B2C) (e.g., e-commerce) as well as business-to-business (B2B) services. The latter is the case of the spare parts management in the automotive industry, where a large number of suppliers and third-party logistics (3PL) providers serve a few producers. Producers offer to their final customers a small product mix with a hundreds degrees of freedom in the customisation (e.g., colours, labels, package, materials). This fact affects all the actors of the supply chain generating a workload peak to be processed within a short service time [2,3]. The manufacturing industry deals with workload peaks using technologies as flexible manufacturing systems (FMS) [4] or reconfigurable manufacturing systems (RMS) [5,6]. These systems allow to perform different processing tasks on a "family" of similar products, i.e. a large number of products having slight differences (e.g., the colour or the shape). FMS and RMS usually involve a significant degree of automation, which requires an initial investment dedicated to a single or a small number of product families. In the majority of the cases, these investments are not suitable for 3PL providers that work with short-term contracts which do not provide an adequate payback time for a long-term investment.
FMS and RMS need balanced product families both from the workload and the product similarity point of view. The literature proposes several methodologies to aggregate products into homogeneous families, enhancing the flexibility of a production system. The majority of these methodologies rely on hierarchical clustering (i.e., dendrograms) based on the

10.1016/j.promfg.2020.02.023

bill of materials (BOM) of the products [7–10] which allows catching a proximity measure of the products since the BOM describes how raw materials converge into a finished product. Other techniques define a proximity matrix between products using a similarity index based on the features of the product or its production cycle [11–13].
Other approaches use unstructured data (i.e., text descriptions) as input. These techniques apply in business analytics [14–16] and asset maintenance [17], but no studies investigate their role in production environments.
A crucial issue is that clustering algorithms do not take into account the workload produced by each cluster [18,19]. Few studies in the field of warehousing science pursue these goals together [19,20]. In general, warehousing operations are much more repetitive and predictable than production ones. All the stock-keeping-units have a similar workflow (i.e., putaway, storage, picking, packing and shipping). 3PL production plant workflow is similar to the one of a warehouse since they have vast production quantities and a wide products portfolio, but they perform a minimal number (i.e., less than 10) of similar tasks on the products.
This paper uses clustering methods to address the grouping of products into families assessing the logistic benefit for a 3PL operations provider. In this study, we aim at answering the following unmet research question:

RQ1: "How to group products into processing families when: the product portfolio is extremely wide (1), any production resource can perform any production task (2), customer demand is unpredictable (3).

The methodology we propose is data-driven. Data-driven means that a methodology uses data available in a real-world scenario. For this reason, data-driven models are practice-ready, and they do not need additional data collection for their implementation. Machine learning and artificial intelligence are entirely data-driven, and they represent a powerful tool to deal with the improvement of a supply chain when a high degree of complexity is involved.

The remainder of this paper is organised as follows: Section 2 presents the algorithms used to group products into families and the KPIs to measure the logistics efficiency of each clustering algorithm. Section 3 presents an application to a real-word third-party logistics (3PL) processing plant. Section 4 discusses and concludes the work.

2. Methodology

This section introduces the methodology to cluster products into families and the KPIs to assess the logistic impact of a product family.
3PL providers often work with a lack of data about the product and the process. In particular, family grouping techniques usually need the definition of the production cycle for each resource, i.e. pinpointing which resource (e.g., machine or operator) can perform a specific task on the product. Often, 3PL providers do not have the definition of the production cycles in their information system because of lack of integration with the suppliers' systems. 3PL providers define the cycle task by labelling every single product (i.e., the label indicates the quantity, the list of task to be processed and the final customer). Alternatively, they broadly define the type of the task which a manual operator will interpret during the operations on the workbench (e.g., a "packing" task must be interpreted by an operator to correctly choose the packaging depending on the type and quantity of the product).
For these reasons, we decide to base our clustering methodology on the few information available for each incoming product: description, weight, length, height and width. 3PL providers often have the complete set of this information: suppliers generally share the description of the products (to avoid processing errors) while product weight and sizes (i.e., length, height and width) are almost always measured by the provider to identify the proper package.
Grouping products into homogeneous clusters leads to the design of a limited number of product families with a predefined production cycle using a precise set of resources.

Machine learning aims at uncovering hidden patterns of a $n \times p$ dataset X where n is the number of observations (i.e., one for each product) and p is the number of features (i.e., description, weight and sizes). Supervised methods link the dataset X to a target vector y of n elements (one for each observation). Unsupervised methods define patterns and similarities among the dataset X without the help of a reference variable y. For this reason, we will introduce a series of logistics KPIs to evaluate the outcome (i.e., the families) of the unsupervised methods implemented.
Table 1 introduces the unsupervised methods (i.e. the clustering algorithms) used to generate product families. Different scenarios are produced depending on the input dataset X and the tuning of the algorithm (for hierarchical clustering). Three different clustering scenarios are proposed: sizes and weight (1), package code (2) and product description (3) -based scenarios. Several clustering algorithms are applied to each scenario. Further details on the grouping rationale of the algorithms and their tuning are provided in the following paragraphs.

Table 1. Clustering Algorithms Implemented

Input dataset scenarios	Clustering Algorithm	Algorithm Tuning
Sizes and weight	K-means Gaussian Mixture Model Hierarchical Clustering with Euclidean distance	Complete linkage Single linkage Average linkage
Package code	Hierarchical Clustering with Jaccard distance	Complete linkage Single linkage Average linkage
Description	Hierarchical Clustering with Jaccard distance	Complete linkage Single linkage Average linkage
Sized and weight	Capacitated Hierarchical Clustering	

2.1. Clustering based on sizes and weight

The input dataset X of the sizes and weights based scenario (1) involves real numbers only. To first investigate the structure of the dataset and to run the algorithm within a brief time, we implement principal component analysis (PCA) on X. PCA reduces the dimension of an initial dataset $n \times p$ into a

principal component (PC) dataset $n \times c$ where c is the number of the PCs. The value of each PC is a linear combination of the p initial features such that the c components are orthogonal. These properties allow expressing the majority of the information of the initial dataset X with a small subset c of orthogonal variables. Each observation is now identified by a point in a c-dimensional space. For the sake of clarity, let assume $c = 2$ (this is also the value used in the case study), without loss of generality.
The machine learning algorithms cluster the points (i.e., the products) depending on their position on a plane defined by the two PCs. The distance is, then, Euclidean and the algorithms perform well even when n is large (e.g., tens of thousands of products). Some of these methods require the number of clusters to be defined in advance; for these reasons, several scenarios with a different number of clusters are proposed in Section 3. All these methods are well-known clustering method in the field of statistics and machine learning [21].

K-means algorithm does not require assumptions on the statistical distribution of the input data. It defines k clusters and assigns each point to a cluster such that the Euclidean distance between each point and the cluster centroid is minimum.

Gaussian Mixture Model assumes the data to be generated by k multidimensional gaussian distribution. The model identifies the parameters (i.e., the mean and the covariance matrix) of each of the k gaussian distributions and the probability for each point to be generated by one of the k distribution. Each point is assigned to the gaussian distribution with the highest probability to have generated that point, obtaining k clusters are obtained.

Hierarchical clustering defines a convergence tree based on the proximity between each couple of points. For each iteration of the algorithm, the two closest points condense into a single cluster. This process is represented as a dendrogram (i.e., a tree representation of the clusters) where at the first iteration of the algorithm each point is a cluster (a leave of the tree) and at the last iteration, all the points form a single cluster (the root of the tree). This algorithm defines the number of clusters *a posteriori*, i.e. any number of cluster $k \in [1, n]$ can be extracted by the dendrogram by defining a proximity threshold. The tuning of the algorithm identifies how the define the distance between a newly created cluster and all the other points: complete and single linkage consider, respectively, the maximum or the minimum distance between the clustered points and all the other points to be aggregated. The average linkage uses the average between them.

2.2. Clustering based on the package code

The input dataset X of the package code -based scenarios is fed by m (with $m \geq n$) observation with two features: the product code and the package code (i.e. the type of package used with that product). Starting from X, a distance matrix D, $n \times n$ (where n is the number of products) is defined and consequently used for hierarchical clustering. In the previous application of hierarchical clustering, the coordinates of each point on the PC space (a matrix $n \times c$) are enough to implicitly define the distance matrix D using the Euclidean distances between points. In this case, the distance matrix D is more complex since all the $n \times n$ values are calculated separately, based on a similarity index. In this case, the Jaccard index s_{ij} is used to measure the similarity between products i and j.

$$s_{ij} = \frac{a}{a + b + c} \tag{1}$$

Where a is the number of observation when i and j have the same package; b is the number of observations where i and j has different packages; c is the number of observations when i and j are not involved. The outcome of the hierarchical clustering is a dendrogram as well as discussed in the previous paragraph.

2.3. Clustering based on the description

The input dataset X of the product description-based scenarios is the list of the description of each of the n products. Text mining techniques are used to pre-process the description text strings and to use this information for hierarchical clustering exactly the same way as for package code. The description contained in X is first cleaned of special characters (e.g., +, /, |, ”, ’, .) and numbers replaced by blank spaces. Then, stopwords, i.e. words with a low information level (e.g., ‘the’, ‘is’, ‘from’, ‘to’), are removed from the description. Finally, a bag of word model implements a frequency analysis on the remaining words. Words with a number of occurrences above a minimum threshold (e.g., 10 occurrences) enter the products vocabulary (PV). Each product i is, then, characterised by a subset of words S_i in the PV obtained as the intersection between its description and the PV. Hierarchical clustering is, then, applied as well as with package code defining a distance matrix D based on the Jaccard index calculated on the values of S_i.

2.4. Capacitated hierarchical clustering

All the previous algorithms are based on the proximity distance between points but do not consider a capacity limit for the generated clusters. In practice, this is an evident limit of clustering techniques, especially when a cluster uses a set of scarce resources. For this reason, we propose an original algorithm to cluster points considering both their proximity and a capacity upper bound. The algorithm is inspired to hierarchical clustering with a capacity constraint. Let d_i be the demand of a point i and C the maximum capacity of a cluster. The algorithm works as follows.

1. Define gg as the sorted list of tuples of point (i, j) by descending values of D.
2. Scan gg and find a tuple (i, j) such that $d_i + d_j \leq C_k$.
   If no tuples are found, exit.
3. Group i and j within k.
   Set $C_k = C_k - d_i - d_j$.
   Set $D_l = -1$ for each point $l \in k$
   Update D according to a linkage algorithm
   Go to 2.

The outcome of this algorithm is a set of clusters whose cardinality is unknown in advance. Each cluster k has a total demand $d = \sum_{i \in k} d_i$ with $d \leq C$.

2.5. KPIs for cluster assessment

A dashboard of KPIs is defined to evaluate the impact of each cluster from a logistics perspective. Table 2 provides the details of the dashboard. KPIs are based on a historical order list storing the information of each processing order within a specified time horizon (e.g. a year). The order list should include the product code, the package type, the client, the quantity and the processing time for each order. All these information are easy to get from the order list of a 3PL provider.

Table 2. Dashboard of KPIs to assess a cluster

KPI	Description
N. of products	Counts the number of products associated with a cluster
N. of orders	Counts the number of orders associated with a cluster
N. of package	Counts the number of different package codes required by the orders of a cluster
N. of tasks	Counts the number of different types of tasks performed on the orders of a cluster
Processed quantity	Counts the total processed quantity (the number of parts) associated with the orders of a cluster
Total working time	Counts the total working time (i.e., seconds) associated with the orders of a cluster

Since each clustering algorithm produces k cluster, the standard deviation of each KPI on the clusters defined by the algorithm is calculated. In particular, a low standard deviation for each cluster identifies which clustering strategy defines balanced clusters. Balancing clusters lead to a similar amount of workload and requires a similar amount of resources, that is the aim of generating product families.

3. Case study

This section presents a real-world application of the proposed methodology in the automotive industry. The clustering algorithms are implemented to define product families in a 3PL packaging plant processing more than 58.000 different products. The processing plant works as an intermediate stage of the automotive supply chain where incoming products are collected, packaged and labelled according to the clients' needs. The clients are production plants where cars or tractors are assembled and prepared for shipping to the final user. Since these clients mainly work Just-In-Time (JIT) the 3PL packaging plant has to absorb an unpredictable demand in a very short time. They offer three different levels of service (LoS) processing products within 24, 48 or 72 hours. The operations of the 3PL packaging plant consist of oiling, packing and labelling spare parts. Figure 1 illustrates the variability of the workload in terms of the number of processed orders and processed quantities. Each colour in the figure indicates a different "service type," i.e. the definition of the series of task to perform on the product and the package to use.

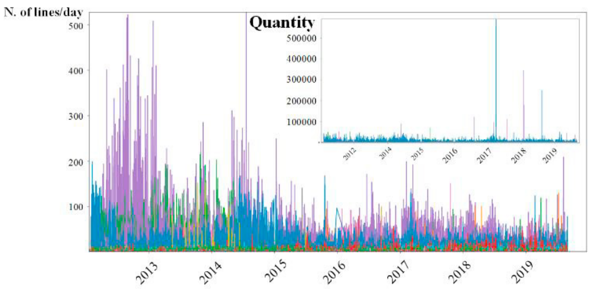

Figure 1: Workload trend over the last 7 years.

As Figure 1 shows, the workload is highly variable and it depends on the service type. In addition, the quantity processed is variable too and slightly correlated with the number of lines processed. Figure 2 shows a heatmap built on about 2 millions of orders over a period of 7 years, identifying the correlation between the parameters of the orders:
-the dimensions, volume and weights of the items;
-the dimensions, volume and weights of the packages;
-the code of the service pack associated with an order.

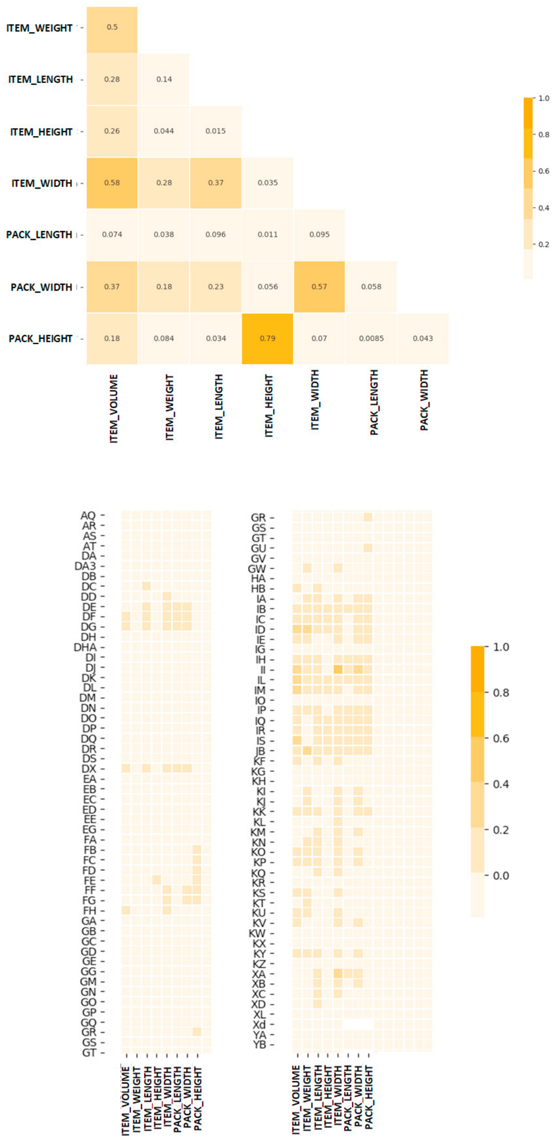

Figure 2: Correlation matrix between items, packages and service types (coded by two or three letters).

The matrix shows an obvious significant correlation between the dimensions of the package and the items. In addition, there are significant correlations between the dimensions of packages and items and some service type. This result suggests that there is the possibility to cluster item based on the service type and assign them to specific workbenches in order to reduce the complexity and the inventory of packages needed on each workbench.

Operators perform the tasks of a specific service type on manual workbenches with no automation. All the operators on the 12 workbenches can process any of the 58.000 products. This fact leads to a very low specialisation of the operators, and unpredictable material flows since any of the workbenches can

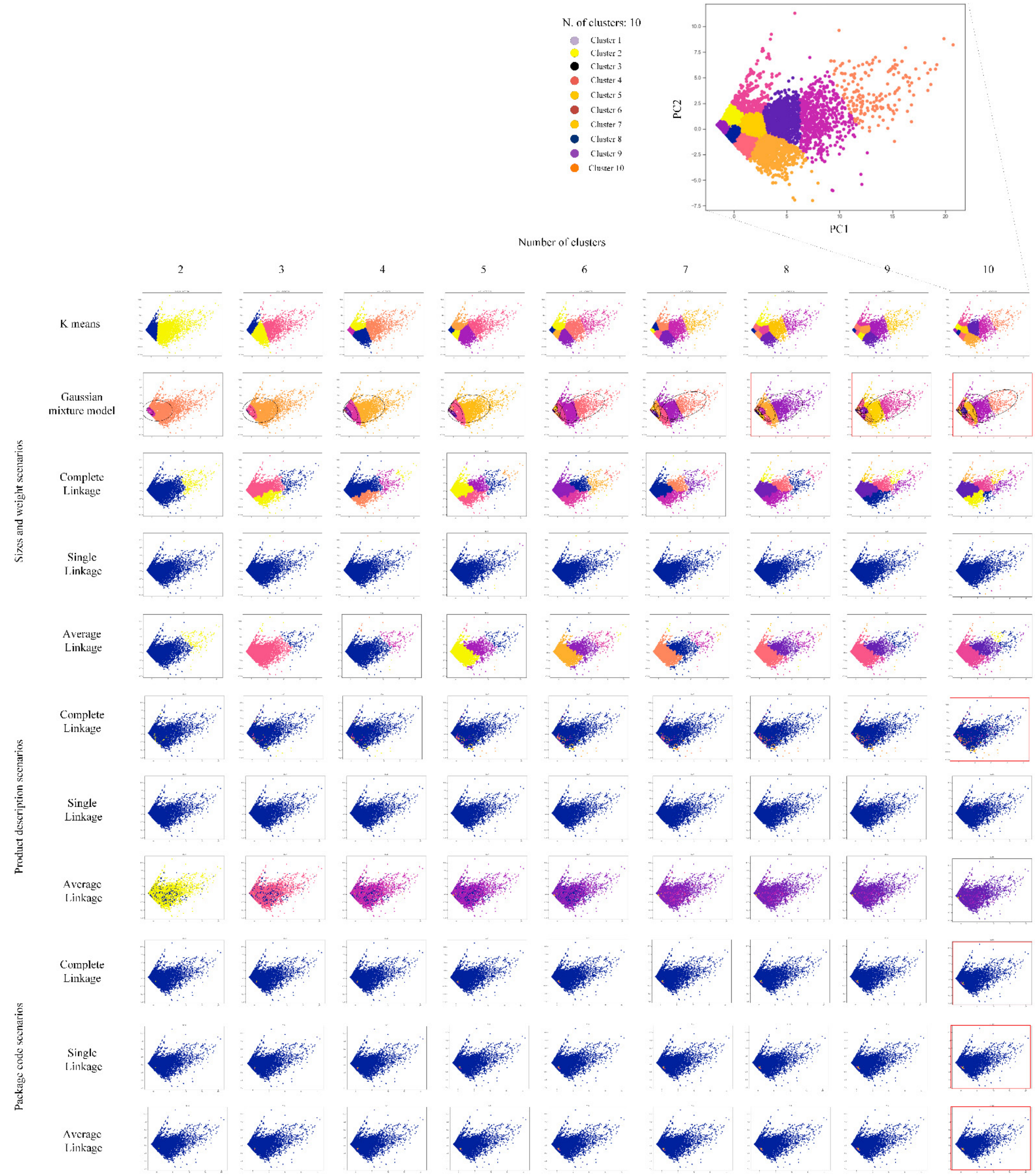

Figure 3: Comparison of the clustering algorithms. The algorithm with the highest logistics performance are marked in red.

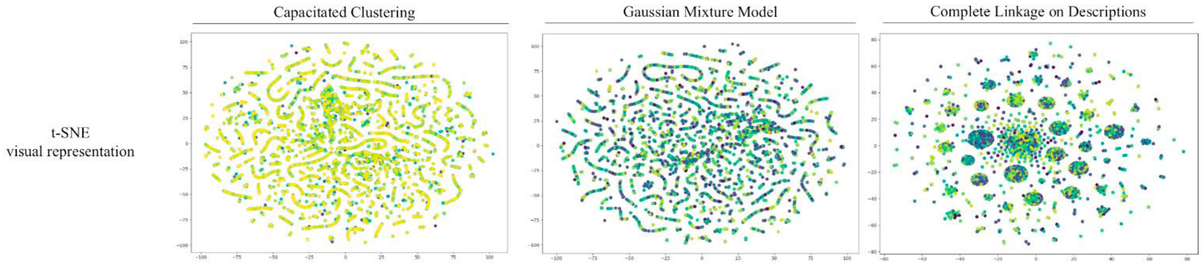

Figure 4: comparison between capacitated and uncapacitated clustering using t-SNE

request all the 1500 different types of packages. Besides, some clients require a customised tertiary package structured as a shelf of the dimension of a pallet. These shelves are placed directly to a workstation of the client's assembly line. For this reason, the 3PL package plant has to deal with high work-in-process (WIP) levels on the workbenches due to products, packages (with the size of a carton box) and customised shelves (with the size of a pallet). To deal with this randomised material flows and WIP of the 3PL plants, we applied the proposed methodology aiming at the definition of a number of families. We start with an increasing number of cluster (from two to ten). This value is, then, compared to an estimation of the real number of workbenches with the application of the capacitated clustering algorithm. Figure 3 presents the graphical results of the algorithms in the different clustering scenarios and with a different number of clusters- To graphically compare all the methods at a glance, each dot is one of the 58.000 products while the axis of each subplot represents the two PCs of the sizes and weight input dataset (even in the clustering based on the product code and description where PCA is not applied). Different colours indicate different product families. In the scenario generated by the weights and size input dataset, points closer in the graph are clustered together (having the same colour). This is not always true in case of package- or description-based clustering. The dashboard of KPIs is introduced to assess the logistic performance of clustering. Table 3 illustrates the top ten most performing scenarios, according to the lowest standard deviation of the KPIs indicated in the dashboard.

Table 3. Most Performing Clustering Algorithms

Algorithm	N. of orders (std)	N. of products (std)	Processed quantity (stf)	N. of package (std)	Total working time (std)
GMM k=10	15961	4797	2587147	23	20766935
GMM k=9	17701	5377	2791238	24	22337420
Descr. Compl. Link. k=10	54152	14680	3468679	40	23504968
GMM k=8	19193	5946	2932669	26	23547446
Package Compl. Link k=10	51269	14351	3328214	49	23718403
Package Single Link k=10	51269	14351	3328214	49	23718403
Package Av. Link k=10	51269	14351	3328214	49	23718403
Descr. Average Link k=10	51269	14351	3328214	49	23718403
KMEAN k=10	24696	7304	3209344	28	24567096
Package Compl. Link k=9	53621	15005	3475843	52	24764986

Table 3 evaluates the performance of the algorithm from a logistic point of view measuring the variability of the process in each clustering scenario. While thinking about the variability of the process, it is necessary to remember that the actual process is completely random, and it results totally out of control since no assignment rules have previously been developed. To compare the performance of these clustering algorithms with a logistic benchmark, we compare the two top algorithms of Table 3 with the outcome of the capacitated clustering algorithm.

The capacitated clustering algorithm considers a maximum allowable capacity that is fixed and equal for all the cluster and an amount of demand required by each product. To feed the algorithm with this data, we set a time & motion monitoring campaign in order to identify an average processing time required by each product. This data collection applied on a subset of the products (i.e. the items belonging to the 95° percentile of the total number of processed lines) due to the very high number of items. The amount of time required by the products with the highest workload defines the maximum capacity for each cluster. The capacitated clustering produces 20 clusters. This number is used to compare the performance with the Gaussian Mixture Model and the Complete Linkage Clustering based on the Descriptions, setting the number of clusters $k = 20$. Figure 4 illustrates the outcome of this comparison using a visual analytics technique called t-SNE. This technique visually identifies clusters based on the matrix X, $n \times p$ of the observation that is projected onto a 2-dimensional space preserving the proximity of each observation according to the t-distribution. The colours are associated accordingly with the cluster assignment given by the algorithms. Figure 4 shows that it is difficult to identify a topology of the cluster (as it happens in Figure 3) since the number of clusters is high and the input data are scattered. On the other side, analysing Table 4 it is possible to evaluate the performance of the algorithm from a logistic point of view, identifying the variability of the processes organised according to this clustering.

Table 4 illustrates the KPIs and compares their variance (using absolute and relative value compared to the capacitated case) calculated on a time horizon of 7 years. It is easy to check that the capacitated clustering provides the highest balanced scenario with the lowest variance. The variance in workload (i.e. seconds) between the 20 clusters has an average of 180 hours per year per workbenches. This is a low gap, considering that the variability in the number of products and packages is dramatically reduced compared to the other scenarios.

Table 4. Comparison between capacitated and uncapacitated algorithms.

Gaussian Mixture Model provides a poorer result that has to be manually checked and assessed before a physical implementation since a couple of clusters results extremely small in workload compared to the average of the others. Nevertheless, it is important to remember that GMM provides the uncapacitated result in short running time (i.e. about 5 minutes) compared to a huge running time of the capacitated algorithm which needs around 20 hours of runtime on a computer equipped with 8Gb memory and a 2.7GHz processor.

4. Discussion

The case study highlights the effects of the use of clustering algorithms to balance the material flows of a 3PL packaging plant. Grouping products into families with a similar workload leads to:

- a static number of packages/pallet-shelves on the workbenches;
- a levelled and more predictable workload on the workbenches;
- higher efficiency due to the specialisation of the operators (i.e., a lower time to perform the tasks).

Also, a more efficient organisation of the plant layout is possible since the families generating the highest material flows can be assigned to the workbenches placed near the inbound/outbound area of the plant leading to a smart plant layout design.

From a mathematical point of view, it is interesting to remark that algorithms producing a higher number of clusters outperform the others. This fact was predictable since a higher number of clusters allows to partition the workload into more levelled subsets. Nevertheless, it is essential to note that the Gaussian Mixture Model (GMM) clustering outperforms the methods based on the process (i.e. package code and product description). GMM clustering is based only on the features of the products, but it produces the higher logistic performance even without considering the production cycles (i.e. the service type) as an input data. This is a great value of the data-driven approach since a good clustering model can be built upon the data which are always available to any 3PL provider without other assumptions or data collections. Besides, it is highly generalizable since the type of data is incredibly simple to be collected and are always available to any 3PL provider working in the packaging sector.

From a logistic point of view, capacitated clustering remains the most reliable choice since it provides more robust results and a lower variance among the WIP. Nevertheless, when capacitated clustering is too hard to solve, data-driven approaches provide interesting results within a short run time. In addition, these approaches are extremely valuable for 3PL providers that process many materials and receives many data connected to them, but they barely can analyse this data and organise their operations efficiently. In this case study, the 3PL provider benefits from the clustering approach since the business-as-usual scenario is completely out-of-control. The lack of assignment of parts and packages to workbenches produces chaos in the daily operations with the impossibility to precisely analyse the process, allocate costs or make it leaner.

Algorithms	N. of orders (std)	N. of products (std)	Processed quantity (std)	N. of packages (std)	Total working time (std)
Capacitated	5665.3	299.1	700459.9	299.1	4330213.5
GMM	23120.0	1184.3	3067547.0	1184.3	25706129.7
Descr. Compl. Link	21284.7	1184.3	5285096.9	1154.3	40495702.6
Percentage comparison					
Capacitated	100%	100%	100%	100%	100%
GMM	408%	396%	438%	396%	594%
Descr. Compl. Link	376%	396%	755%	386%	935%

Clustering products into families is a prerequisite for process optimisation when a production plant has the characteristics remarked in the RQ1. It is true that clustering may produce a little imbalance in the workload assigned to each workbench, but it opens to a scientific analysis of the WIP allowing to implement a lean organisation by controlling the inventory level and the workforce needed.

Future studies should focus on embedding clustering and other data science techniques in the design of an industrial plant and in the redesign of 3PL production processes and layout.

5. Conclusion

This paper presents an application of machine learning unsupervised algorithms to define product families in a 3PL production plant. Products are clustered using k-means, Gaussian mixture models and hierarchical clustering based on the product or process features (weight, sizes, description, the package used). A case study applies this methodology to a real-world environment represented by a 3PL packaging plant of the automotive sector. The impact of each clustering algorithm is measured using logistics KPIs. The results are compared with capacitated clustering based on the time workload assignable to each cluster. The application shows that Gaussian Mixture Models based on the product features (i.e., sizes and weight) outperform the others producing a more balanced workload even if they are uncapacitated and not based on the process features.

References

[1] Research and markets, Logistics Market: Global Industry Trends, Share, Size, Growth, Opportunity and Forecast 2018-2023, 2018.

[2] E. Dehdar, A. Azizi, S. Aghabeigi, IEEE Int. Conf. Ind. Eng. Eng. Manag. 2019-Decem (2019) 84–88.

[3] G. Baryannis, S. Validi, S. Dani, G. Antoniou, Int. J. Prod. Res. 7543 (2018) 1–24.

[4] H. Seifoddini, B. Tjahjana, Int. J. Prod. Res. 37 (1999) 3263–3273.

[5] R. Galan, J. Racero, I. Eguia, J.M. Garcia, Robot. Comput. Integr. Manuf. 23 (2007) 489–502.

[6] R. Kumar, P.K. Jain, N.K. Mehta, Adv. Prod. Eng. Manag. 5 (2010) 45–58.

[7] P. Daie, S. Li, Int. J. Prod. Res. 54 (2016) 5468–5479.

[8] K. Hölttä-Otto, V. Tang, K. Otto, Res. Eng. Des. 19 (2008) 127–141.

[9] R. Accorsi, F. Garbellini, F. Giavolucci, R. Manzini, A. Tufano, in: Sustain. Food Supply Chain., Elsevier, 2019.

[10] C. J. Romanowski, R. Nagi, M. Sudit, Comput. Oper. Res. 33 (2006) 3150–3160.

[11] Y. Yin, K. Yasuda, Int. J. Prod. Econ. 101 (2006) 329–352.

[12] D.R. Moscato, Int. J. Prod. Res. 14 (1976) 401–406.

[13] C. Chen, L. Wang, Int. J. Prod. Res. 46 (2008) 4259–4284.

[14] N. Ur-Rahman, J.A. Harding, Expert Syst. Appl. 39 (2012) 4729–4739.

[15] R. Menon, L.H. Tong, S. Sathiyakeerthi, Reliab. Eng. Syst. Saf. 88 (2005) 171–180.

[16] M.A. Serbout, A. Berrado, L. Benabbou, Logist. Oper. Manag. (GOL), 3rd Int. Conf. on. IEEE (2016) 1–6.

[17] L. Huang, Y.L. Murphey, Lect. Notes Comput. Sci. (Including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 4031 LNAI (2006) 1309–1317.

[18] R. Rajesh, S. Pugazhendhi, K. Ganesh, Int. J. Adv. Manuf. Technol. 61 (2012) 431–440.

[19] D. Knoll, M. Prüglmeier, G. Reinhart, Procedia CIRP 52 (2016) 145–150.

[20] H.Y. Lam, K.L. Choy, G.T.S. Ho, S.W.Y. Cheng, C.K.M. Lee, Int. J. Prod. Econ. 170 (2015) 763–779.

[21] T. et. all. Hastie, Math. Intell. 27 (2009) 83–85.