logistics

MDPI

Article

A Machine Learning and Multi-Criteria Decision-Making Approach to Cycle Counting

Laura Vaccari [1,*], Elia Balugani [1,2], Francesco Lolli [1,2] and Rita Gamberini [2]

[1] Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, Via Amendola 2, 42122 Reggio Emilia, Italy; elia.balugani@unimore.it (E.B.); francesco.lolli@unimore.it (F.L.)
[2] Interdepartmental Centre En&Tech, Piazzale Europa, 1, 42124 Reggio Emilia, Italy; rita.gamberini@unimore.it
* Correspondence: laura.vaccari@unimore.it

Abstract

Background: Inventory record inaccuracy (IRI) causes discrepancies between physical and digital inventories, leading to production delays and customer dissatisfaction. Cycle counting, in this context, is a common corrective action. Pareto-based ABC analysis is widely used to decide which items to inspect, but it often oversimplifies inventory decisions, and recent studies suggest that multi-criteria decision-making (MCDM) and machine learning (ML) may offer more effective solutions. *Methods:* This study applies the analytic hierarchy process (AHP) method, combined with K-means (AHP-K), to classify stock-keeping units (SKUs) into three groups with distinct counting policies. A selection procedure is then applied to identify an optimal ML algorithm and compare its classification with the original AHP-K results; each model in this phase is trained on a subsets of 100 SKUs. A Veto method is also introduced to improve output consistency for both AHP-K and the best ML method, and a comparative cost evaluation is presented. *Results:* The ML-AHP-K-Veto classification achieves over 90% accuracy. Analysis of a dataset of 12,863 SKUs from a mechanical manufacturing company shows minimal cost differences between ML-based and MCDM classifications, but significant differences compared to Pareto-based costs. *Conclusions:* ML can effectively address IRI, supporting the development of pure ML applications, including decision-maker (DM) preferences, to manage cycle counting strategies.

Keywords: multi-criteria inventory classification (MCIC); cycle counting; multi-criteria decision-making (MCDM); inventory control; machine learning (ML); inventory inaccuracy; ABC analysis; cost analysis

Check for updates

Academic Editor: Robert Handfield

Received: 21 November 2025
Revised: 17 December 2025
Accepted: 25 December 2025
Published: 29 December 2025

1. Introduction

The increasing complexity of global supply chains and warehouses combined with the large quantities of products that need to be managed and stored, has heightened the challenges associated with inventory record inaccuracy (IRI). The IRI problem, also known as the inventory misalignment problem, takes place when the quantities available in a warehouse differ from those on record in the company database. Research highlights that 25% of the stock-keeping units (SKUs) show such a misalignment [1], leading to disruptions in production, cascading effects that impair operational performance [2], stockouts, lower sales, reduced customer satisfaction, and damaged company reputation [3]. The main causes of IRI include manual counting errors, incorrect transaction recordings, material losses, damage, theft, and data synchronization issues [4–6]. In particular, critical activities like order picking alone can account for approximately 60% of the manual labor [7]. To manage IRI, prevention (e.g., radio-frequency identification) and correction are the most common approaches [5] and, among those, the most frequent corrective action is cycle counting; the periodic inventory inspections of selected SKUs [8]. Compared to perpetual inventory, cycle counting reduces production interruptions, enables faster error detection, minimizes inventory adjustments, and improves stock availability [9]. Grouping methods for cycle counting, to define inspection policies at a class level, include ABC classification, process-based grouping, location-based grouping, and event-triggered counts [10,11]. ABC classification, in particular, categorizes SKUs into three classes (A, B, and C) of decreasing importance and different counting policies, allowing managers to focus on the most important ones [12–14]. Cycle counting remains a critical yet challenging task in inventory management, requiring efficient methods to prioritize and classify items. This study makes three key contributions to this area:

1. It uniquely integrates machine learning (ML) algorithms with multi-criteria decision-making (MCDM) techniques, specifically analytic hierarchy process (AHP), to cycle counting in inventory management on a large-scale industrial dataset of 12,863 SKUs.
2. Unlike prior work that applies MCDM or ML to cycle counting, this research evaluates model performance both in classification accuracy and economic impact, explicitly quantifying cost savings related to counting and downtime.
3. We establish ML models that effectively capture decision-maker (DM) preferences alongside traditional MCDM techniques. We intend to extend their effectiveness in future articles by completely eliminating the dependence on MCDM.

In this article, we:

1. Select the best ML method from a small set of SKUs classified through the AHP with the K-means (AHP-K) [15].
2. Measure to what extent the selected ML method follows such AHP-K classification.
3. Compare the impact of a Veto step on both the original AHP-K and the best ML model.
4. Evaluate the impact of the different models on costs.

The paper is structured as follows. Section 2 provides an analysis of the state of the art on MCDM and ML methods for cycle counting. Section 3 elaborates on the procedure for selecting the best ML algorithm and on the methodology to calculate the overall costs within the ABC classification framework. In Section 4, we apply both the traditional AHP-K method and the ML approaches to a large industrial dataset, comparing them from both a classification accuracy (CA) and a monetary point of view. Section 5 discusses our key findings. Section 6 presents the results, the managerial implications, the observed limitations, and proposes directions for future research.

2. Literature Review

We conducted two searches in the Scopus database to explore the state of the art in both multi-criteria and machine learning inventory management, to look for applications related to cycle counting.

2.1. Multi-Criteria Inventory Management Background

We searched in the Scopus database with the query TITLE-ABS-KEY ((“multicriteria” OR “mcda” OR “mcdm” OR “multi-criteria” OR “multiple criteria”) AND (“inventory” OR “stock control” OR “warehouse management” OR “inventory management” OR “stocktaking”) AND (“cycle count*” OR “cyclic count*” OR “inventory audit*” OR “stock audit*” OR “inventory accurac*” OR “discrepanc*”)), and found that only [9,16] integrated ABC classification for cycle counting with multi-criteria inventory classification (MCIC) approaches. Medina et al. [9] directly outlined the use of analytic hierarchy process (AHP) to define the three classes, while Montes et al. [16] only referenced its use in the cycle counting context. We also analyzed studies, outside cycle counting but within the broader inventory management field, that integrated MCIC approaches with machine learning (ML) ones. The review by Shahmohammad et al. [17] highlighted the advantages of combining ML techniques with fuzzy multi-criteria decision-making (MCDM) to address the complexities of modern real-world environments. Costa et al. [18] used principal component analysis (PCA) to support the MCDM approach in determining criteria weights, addressing the subjectivity in criteria selection and the lack of formal validation practices. Ali et al. [19] explored the potential of MCDM integration with ML models as a future research direction of their review. Lolli et al. [15] directly combined AHP and K-means (AHP-K). Within this line of research, validating the most effective ML model for replicating MCDM classification is crucial to ensure its reliable application. Roy et al. [20] applied a Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) for ABC classification on real data, and leveraged ML (KNN and SVM) to replicate this classification for future inventory items. Their approach, still, was not applied to cycle counting and did not show the monetary impact of the classification decisions. Kartal et al. [12], used naïve Bayes, Bayesian networks, neural networks (NNs), and support vector machines (SVMs) to replicate MCDM methods in an MCIC context. Similarly, Kabir et al. [13] replicated fuzzy AHP results with an NN. Meanwhile, Abidin et al. [21] explored both decision trees and SVM approaches, reporting accuracies of 99% and 68%, respectively. Differently, the article of Lolli et al. [22] compared SVM and NN to enhance inventory classification and reduce costs, training the data on historical data. This comparison of ML methods with MCDM can also be seen in more disparate fields. In environmental science, for example, Aslam et al. [23] compared AHP and TOPSIS with logistic regression and SVM, finding that SVM achieved the highest accuracy (88%).

2.2. Machine Learning Inventory Management Background

We searched in the Scopus database with the query TITLE-ABS-KEY ((“machine learning” OR “ml” OR “artificial intelligence” OR “ai” OR “deep learning” OR “neural network*” OR “data-driven” OR “predictive model*”) AND (“inventory” OR “stock control” OR “warehouse management” OR “inventory management” OR “stocktaking”) AND (“cycle count*” OR “cyclic count*” OR “inventory audit*” OR “stock audit*” OR “inventory accurac*” OR “discrepanc*”)) and we observed that only Weaver et al. [24] utilized a naïve Bayes algorithm, trained on historic data, to prioritize cycle counting. Inventory accuracy can also be improved by automating the counting using ML techniques for computer vision [25], a method for inventory monitoring [26]. Villegas et al. [27], in particular, used a computer vision platform with image capture and an NN to recognize products and automate the counting process. Similarly, Ye et al. [28] employed an NN to recognize and count assets following the detection of object images. Some studies utilized ML in a non-cycle counting environment to more broadly manage inventory inaccuracy. Andrade et al. [29] forecasted inventory inaccuracy with the help of gradient boosting. Shen et al. [30] conducted a case study on an inventory management system that utilized artificial intelligence (AI) to reconcile discrepancies between two different information systems.

This analysis shows how the application of ML and MCIC techniques to cycle counting remains understudied. Table 1 compares inventory classification papers that either apply MCDM or ML methods to cycle counting; no study presents all three features. Our research addresses this gap by evaluating and identifying the best ML algorithms for cycle counting, considering multiple models trained on MCDM input data. Furthermore, our study directly quantifies the economic impact of inventory classification on cycle counting by incorporating both counting-related and downtime costs in a monetary evaluation.

Table 1. Comparison of papers.

Paper	MCDM	Cycle Counting	Real Data	ML	Monetary Approach
This study	✓	✓	✓	✓	✓
[24]	✗	✓	✓	✓	✗
[16]	✓	✓	✓	✗	✗
[9]	✓	✓	✓	✗	✗

3. Materials and Methods

This section describes the methodological framework adopted in the study. First, we introduce the ML procedure designed to replicate the AHP-K classification and to identify the most suitable algorithm for inventory classification in cycle counting in order to prove that in the future the ML could be use as a standalone method. Then, we present the costing method developed to evaluate the economic impact of different inventory classification. By combining these two components, the classification and cost analysis, the proposed approach allows both the technical validation of ML-based methods against traditional MCDM techniques and the assessment of their practical implications in terms of financial outcomes.

3.1. Machine Learning Method

In this section, we outline the ML procedure, Figure 1, to select the best method to replicate AHP-K results [15]. Commonly used criteria in ABC classification include rotation index, error rate, material value, supply chain length, lead time (LT), storage costs, bill of materials (BOM), criticality, procurement strategy, demand variability, and cost of use (CU) [31]. In this study, we selected the following:

- LT;
- CU;
- Number of work centers involved in the processing of SKUs.

Although various structured approaches for selecting classification criteria exist [32], ours were chosen based on their relevance to the research context and to the decision-maker, to minimize downtime caused by poorly organized cycle counting. The first criterion represents the time required for the suppliers to deliver raw materials or for production to be completed. Longer production LT can complicate recovery efforts during shortages, as they extend operational processes and impact multiple related activities. The second is the total demand recorded over a three-year reference period multiplied by the standard purchase cost of the SKU. This criterion is particularly important because materials with higher consumption volumes are more prone to handling errors. Additionally, the unavailability of high-cost items can significantly disrupt the production of finished goods. Although many companies rely solely on CU for inventory classification, it primarily reflects a management control perspective rather than a logistics-oriented approach. The third aims to identify how many work centers could experience downtime due to the unavailability of SKUs. If an SKU is needed in multiple work centers, the probability of stopping the production line increases. Considering additional criteria could introduce unnecessary redundancy and complexity, making it harder to compare and gather data effectively.

After selecting the criteria, AHP-K is applied to classify the SKUs into three classes (A, B, and C), as illustrated in Figure 1. Each of the n SKUs is, thus, associated to three criteria and a class; we, then, select 100 SKUs from this complete dataset. The selected SKUs are divided into a training and validation set (80% of the total) and a test set (the remaining 20%). The training and validation set is also divided into three folds; two of which, in turn, are used together in the cross-validation process to train each ML model, while the remaining one is used to assess its CA. The method with the best average CA over the three folds is retrained over the whole training and validation set and is used to predict the test set classes. The resulting CA is an unbiased estimate of the performance of the selected ML model. The optimal classification model, referred to as ML-AHP-K, is retrained using all 100 selected SKUs and can, subsequently, be applied to predict the classification of the non-selected SKUs. We used 100 SKUs to test the method effectiveness on small dataset. This design choice addresses a common challenge in industry: the scarcity of high-quality labeled data and abundance of unlabeled operational data. A Veto method can also be applied to the ML-AHP-K results to obtain ML-AHP-K-VETO; this is the ML equivalent to the original AHP-K-VETO model.

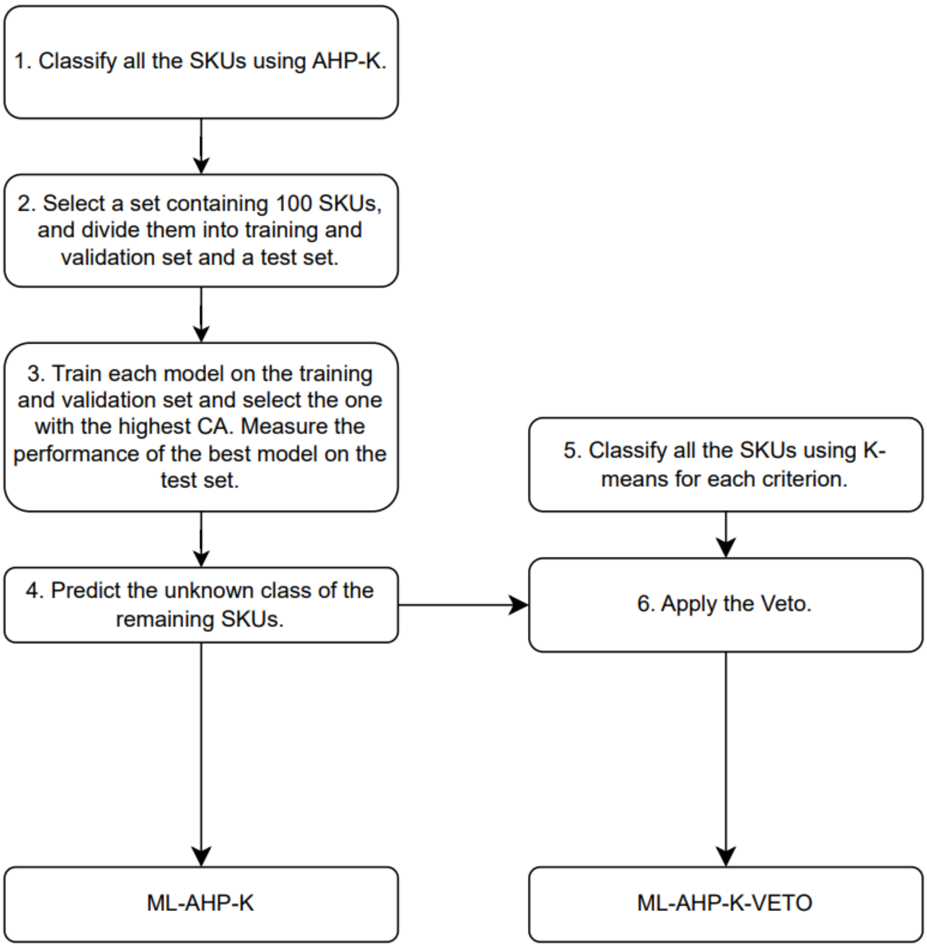

Figure 1. Steps of ML-AHP-K and ML-AHP-K-VETO.

In the original AHP-K method, outlined in Figure 2, AHP is applied to the initial SKUs dataset to obtain a score for each SKU. The K-means algorithm is, then, used to classify them into three classes. Moreover, to obtain the AHP-K-VETO, a K-means clustering algorithm is, instead, individually applied to each criterion. The results of these classifications are mixed with the AHP-K ones to obtain a more nuanced classification. In the next subsection, the AHP-K-VETO will be explained in detail.

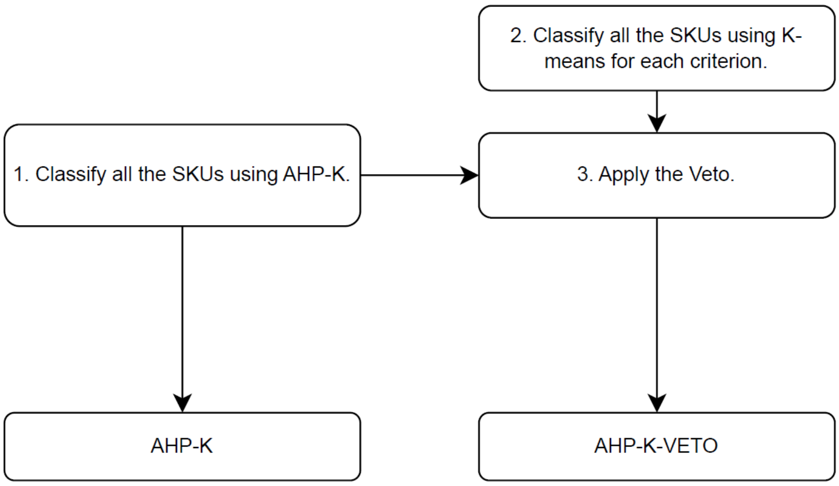

Figure 2. Steps of the original AHP-K and AHP-K-VETO.

AHP-K-VETO

The following section explains the methods from the literature that were incorporated into our methodology, which are AHP, K-means, and VETO.

According to [33], AHP enables the classification of items based on multiple criteria, whose weights are derived through pairwise comparisons using a square, reciprocal matrix A, constructed subjectively [34]. Each matrix element a_{ij} is the weight of criterion i relative to criterion j, and, therefore, its reciprocal $1/a_{ij}$ is the weight of criterion j relative to criterion i. Given the m criteria, the number of comparisons required is, therefore, $(m \cdot (m-1)/2)$. The ratio between criteria is evaluated using an importance scale [33], where

- 1 means that the criteria i and j have equal importance;
- 3 means that the criterion i has moderate importance compared to criterion j;
- 5 means that the criterion i has strong importance compared to criterion j;
- 7 means that the criterion i has very strong importance compared to criterion j;
- 9 means that the criterion i has extreme importance compared to criterion j.

A matrix is consistent when $a_{ij} \cdot a_{jk} = a_{ik}$. Evaluating three criteria, for example, a decision-maker must determine the relative importance of criterion 1 compared to criteria 2 and 3, as well as criterion 2 compared to criterion 3. In this example, criterion 1 is assigned an importance of 5 relative to criterion 2, and an importance of 3 relative to criterion 3. In this case, the consistency condition is not satisfied, as $5 \cdot 0.33 \neq 3$. For a consistent matrix A, there exists a single eigenvector $\overrightarrow{w}$ such that $A \cdot \overrightarrow{w} = m \cdot \overrightarrow{w}$, where m is the number of criteria. In contrast, when A is not consistent, multiple eigenvectors $\overrightarrow{w_1}, \overrightarrow{w_2}, \ldots, \overrightarrow{w_m}$ exist. The criteria weights are obtained in practice by computing the normalized eigenvector $\overrightarrow{w}$ associated with the largest eigenvalue. Such an eigenvalue can also be used to obtain a consistency indicator (CI):

$$CI = \frac{\lambda_{\max} - m}{m - 1} \tag{1}$$

to be compared with a random indicator (RI) tabulated in Table 2. As m increases, the RI also increases, reflecting the growing complexity in pairwise comparisons. Finally, a consistency ratio (CR),

$$CR = \frac{CI}{RI}, \tag{2}$$

is obtained, where a low CI indicates that the matrix is close to being consistent and a CI of zero shows perfect consistency. A CR value less than 0.10 is considered acceptable [35].

Table 2. Random index (RI) values for different matrix sizes m.

m	3	4	5	6	7	8	9	10
RI	0.58	0.9	1.12	1.24	1.32	1.41	1.45	1.49

After applying the AHP and obtaining the overall score for each SKU, we need a method to group them into three classes: K-means. K-means is an unsupervised ML clustering algorithm that groups SKUs into K clusters based on their similarity [36]. The algorithm relies on the Euclidean distance between each SKU x_i and the centroid x'_k of cluster k; a smaller Euclidean distance between two SKUs leads to a higher probability that they will be assigned to the same cluster, ensuring that SKUs with comparable managerial implications are not separated. The algorithm proceeds through the following steps:

1. Normalization. SKUs features are normalized.
2. Initialization. Initial centroids x'_k are randomly assigned for all K clusters.
3. Assignment. Each SKU is assigned to the cluster C_k with the closest centroid:

$$C_k = \{x_i : ||x_i - x'_k|| \leq ||x_i - x'_j|| \quad \forall j \in 1, \dots, K\}. \tag{3}$$

4. Centroid update. The centroid of each cluster is recalculated as the mean of the SKUs assigned to it:

$$x'_k = \sum_{x_i \in C_k} x_i. \tag{4}$$

The steps are repeated until no SKU is reassigned in the assignment step. This algorithm typically produces spherical clusters separated by clear boundaries and, due to its computational efficiency and simplicity, it is widely used in manufacturing and industrial contexts.

The last method used in our study is VETO, as proposed in [15]. While K-means generates clusters for each criterion individually, the VETO method addresses a limitation of AHP: the possibility of compensation effects, where an item with a low score in one criterion may still be classified into class A due to high scores in others. A K-means is computed for each individual criterion to integrate the AHP-K classification. The VETO rule enforces that no item with a K-means classification of A on a single criterion can be assigned to class C by AHP-K, and vice versa. Such items are, instead, assigned to the intermediate class B to ensure a more balanced and criterion-sensitive classification.

3.2. Costing Method

This section presents the methodology used to quantify the costs associated with the classification of SKUs into different inventory classes. Specifically, it details the calculation of both downtime costs, caused by production interruptions due to missing SKUs, and counting costs, linked to the inspection activities performed during cycle counting. By combining these cost components, the method provides an overall monetary evaluation of the different classification approaches, allowing a comparative analysis in terms of their financial impact within an industrial context. The procedure used to measure the costs associated with an ABC classification is the following:

1. Calculate the downtime cost for a production line blocked by the absence of SKUs:
   - 1.1. Given an SKU, compute the number of work centers where it is processed.
   - 1.2. Calculate the cost as if all those work centers stopped.
   - 1.3. Each class is associated with a probability that an SKU in that class could be missing; multiply that probability by the cost computed in the previous step to obtain the unitary downtime cost for that SKU.

  1.4. This unitary downtime cost is then multiplied by the class-dependent number of counts performed annually. We assume that after each one, there is only one downtime opportunity, and that downtimes do not happen simultaneously.
  1.5. The downtime costs for all the SKUs are added to obtain an overall downtime cost.
2. Calculate the counting cost:
  2.1. Given the cost to count an SKU.
  2.2. Multiply the unit counting cost by the number of counts performed annually, which is determined by the SKU class.
  2.3. The counting costs for all SKUs are added to obtain an overall counting cost.
3. Add the total downtime cost to the total counting cost.

4. Case Study

To validate the proposed methodology, a real-world case study is conducted in collaboration with a mechanical manufacturing company. The case study serves a dual purpose, in order to prove that pure ML can be used in the future: on one hand, it provides empirical ground to test the accuracy and robustness of the ML procedure compared with traditional MCDM and Pareto approaches; on the other hand, it allows the evaluation of the economic consequences of different cycle counting policies in an industrial environment. The dataset, consisting of 12,863 SKUs, includes items with heterogeneous characteristics in terms of LT, CU, and number of work centers involved, thus offering a realistic and challenging context for inventory classification.

First, the AHP-K method is applied to generate the reference classification; second, different ML algorithms are trained and validated on a subsets of SKUs to replicate the AHP-K results; third, the VETO method is applied to both MCDM- and ML-based classifications. Finally, the economic impacts of each classification are computed by considering both downtime and counting costs. This structured approach ensures not to limit the comparison to statistical performance, but also to account for the practical managerial implications of cycle counting.

4.1. Design Phase

In this subsection, we present the data used for both the ML method and the monetary method. A dataset from a mechanical manufacturing company, containing 12,863 SKUs, is used to compare the following:

- AHP-K;
- ML-AHP-K;
- AHP-K-VETO;
- ML-AHP-K-VETO;
- a traditional Pareto method;

from both a CA and a cost perspective.

In order to apply these methods, data for each criterion selected for every SKU was collected. The company operates in a sector where the supply chain is strongly dependent on both local and international suppliers, which increases the relevance of LT variability. LT varies significantly depending on the product and supplier, ranging from 0 to 210 days, with 45 days being the most frequently observed value. Internally supplied items have an average LT of 16 days, although this depends on the specific product type. For internally produced materials, LT is determined by analyzing families of finished products. CU, on the other hand, measures consumption demand by multiplying the total demand recorded over a three-year reference period by the standard purchase cost of each SKU. CU values range from 0 to 1,690,355. The third criterion reveals that raw materials are generally processed in a larger number of machining centers, semi-finished products in fewer, and finished products often in just one or none. The SKU with the highest number of associated centers is linked to 23 out of the 30 machining centers. The dataset was preprocessed by replacing missing criteria values with their corresponding means, analyzing product families, and substituting zero values with 0.001 to enable AHP normalization, preventing division by zero errors. Obsolete SKUs, newly introduced, prototypes, or not purchased in the last three years were excluded from the database.

Firstly, the AHP-K method is applied using Table 3 to calculate the criteria weights, which resulted in 0.64 for LT, 0.26 for the number of work centers, and 0.10 for CU. The CR resulted in 4%. These are the preferences between the three criteria that were chosen by the decision-maker in the company.

Table 3. Preferences used in AHP-K to compute the criteria weights.

	LT	CU	Number of work centers
LT	1.00	5.00	3.00
CU	0.20	1.00	0.33
Number of work centers	0.33	3.00	1.00

The AHP-K results in 2219 SKUs in class A, 6202 in class B, and the remaining 4442 in class C. From the original 12,863 SKUs, two different datasets of 100 SKUs are obtained. Table 4 presents the classification of SKUs into three categories (A, B, and C) according to the AHP-K method, which will be used for training and validation of the ML models. Dataset 1 is designed to be balanced, containing almost the same number of SKUs in each of the three classes to ensure equitable representation during model training. Dataset 2, instead, reflects the actual distribution proportions of the classes within the full industrial dataset, with a larger share of SKUs in class B and fewer in class A, thus representing the real-world data imbalance. The distribution differences between the two datasets allow an analysis of how balanced versus realistic data sampling can impact model performance.

Table 4. AHP-K classification of dataset 1 and 2 SKUs.

	SKUs	
Class	**Dataset 1**	**Dataset 2**
A	37	18
B	32	46
C	31	36
Tot	100	100

The ML methods compared in the ML procedure (Section 3.1) are the following:

- Decision tree.
- Random forest.
- SVM.
- NN.
- AdaBoost.
- KNN.
- Naïve Bayes.
- Gradient boosting.
- Logistic regression.

These models were used starting from the AHP-K classification and the two datasets created. All ML training and validation experiments were implemented using the software Orange 3.36 using threefold cross-validation. The ML models used a set of parameters as described in the following. For NN, a three-layer architecture with ReLU activation functions and Adam optimizer was adopted, with a learning rate of 0.001, and training stopped after 200 iterations. For SVM, a radial basis function (RBF) kernel was used, with a regularization parameter $C = 1.0$, regression loss epsilon 0.1 and a kernel coefficient $\gamma = scale$. The random forest classifier was configured with 10 trees, Gini index as the splitting criterion, and a maximum tree depth of five. The decision tree used the Gini index with a maximum depth of 100 to avoid overfitting. For KNN, the number of neighbors was set to $k = 5$ with the Euclidean distance as a metric. Logistic regression was implemented with an L2 penalty and a regularization strength of $C = 1.0$. Naïve Bayes was applied in its Gaussian form, with prior probabilities estimated directly from the data. AdaBoost was used with 50 estimators and a learning rate of 1.0, while gradient boosting had 100 trees with a learning rate of 0.1. K-means clustering was applied using three clusters, initialized with the k-means++ method, with 10 reruns and a maximum of 300 iterations.

A traditional mono-criteria Pareto method is also used as a comparison to the MCDM and ML ones. The SKUs are ranked by CU, and the first ones are associated with class A to reach 80% of the cumulative CU, the following ones are associated with class B to reach 95% of the cumulative CU (between the two classes), and the remaining ones are associated with class C. Table 5 outlines the number of SKUs in each class, resulting in 1616 SKUs in class A, 2516 SKUs in class B, and 8731 SKUs in class C.

Table 5. Pareto method classification.

Class	SKUs	Percentage of SKUs	Percentage of CU
A	1616	13%	80%
B	2516	20%	15%
C	8731	68%	5%
Tot	12,863	100%	100%

A monetary method needs to be applied, and our case study assumptions are as follows:

- The number of missing SKUs varies by class and is equal to 0.5% for class A, 1% for class B, and 5% for class C, as suggested in [10,11,37].
- The downtime cost is equal to one hour of lost work per affected SKU.
- The counting costs are equal to EUR 1.50 per SKU counted.
- The work centers are divided into three value streams, and their downtime cost is associated with the value stream they belong to:
  - 110 $\frac{\text{EUR}}{hour \cdot work\,center}$ for value stream 1.
  - 160 $\frac{\text{EUR}}{hour \cdot work\,center}$ for value stream 2.
  - 90 $\frac{\text{EUR}}{hour \cdot work\,center}$ for value stream 3.

  This data has been provided by the company, based on the profitability of each value stream.
- The number of counts for an SKU in each class, as suggested in [1,11], is
  - 4 times for class A.
  - 2 times for class B.
  - 1 time for class C.

Starting from these assumptions, both costs will be calculated and the results compared.

4.2. Classification Results

This section presents the results of applying the proposed ML procedures and the MCDM classification methods for cycle counting to the industrial dataset. The performance of the various ML models is evaluated in terms of CA across different training and validation datasets. Moreover, the predictive capability of the best performing models is examined on independent test sets. Quantitative results are systematically compared to assess the ability of ML approaches to replicate traditional MCDM classifications, providing insight into their practical applicability in cycle counting scenarios as pure potential methods.

Based on the design phase and the procedure outlined in Section 3.1, the following results were achieved. The cross-validation results, summarized in Table 6, indicate that, for dataset1, SVM, NN, Adaboost reached the best CA, while for dataset 2, they are decision tree, SVM, NN, Adaboost, and gradient boosting. The ML models, NN for dataset1 and SVM for dataset 2, were chosen among the highest performance models, reaching values up to 96.2%. The choice of one model among the others with the same CA is based on the fact that they are equally effective.

Table 6. Dataset 1 and 2 cross-validation CA results.

	Dataset 1	**Dataset 2**
Decision tree	0.846	0.923
Random forest	0.923	0.769
SVM	0.962	0.923
NN	0.962	0.923
AdaBoost	0.962	0.923
KNN	0.500	0.538
Naïve Bayes	0.846	0.808
Gradient boosting	0.923	0.923
Logistic regression	0.346	0.692

In the test set, those methods obtained a CA of 0.818 with the NN method in dataset 1, and of 1 with the SVM method in dataset 2. Using these selected methods, predictions are made to classify the remaining SKUs in the dataset. Table 7 shows the predicted classes, in terms of both SKUs quantity and percentages, for all the 12,863 SKUs of the industrial dataset. These were obtained with ML-AHP-K, ML-AHP-K-VETO, and the application of VETO to the AHP-K method, the AHP-K-VETO.

Table 7. Classification of 12,863 SKUs using different methods.

	SKUs					
		ML-AHP-K		**ML-AHP-K-VETO**		
Class	**AHP-K**	**Dataset 1**	**Dataset 2**	**Dataset 1**	**Dataset 2**	**AHP-K-VETO**
A	2219	1260	181	123	6	143
B	6202	6809	8197	7982	9078	8296
C	4442	4794	4485	4758	3779	4424
Tot	12,863	12,863	12,863	12,863	12,863	12,863
Class:						
A	17%	10%	1%	1%	0%	1%
B	48%	53%	64%	62%	71%	64%
C	35%	37%	35%	37%	26%	34%
Tot	100%	100%	100%	100%	100%	100%

4.3. Results Comparisons

In Table 8, we compare the results of the original AHP-K with the new ML-AHP-K in all 12,863 SKUs, explained in Section 4.2; each row contains the fraction of SKUs true positive and false positive, the F1 score, and their overall accuracy. Based on dataset 1, the NN reached a true positive rate of 72% of SKUs in class A, 82% in class B, and 93% in class C. SVM, starting from dataset 2, reached a true positive rate of 43% SKUs in class A, 72% in class B, and 83% in class C. Then, we calculated the F1 score for each class and ML method, which resulted in 0.72 for NN in class A, 0.81 in class B, and 0.92 in class C; while the F1 score for SVM resulted in 0.42 in class A, 0.71 in class B, and 0.82 in class C. In total, NN predicted 85% SKUs and SVM predicted 75% SKUs. Dataset 1 shows a higher CA as its results align more closely with those of AHP-K, as shown in Table 7. The CA of SVM is lower than that obtained in the test sets; a similar trend was observed in [13], where the prediction error increased from 0.3 to 0.7 after the introduction of new data.

Table 8. Comparison between AHP-K and ML-AHP-K in terms of the fraction of SKUs in common.

Class	Dataset 1 (NN)		Dataset 2 (SVM)	
	True Positive	**False Positive**	**True Positive**	**False Positive**
A	72%	28%	43%	57%
B	82%	18%	72%	28%
C	93%	7%	83%	17%
F1 class A	0.72		0.42	
F1 class B	0.81		0.71	
F1 class C	0.92		0.82	
Tot in common	85%		75%	

When comparing the two Veto methods (AHP-K-VETO and ML-AHP-K-VETO), as shown in Table 9, their similarity increases as both datasets achieve a CA higher than 90%, specifically 97% for NN and 93% for SVM. Based on dataset 1, NN reached a true positive rate of 98% of SKUs in class A, 100% in class B, and 93% in class C. SVM, starting from dataset 2, reached a true positive rate of 83% SKUs in class A, 91% in class B, and 98% in class C. Then, we calculated the F1 score for each class and ML method, which resulted in 0.97 for NN in class A, 0.99 in class B, and 0.93 in class C. Meanwhile, the F1 score for SVM was 0.83 in class A, 0.91 in class B, and 0.98 in class C. The increase in CA is because, with the VETO approach, more SKUs are classified in class B, as shown in Table 7. Applying VETO is crucial as it eliminates compensation effects that may arise from the global classification of multiple criteria.

Table 9. Comparison between AHP-K-VETO and ML-AHP-K-VETO in terms of the fraction of SKUs in common.

Class	Dataset 1 (NN VETO)		Dataset 2 (SVM VETO)	
	True Positive	**False Positive**	**True Positive**	**False Positive**
A	98%	2%	83%	17%
B	100%	0%	91%	9%
C	93%	7%	98%	2%
F1 class A	0.97		0.83	
F1 class B	0.99		0.91	
F1 class C	0.92		0.98	
Tot in common	97%		93%	

The comparative analysis between the original AHP-K and ML-AHP-K methods reveals a substantial alignment in the classification outputs, particularly when employing the VETO methods, which achieve classification accuracies above 90%. The general consistency supports the viability of ML models as reliable replicators of traditional MCDM approaches to use in cycle counting classifications. This aligns with the emerging evidence that ML techniques can effectively complement and potentially replace classical MCDM methods in large-scale inventory classification tasks, offering advantages in time savings [20].

In Figure 3, we chose to analyze for the best method, NN and NN VETO, which SKUs are misclassified and whether classification errors cluster with respect to the criteria. The first plot reports the number of work centers on the x-axis and LT on the y-axis. The second plot reports CU on the x-axis and LT on the y-axis. The third plot reports CU on the x-axis and the number of work centers on the y-axis. In each plot, SKUs correctly classified with respect to the AHP-K benchmark are shown as dots, whereas misclassified SKUs are shown as crosses. Across all six views, no clear clustering pattern of errors emerges.

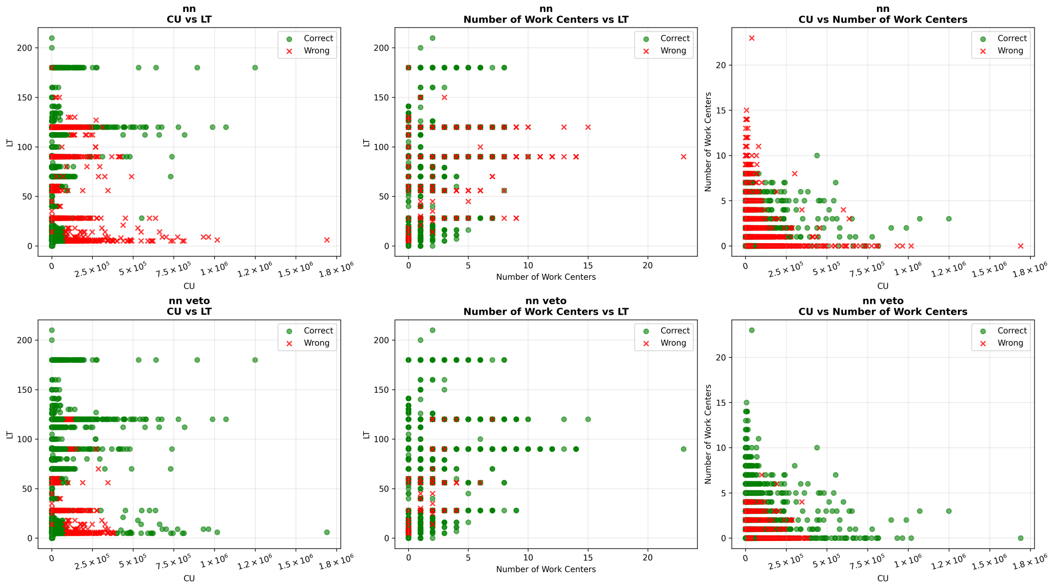

Figure 3. SKUs correct and misclassified by criteria, NN and NN Veto.

4.4. Monetary Results' Comparisons

We apply the Section 3.2 method to compare the results of previous sections from a monetary point of view in the cycle counting operations. The downtime cost is inversely proportional to the counting cost: if the counting cost is higher, we expect a lower downtime cost, while if the operator counts the SKUs less frequently, the alignment will be lower and the downtime cost associated will be higher. If the materials were not counted and then aligned, there would be a higher percentage of breakage in the production line. Figure 4 compares downtime costs among the Pareto method, MCDM methods, and ML methods, with the ML results further divided by two datasets. The total costs amount to approximately EUR 36,000, except when using the Pareto method, in which case the costs almost double. Pareto as expected, resulted in higher downtime costs because its classification has the highest amount of SKUs in class C, which produces a higher probability of having a missing item. The lowest costs are observed with the ML-AHP-K-VETO, AHP-K, and AHP-K-VETO methods.

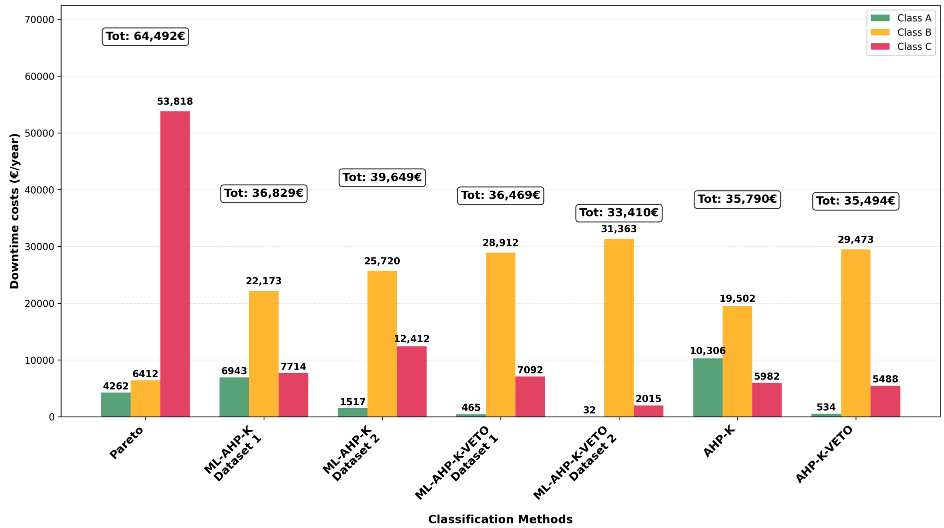

Figure 4. Downtime costs for each method.

Figure 5 gives the counting costs for the same methods, such as Pareto, ML-AHP-K, ML-AHP-K-VETO, AHP-K, and AHP-K-VETO. The lowest counting costs are observed with Pareto and ML-AHP-K-VETO.

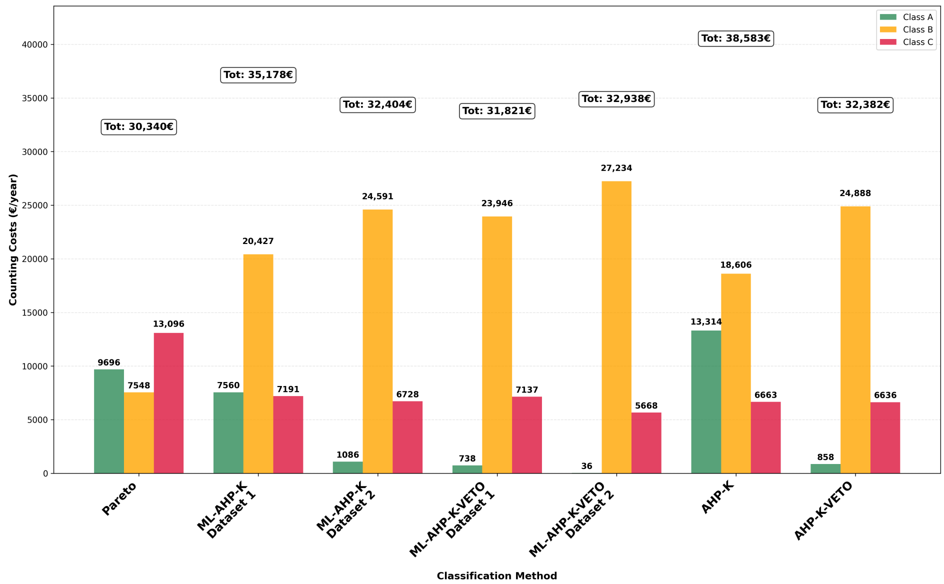

Figure 5. Counting costs for each method.

Adding the costs outlined in both tables, as shown in Figure 6, reveals that the Pareto method is the least effective, with an annual cost 43% higher than the most effective alternative (ML-AHP-K-VETO). The results for different methods following similar procedures are very close to each other; the ML ones, without Veto, fall within 3% of the AHP-K one, regardless of the dataset. The ML Veto ones are even closer to the non-ML counterpart, AHP-K-VETO, with a maximum gap of 2%. This shows that, independent of the CA results, the ML methods can substitute for the MCDM ones without any actual monetary impact. This suggests that companies can confidently adopt ML-based procedures for cycle counting, benefiting from faster computation and greater scalability [20] without incurring financial risks.

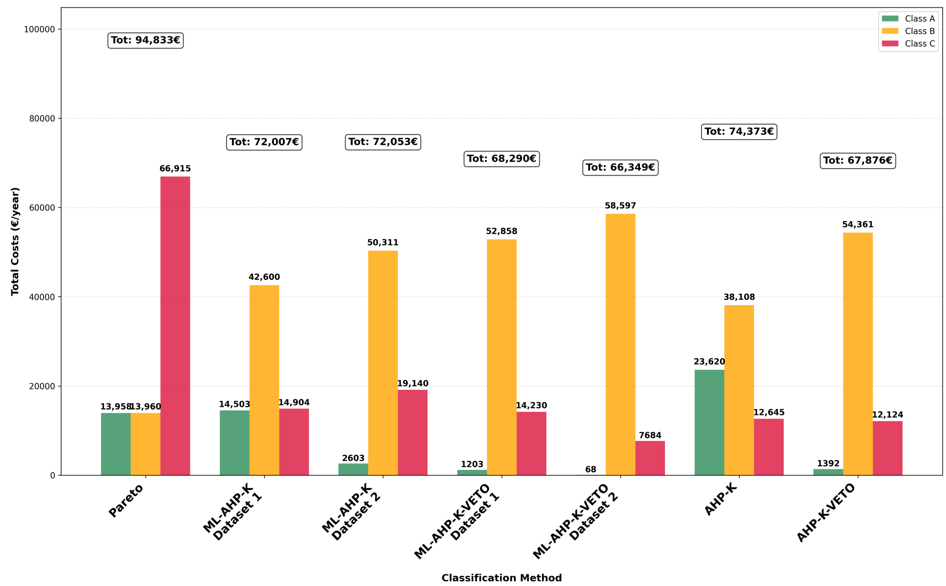

Figure 6. Total costs for each method.

5. Discussion

This study investigates IRI and its implications for cycle counting, comparing traditional MCDM methods with ML approaches from both accuracy and monetary perspectives. We tested the applicability of multiple ML methods to ensure that they can be applied to expand their use. Using an industrial dataset comprising 12,863 SKUs, the ML-AHP-K approach achieved a maximum CA of 85% when replicating the AHP-K method (Table 8). This performance improved significantly, by up to 97%, with the application of the ML-AHP-K-VETO method when replicating the AHP-K-VETO method, as shown in Table 9. Specifically, the VETO method contributed to an increase in the number of SKUs correctly classified in class B, as illustrated in Table 7. With the VETO application, the F1 score for class B increased from 0.81 to 0.99 for NN and from 0.71 to 0.90 for SVM. Moreover, among the nine ML models evaluated, based on dataset 1 and dataset 2, NN and SVM achieved the highest accuracy on the training set, 0.962 and 0.923, respectively, as reported in Table 6. These results are aligned with the findings of previous studies from other fields [12,13] where the same ML models achieved good results. Kartal et al. [12] defined SVM as one of the best models for replicating the SAW, AHP, and Vikor methods, while Kabir et al. [13] noted that NN exhibits a low mean absolute percentage error (MAPE) when replicating fuzzy AHP. A variation in accuracy was observed between the training and test sets. In dataset 1, the NN achieved a training CA of 0.962, but its test accuracy dropped to 0.818. In contrast, for dataset 2, the SVM recorded a training CA of 0.923 and reached a test accuracy of 1. Similar trends have been reported in other studies; for instance, ref. [20] achieved a training CA of 69% and a test CA of 82% using SVM for TOPSIS prediction. Another situation was observed in [13], where the mean absolute percentage error (MAPE) increased between the training and test sets. A comparison between the two datasets used for training also revealed that dataset 1 yielded superior CA results, as shown in Tables 8 and 9; notably, its construction adheres to the proportions defined by the AHP-K classification, as shown in Table 7. The predictions made on the entire dataset of 12,863 SKUs used in this study yielded lower accuracy for SVM values compared to those observed in the training and test phases, dropping from approximately 90% to 75%, while remaining almost stable for NN around 85%.

The monetary results showed that the MCDM and ML approaches to cycle counting outperformed Pareto in all scenarios, as shown from the overall costs in Figure 6. Pareto showed the worst results, as it classifies most SKUs in class C, causing high downtime costs. The most favorable monetary result was achieved by the ML-AHP-K-VETO method. The second-best result came from the non-ML counterpart, AHP-K-VETO, which closely matched the cost performance of the previous one, showing a gap of only 2%. Additionally, the methods without the VETO component produced similar results with slightly higher costs. This proves that ML methods can effectively substitute traditional MCDM ones in cycle counting scenarios.

6. Conclusions

This study explores the use of MCDM and ML classification methods for cycle counting. MCDM has been applied to SKU classification for cycle counting in only two articles [9,16], with a traditional MCDM method, while ML has only been applied once [24]. We extended this line of research by comparing different ML methods with AHP and AHP-K to assess the potential for a pure ML implementation, leveraging DM preferences, in a cycle counting context. Our results demonstrate that ML models, when trained on AHP-derived multi-criteria classifications, outperform traditional ABC methods in cycle counting prioritization, achieving up to 43% cost reduction on a real-world dataset of 12,863 SKUs. Unlike prior studies, our comprehensive economic analysis reveals the full monetary impact of classification decisions, including both counting labor and downtime costs. The most effective monetary outcome, based on an industrial dataset of 12,863 SKUs, was achieved by the ML-AHP-K-VETO method using an SVM; this technique had the lowest total cost, outperforming its non-ML counterpart and the other methods tested. Most notably, we prove that certain ML algorithms can reliably replicate complex DM preferences embedded in MCDM frameworks. The ML-AHP-K-VETO method using an NN reached the best CA of 97% when compared with the AHP-K-VETO.

This study provides actionable guidance for companies managing large product assortments (e.g., >10,000 SKUs), demonstrating how to implement ML-enhanced multi-criteria classification for cycle counting to minimize both downtime costs from inventory discrepancies and personnel costs from counting activities. The implementation steps are

1. Data preparation. Collect historical inventory data on the criteria selected and elicit DM preferences via AHP pairwise comparisons.
2. Model training. Apply AHP-K-VETO classification to label training data, then train an NN or SVM model.
3. Deployment. Replace traditional ABC with ML-AHP-K-VETO, generating cycle count schedules.
4. Monitoring. Track classification accuracy and cost savings monthly, retraining models quarterly with new data.

The expected operational changes are

- Resource reallocation. Reduced cycle counting frequency, particularly for SKUs in class A; the personnel can be assigned to other value-added activities.
- Cost reduction. Achieved up to 40% lower total costs from the Pareto approach (as demonstrated on 12,863 SKUs).
- Improved service levels. Higher inventory accuracy (up to 97% CA) and minimized stockouts.

The implementation risks are

- Data quality. Incomplete historical data can make the NN model less effective.
- Changing business conditions require new data and the assignment of different importance values to the criteria.

The limitations of this study are its focus on a single MCDM technique, AHP-K, and its Veto alternative, and the use of a limited (100 SKUs) training, validation, and test set. Future research could explore additional decision-making techniques, such as the TOPSIS, as well as other hybrid ML-MCDM models [17], or the use of criteria, in addition to LT, CU, and the number of work centers. Moreover, spare parts are important to manage [38] and could be particularly managed with cycle counting. We aim to test more explainable methods in the future. Other weighting methods, such as the entropy weighting method, the method based on the removal effects of criteria (MEREC), and the Gini coefficient, can be explored. In this study, CA was used as the primary validation metric, but alternative measures, such as area under the curve (AUC), could provide further insights into model performance [39]. Other robust validation strategies should also be considered. Further research could investigate the use of ML for inventory classification during cycle counting, utilizing various algorithms to develop a pure ML approach.

Author Contributions: Conceptualization, F.L. and R.G.; methodology, E.B. and R.G.; software, L.V.; validation, E.B. and F.L.; formal analysis, E.B. and L.V.; investigation, L.V.; resources, L.V.; data curation, L.V.; writing—original draft preparation, L.V. and E.B.; writing—review and editing, L.V. and E.B.; visualization, L.V. and R.G.; supervision, F.L.; project administration, F.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Although the industrial dataset cannot be shared due to confidentiality agreements, an anonymized version will be made available upon request.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

AHP	Analytic Hierarchy Process
BOM	Bill of Materials
CA	Classification Accuracy
CU	Cost of Use
IRI	Inventory Record Inaccuracy
KNN	K-Nearest Neighbors
LT	Lead Time
MCIC	Multi-Criteria Inventory Classification
MCDM	Multi-Criteria Decision-Making
ML	Machine Learning
NN	Neural Networks
SAW	Simple Additive Weighting
SKUs	Stock-Keeping Units
SVM	Support Vector Machines
TOPSIS	Technique for Order of Preference by Similarity to Ideal Solution
VIKOR	VlseKriterijumska Optimizacija I Kompromisno Resenje
ETO	Engineering To Order
RBF	Radial Basis Function
ERP	Enterprise Resource Planning
CI	Consistency Indicator
RI	Random Indicator
CR	Consistency Ratio

MAE	Mean Absolute Error
MADRL	Multi-Agent Deep Reinforcement Learning
RFID	Radio Frequency Identification

References

1. Agrawal, P.M.; Sharda, R. Impact of frequency of alignment of physical and information system inventories on out of stocks: A simulation study. *Int. J. Prod. Econ.* **2012**, *136*, 45–55. [CrossRef]
2. Vaccari, L.; Coruzzolo, A.M.; Lolli, F.; Gamberini, R. Resilience in Supply Chains: A Review of Key Metrics, Disruption Models, and Simulation Approaches. In *Emerging Trends in Performance Measurement and Management for Industrial Applications—Proceedings of the Third Conference on Performance Management (COPERMAN)*; Springer Nature Switzerland AG: Cham, Switzerland, 2025.
3. Gumrukcu, S.; Rossetti, M.; Buyurgan, N. Quantifying the costs of cycle counting in a two-echelon supply chain with multiple items. *Int. J. Prod. Econ.* **2008**, *116*, 263–274. [CrossRef]
4. Shabani, A.; Maroti, G.; de Leeuw, S.; Dullaert, W. Inventory record inaccuracy and store-level performance. *Int. J. Prod. Econ.* **2021**, *235*, 108111. [CrossRef]
5. Salehi Shahrabi, M. A quantitative analysis of inaccuracy inventory reducing in multi-period mode: Comparison between RFID and inventory counting. *Comput. Ind. Eng.* **2023**, *177*, 109089. [CrossRef]
6. Wang, F.; Fang, X.; Chen, X.; Li, X. Impact of inventory inaccuracies on products with inventory-dependent demand. *Int. J. Prod. Econ.* **2016**, *177*, 118–130. [CrossRef]
7. Coruzzolo, A.M.; Lolli, F.; Balugani, E.; Magnani, E.; Sellitto, M.A. Order Picking Problem: A Model for the Joint Optimisation of Order Batching, Batch Assignment Sequencing, and Picking Routing. *Logistics* **2023**, *7*, 61. [CrossRef]
8. Wijffels, L.; Giannikas, V.; Woodall, P.; McFarlane, D.; Lu, W. An enhanced cycle counting approach utilising historical inventory data. *IFAC-PapersOnLine* **2016**, *49*, 1347–1352. [CrossRef]
9. Medina, G.C.; Cabello, S.E.; Garcia, Y.S.; Santivanez, G.Q.; Castaneda, E. Inventory Management Model Applying Cyclic Counting by ABC-AHP Classification in a Laboratory Equipment Distribution Company. In Proceedings of the International Conference on Information Management, Cambridge, UK, 25–27 March 2022; pp. 151–154. [CrossRef]
10. Tayi, G.K. Optimal inventory cycle counting. *Omega* **1985**, *13*, 535–539. [CrossRef]
11. Rossetti, M.D.; Collins, T.; Kurgund, R. Inventory cycle counting–A review. In Proceedings of the 2001 Industrial Engineering Research Conference, Dallas, TX, USA, 20–22 May 2001; Volume 1, pp. 457–463.
12. Kartal, H.; Oztekin, A.; Gunasekaran, A.; Cebi, F. An Integrated Decision Analytic Framework of Machine Learning with Multi-Criteria Decision Making for Multi-Attribute Inventory Classification. *Comput. Ind. Eng.* **2016**, *101*, 599–613. [CrossRef]
13. Kabir, G.; Hasin, A. Multi-criteria inventory classification through integration of fuzzy analytic hierarchy process and artificial neural network. *Int. J. Ind. Syst. Eng.* **2013**, *14*, 74–103. [CrossRef]
14. Balaji, K.; Kumar, V.S. Multicriteria Inventory ABC Classification in an Automobile Rubber Components Manufacturing Industry. *Procedia CIRP* **2014**, *17*, 463–468. [CrossRef]
15. Lolli, F.; Ishizaka, A.; Gamberini, R. New AHP-based approaches for multi-criteria inventory classification. *Int. J. Prod. Econ.* **2014**, *156*, 62–74. [CrossRef]
16. Montes, J.; Pérez, A.; Cabel–Pozo, J. *Inventory Record Accuracy Improvement Based on Multicriteria ABC Analysis, RFID Technology and Sustainability: An Agricultural Company Case Study*; Latin American and Caribbean Consortium of Engineering Institutions (LACCEI): Boca Raton, FL, USA, 2025. [CrossRef]
17. Shahmohammad, F.N.; Pourrahimian, Y.; Akbari-Gharalari, N. Synthesizing complexity: Trends, challenges, and future directions in fuzzy-based multicriteria decision-making (FMCDM) methods. *Appl. Soft Comput.* **2024**, *167*, 112362. [CrossRef]
18. Costa, A.P.d.A.; Choren, R.; Pereira, D.A.d.M.; Terra, A.V.; Costa, I.P.d.A.; Junior, C.d.S.R.; Santos, M.d.; Gomes, C.F.S.; Moreira, M.A.L. Integrating multicriteria decision making and principal component analysis: A systematic literature review. *Cogent Eng.* **2024**, *11*, 2374944. [CrossRef]
19. Ali, R.; Hussain, A.; Nazir, S.; Khan, S.; Khan, H.U. Intelligent Decision Support Systems—An Analysis of Machine Learning and Multicriteria Decision-Making Methods. *Appl. Sci.* **2023**, *13*, 12426. [CrossRef]
20. Roy, A.; Islam, M.; Karim, M.; Ahmed, K.A.; Khan, A.R.; Uddin, M.; Xames, M.D. Comparative analysis of KNN and SVM in multicriteria inventory classification using TOPSIS. *Int. J. Inf. Technol.* **2023**, *15*, 3613–3622. [CrossRef]
21. Abidin, M.N.A.Z.; Bazin, N.E.N.; Zebari, D.A.; Rahmat, A.N.; Asaad, R.R. Inventory Categorization Using Multiple Criteria Classification. *J. Soft Comput. Data Min.* **2024**, *5*, 142–151. [CrossRef]
22. Lolli, F.; Balugani, E.; Ishizaka, A.; Gamberini, R.; Rimini, B.; Regattieri, A. Machine learning for multi-criteria inventory classification applied to intermittent demand. *Prod. Plan. Control* **2019**, *30*, 76–89. [CrossRef]

23. Aslam, B.; Maqsoom, A.; Khalil, U.; Ghorbanzadeh, O.; Blaschke, T.; Farooq, D.; Tufail, R.F.; Suhail, S.A.; Ghamisi, P. Evaluation of Different Landslide Susceptibility Models for a Local Scale in the Chitral District, Northern Pakistan. *Sensors* **2022**, *22*, 3107. [CrossRef]
24. Weaver, Z.; Bharadwaj, R. A Machine Learning Approach to Predict Bin Defects in E-commerce Fulfillment Operations. *Commun. Comput. Inf. Sci.* **2024**, *2118*, 105–112. [CrossRef]
25. Panigrahi, S. Comparative Analysis of Approaches for Autonomous Warehouse Stock Counting Solution. In Proceedings of the ISR Europe 2022; 54th International Symposium on Robotics, Munich, Germany, 20–21 June 2022; VDE Verlag: Berlin, Germany, 2022; pp. 285–291.
26. Vaccari, L.; Coruzzolo, A.M.; Lolli, F.; Sellitto, M.A. Indoor Positioning Systems in Logistics: A Review. *Logistics* **2024**, *8*, 126. [CrossRef]
27. Villegas-Ch, W.E.; Maldonado Navarro, A.; Sanchez-Viteri, S. Optimization of inventory management through computer vision and machine learning technologies. *Intell. Syst. Appl.* **2024**, *24*, 200438. [CrossRef]
28. Ye, L.; Zhang, C.; Zhao, D.; Bu, L.; Xu, H. Implementation of Asset Inventory Model Based on Intelligent Algorithm. In Proceedings of the 2023 4th IEEE Global Conference for Advancement in Technology (GCAT), Bangalore, India, 6–8 October 2023. [CrossRef]
29. Andrade, L.A.C.; da Cunha, C.B. Disaggregated retail forecasting: A gradient boosting approach [Formula presented]. *Appl. Soft Comput.* **2023**, *141*, 110283. [CrossRef]
30. Shen, J.; Bu, F.; Ye, Z.; Zhang, M.; Ma, Q.; Yan, J.; Huang, T. Management of drug supply chain information based on "artificial intelligence + vendor managed inventory" in China: Perspective based on a case study. *Front. Pharmacol.* **2024**, *15*, 1373642. [CrossRef]
31. Ishizaka, A.; Lolli, F.; Balugani, E.; Cavallieri, R.; Gamberini, R. DEASort: Assigning items with data envelopment analysis in ABC classes. *Int. J. Prod. Econ.* **2018**, *199*, 7–15. [CrossRef]
32. Saini, R.; Vaidya, O.S.; Venkitasubramony, R.; Daultani, Y. Identifying critical criteria for warehouse performance using machine learning based hybrid methodology. *OPSEARCH* **2025**. [CrossRef]
33. Saaty, R. The analytic hierarchy process—What it is and how it is used. *Math. Model.* **1987**, *9*, 161–176. [CrossRef]
34. de Paula Vidal, G.H.; Caiado, R.G.G.; Scavarda, L.F.; Ivson, P.; Garza-Reyes, J.A. Decision support framework for inventory management combining fuzzy multicriteria methods, genetic algorithm, and artificial neural network. *Comput. Ind. Eng.* **2022**, *174*, 108777. [CrossRef]
35. Burak, S.; Samanlioglu, F.; Ülker, D. Evaluation of irrigation methods in Söke Plain with HF-AHP-PROMETHEE II hybrid MCDM method. *Agric. Water Manag.* **2022**, *271*, 107810. [CrossRef]
36. Balugani, E.; Lolli, F.; Gamberini, R.; Rimini, B.; Regattieri, A. Clustering for inventory control systems. *IFAC-PapersOnLine* **2018**, *51*, 1174–1179. [CrossRef]
37. Morey, R.C. Estimating service level impacts from changes in cycle count, buffer stock, or corrective action. *J. Oper. Manag.* **1985**, *5*, 411–418. [CrossRef]
38. Coruzzolo, A.; Balugani, E.; Gamberini, R. Spare parts management with Additive Manufacturing (AM): A critical review. *IFAC-PapersOnLine* **2022**, *55*, 1159–1164. [CrossRef]
39. Mebirouk, N.; Amrane, M.; Messast, S.; Ayadat, T. Optimizing machine learning model selection for landslide susceptibility mapping: Analysis of similar performance metrics and the critical role of multi-criteria evaluation. *Environ. Sci. Pollut. Res.* **2025**, *32*, 18434–18460. [CrossRef]