

Predicting weighing deviations in the dispatch workflow process: A case study in a cement industry

JÚLIO BARROS^{1,3}, FRANCISCO CUNHA², CÂNDIDO MARTINS², PAULO PEDROSA¹ AND PAULO CORTEZ³

¹DTx - Digital Transformation CoLAB, University of Minho, Guimarães 4800-058, Portugal

²Cachapuz - Weighing & Logistics Systems, Lda, Braga 4701-952, Portugal

³ALGORITMI/LASI, Department of Information Systems, University of Minho, Guimarães 4800-058, Portugal

Corresponding author: Júlio Barros (e-mail: julio.barros@dsi.uminho.pt, julio.barros@dtx-colab.pt).

This work is supported by European Structural and Investment Funds in the FEDER component, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) [Project n° 069716; Funding Reference: POCI-01-0247-FEDER-069716].

• **ABSTRACT** The emergence of the Industry 4.0 concept and the profound digital transformation of the industry plays a crucial role in improving organisations' supply chain (SC) performance, consequently achieving a competitive advantage. The order fulfilment process (OFP) consists of one of the key business processes for the organization SC and represents a core process for the operational logistics flow. The dispatch workflow process consists of an integral part of the OFP and is also a crucial process in the SC of cement industry organizations. In this work, we focus on enhancing the order fulfilment process by improving the dispatch workflow process, specifically with respect to the cement loading process. Thus, we proposed a machine learning (ML) approach to predict weighing deviations in the cement loading process. We adopted a realistic and robust rolling window scheme to evaluate six classification models in a real-world case study, from which the random forest (RF) model provides the best predictive performance. We also extracted explainable knowledge from the RF classifier by using the Shapley additive explanations (SHAP) method, demonstrating the influence of each input data attribute used in the prediction process.

• **INDEX TERMS** Anomaly prediction, Dispatch workflow process, Machine learning, Order fulfilment process, Weighing systems.

I. INTRODUCTION

A. MOTIVATION

The supply chain (SC) represents a complex and unique network composed of several entities, processes and resources [17], [31], [32]. Logistics consists of a set of fundamental SC processes, which aim to plan and coordinate the movement of products in a timely, safe and effective way [15]. Moreover, Logistics management activities comprise inbound and outbound transportation management, fleet management, warehousing, materials handling, order fulfilment, logistics network design, and inventory management, among others. Over the last few years, the introduction of the Industry 4.0 concept and the profound digital transformation of the industry associated with globalization and global market competitiveness has led companies to spend significant time and effort in re-engineering their SC, changing their business processes and technology regarding the implementation of an

integrated SC management. Logistics is one of the crucial factors for SC's success, and when well managed, it can lead an organization to improve its competitive advantage and overall performance.

The global supply chain forum identified eight core processes for supply chain management (SCM), namely customer relationship management, customer service management, demand management, order fulfilment, manufacturing flow management, supplier relationship management, product development and commercialization, and returns management [17], [33]. Furthermore, according to [34], an organization is supported by three core business processes (pillars), namely the product development process, order fulfilment process, and customer service process, as well as several functional entities that support these business processes (foundations), such as marketing, sales, logistics, finance, among others. Thus, this work focuses on improving

the order fulfilment process (OFP), which is a central process regarding the operational logistics flow [33]. OFP aims to deliver products in order to fulfil customer orders at the right time and place, and achieve agility (in terms of efficiency, flexibility, robustness, and adaptability) to deal with uncertainties from internal or external environments [34]. As such, it is a complex process composed of several activities executed by different functional entities [17], [34], which includes three main activities: order management, manufacturing, and distribution (includes the dispatch workflow process, which consists of the logistic flows of products in an industry). In addition, OFP improvement can be achieved by enhancing the operations performed within and between functional entities [34], involving several methods such as optimization, simulation or business analytics, among others.

Our study is motivated mainly by (i) the nonexistence of scientific works that focus on improving OFP, more specifically regarding the problem of deviation in weighings during the dispatch workflow process, (ii) the lack of studies regarding the cement industry supply chain, although the SCs problems are an attracted topic [16], (iii) the scarce consideration of ML techniques in SC management scientific studies [17].

B. RESEARCH OBJECTIVES AND CONTRIBUTIONS

This paper aims to propose an ML approach to predict weighing deviations in the dispatch workflow or vehicle dispatch process to improve the OFP process in the cement industry. The dispatch workflow process is an integral part of the OFP (distribution activity) and represents a crucial process in the cement industry SC. Hence, the occurrence of delays, errors, or weight deviations, which represents an anomaly in the loading of cement bags, directly impacts the OFP and consequently the SC performance, resulting in several losses, including monetary and service level losses. We explore six different ML classification models, applied and evaluated in a real-world case study in the company Cachapuz - Weighing & Logistics Systems, Lda, Portugal. This work was conducted using the cross-industry standard process for data mining (CRISP-DM) methodology [35]. The main contributions can be summarised:

- we proposed a technological ML pipeline that is able to provide value to the prediction of deviation in weighings during the dispatch workflow process;
- we explore and compared six ML classification models (decision tree, random forest, support vector machines, gradient-boosted tree, extreme gradient-boosting tree, and multilayer perceptron);
- we adopt a realistic and robust experimentation setup, in which a rolling window (RW) scheme is adopted for evaluating the ML models during several training and testing iterations;
- we extract explainable knowledge (XAI) from the proposed ML model by using the Shapley additive explanations (SHAP) method;

- we address a real-world case study using real-world supply chain data from a Portuguese weighing and logistics company.

The remaining work in this paper is organized as follows. Section II introduces the related work. Section III formulates the problem under consideration. Section IV describes the industrial case study, materials, data and methods applied to develop the ML models. Section V reports the obtained results. Finally, Section VI highlights the main results, future works and conclusions.

II. RELATED WORK

Nowadays, data-driven approaches such as machine learning (ML), deep learning (DL), and big data analytics (BDA) have been increasingly adopted in supply chain management for improving the logistics decision-making process [20]. Such approaches allow for creating value and competitive advantages for companies and provide several benefits such as an increase in revenues, customer satisfaction, and product quality, among others. Several approaches have been adopted in the literature to tackle problems related to the dispatch workflow process. Indeed, this process consists of the logistic flows of material in the supply chain industry. Moreover, techniques such as optimization, simulation or simulation-based optimization are the main and commonly used approaches regarding the improvement of the such logistic process (see [15], [60]–[62]). Table 1 consists of an overview of literature contributions regarding heavy vehicle weighing processes, which consists of a crucial task related to the problem of percentage deviation in the weighing process. Indeed, the vehicle weighing process is an essential step for society and organizations quotidian [38] once is always necessary to know the exact weight of the products and objects transported [42], and therefore brings economic benefits and service quality [43]. Several scientific contributions have been proposed that applied ML and DL techniques for the vehicle weighing process, although none of those focuses on the improvement of the dispatch workflow process. Thus, to the best of our knowledge, this research work is the first in the literature that focuses on the improvement of the dispatch workflow process or vehicle dispatching process through the prediction of future deviation in weighings of vehicles using ML techniques with the objective of avoiding security consequences and delays in the entire process, which can cause huge monetary and service level losses. Regarding studies associated with the vehicle weighing process, [44] employed an artificial neural network (ANN) for improving bridge weigh-in-motion (B-WIM) systems. The author's proposed approach aims to determine both the gross weight and individual axis weights of vehicles on a bridge (two different types of bridges) using data acquired from WIM systems to train ANN algorithms. [45] proposed a data-driven framework for estimating the rate of over-weighted vehicles (i.e., vehicles with the exceeded weight) on a bridge using data from WIM stations. Afterwards, the results were combined with national bridge inventory (NBI) data to train a

support vector machine (SVM) model for predicting the status of the bridge (bridge deck condition), providing an overall accuracy of 73.57%, micro-recall of 74.58% and micro-recall of 74.58%. Later on, [46] developed an automated field earthmoving quantity statistics (FEQS) framework that applies vision-based deep learning for classifying vehicles as full or empty loads. The authors used a total of 2,454 images (1,588 of full-load trucks and 866 of empty trucks) to train, test, and validate the deep learning convolutional neural networks (CNN) model. In the same year, [47] proposed a probabilistic ML framework for predicting the probability of fatigue failure of bridges related to traffic overloading using the feedforward neural network and the Monte Carlo techniques. [49] explored several ML algorithms such as gaussian naive bayes (GNB), k-nearest neighbour (kNN), SVM, and decision tree for diagnosing problems in truck ore in underground mines. The authors use data related to truck travel time in order to evaluate and determine anomaly situations during the mineral transport process system in underground mines. Following the application of unsupervised learning techniques, [50] developed a study regarding the establishment of truck traffic classification (TCC) groups for pavement mechanistic-empirical design. TCC groups are crucial for designing specific road pavements. The authors used data collected at the WIM station (vehicle class and weight data) to first reduce the high-dimensional traffic features using the PCA technique and then applied the K-means algorithm to establish appropriate TCC groups. [51] addressed the problem of payload distribution in goods transportation vehicles under complex environments. The payload distribution in the truck is fundamental to ensuring its long useful life. The authors developed a novel DL-based approach to predict the pile-up status and payload distribution (PSPD) in trucks on bridges from images using the CNN model and backwards-propagation neural network. [52] proposed a DL-based methodology to identify the transportation vehicle's weight using data extracted from bridge WIM. Recently, [53] focused on applying the DL technique in order to create a system able to estimate the material quantity transported in the uncovered dump trucks. They used a total of 4,884 images for training the VGG16 model (convolutional neural network). Finally, regarding the logistic context, [48] developed a supervised machine learning classifier (random forest) to predict the industry group (farm products, mining materials, chemicals, manufactured goods, and miscellaneous mixed goods) of the carried goods through the anonymous freight movement data (trip and stop sequences) extracted from global positioning system (GPS) of the transportation vehicle. In this work, we proposed a novel approach, as highlighted by the last row of Table 1. In sharp contrast with previous studies (e.g., [44], [45]), we investigate the performance (in terms of predictive power) of ML-based models to predict weighing deviations in the dispatch workflow so that to avoid monetary, logistic, and security consequences, which includes long delays due to blockage of transportation vehicles inside the factory, monetary, and service level losses.

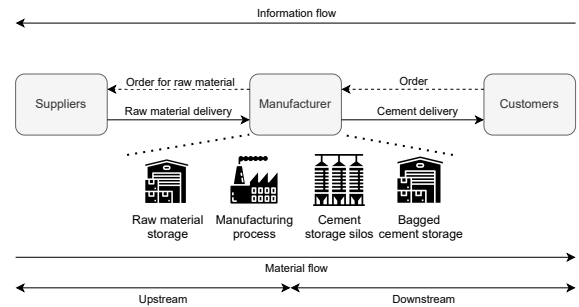


FIGURE 1. Cement industry supply chain network adapted from [16].

We intend to fill several gaps in the literature, namely the lack of studies associated with the cement industry supply chain and the exploration of data-driven approaches using real-world supply chain case study to enhance the OFP by improving the dispatch workflow process, more properly the problem of the percentage deviation of weighing.

III. PROBLEM FORMULATION

Fig. 1 illustrates the considered standard cement industry supply chain topology composed of a single manufacturer with several suppliers and customers, which stands on the supply chain operation reference (SCOR) model processes (Fig. 2).

The SCOR model comprises the following six primary management processes: plan, source, make, deliver, return, and enable. The plan process consists of developing plans to operate the SC, source aims to describe the ordering and receipt of goods and services, make consists of transforming products into a finished state in order to meet planned or actual demand, deliver aims to describe the activities associated with the creation, maintenance, and fulfilment of customer orders, return process describes all activities related to the reverse flow of goods, and enable process describes all activities related to supply chain management including the management of business rules, performance management, and data management, among others. In this paper, we focus our attention on the delivery management process, more precisely on the cement distribution operations regarding the problem of the percentage deviation of weighing during the dispatch workflow process. The dispatch workflow process is defined here as the logistic flows of material (cement) in the cement industry and it is considered a crucial process in such an industry value chain. The problem of percentage deviations in the weighing processes out of thresholds stipulated for security purposes compromises the entire dispatch workflow process, leading to monetary, logistic, and security consequences, including long delays due to blockage of transportation vehicles inside the factory, and monetary and service level losses. In this way, certain improvements in the cement distribution operations can lead to cost reduction, and improvement in service levels and consequently achieve SC performance.

TABLE 1. Chronological scientific works regarding heavy vehicle weighing processes using ML techniques.

Study	T ^a	Algorithm ^b	Evaluation measures ^c	D ^d	E ^e	P ^f	R ^g	TD ^h
[44]	ML	ANN	ACC	Yes	HO	-	-	T
[45]	ML	SVM	ACC, Micro-TPR, Micro-PPV, Macro-TPR, Macro-PPV	Yes	CV	-	16k	T
[46]	ML	CNN, TL	ACC	Yes	HO	-	2.4k	I
[47]	ML, S	ANN, MC	MSE	No	CV	-	500k	T
[49]	ML	GNB, KNN, SVM, DT	ACC, F1, TPR, PPV	Yes	CV	15w	33.4k	T
[50]	ML	PCA, KNN	RMSE	Yes	-	1y	564	T
[51]	ML	CNN	ACC, RMSE	No	HO	-	8.8k	I
[52]	ML	CNN	ACC, TPR, PPV	No	HO	-	212	I
[53]	ML	TL	ACC	No	HO	-	4.8k	I
[48]	ML	RF	TPR, FPR, PPV, ROC, F1	Yes	HO	-	2k	T
This work	ML	SVM, DT, RF, XGBT, GBT, MLP	AUC, TPR, FPR	Yes	RW	2.4y	45k	T

^a Modeling Technique: ML - machine learning, S - simulation.

^b ANN - artificial neural network, SVM - support vector machine, CNN - convolutional neural network, RF - random forest, KNN - k-nearest neighbours, MLP - multi-layer perceptron MARS - multivariate adaptive regression, GBT - gradient-boosted tree, XGBT - extreme gradient boosting, DT - decision tree, PCA - principal component analysis, GNB - gaussian naïve bayes, TL - transfer learning, MC - Monte Carlo.

^c RMSE - root Mean square error, MSE - mean squared error, TPR - true positive rate, PPV - positive predictive value, ACC - accuracy, F1 - f1 score, FPR - false positive rate, ROC - receiver operating characteristic, AUC - area under the ROC curve.

^d Empirical Data.

^e Model Evaluation scheme: HO - hold-out, CV - cross-validation, RW - rolling window.

^f Data Period: w - week, m - month, y - years.

^g Data Records: k - thousands of records.

^h Type of Data: T - tabular, I - images.

**FIGURE 2.** SCOR model extracted from [3].

Let $\mathcal{T} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ be the training set with $\mathbf{x}_i \in \mathbb{R}^p$ the vector of predictor variables and $y_i \in \{0, 1\}$ the response variable of each individual i . The main goal of this work is to construct a classification model that predicts the normal or abnormal percentage deviation in weighings during the dispatch workflow process (response variable y) by taking advantage of the predictor variables x .

IV. MATERIAL AND METHODS

A. INDUSTRIAL CASE STUDY

This work was developed in the Engineering and Innovation Department of Cachapuz - Weighing & Logistics Systems, Lda, Portugal. This company is a leader in Portugal in the design and manufacture of weighing equipment and a Euro-

pean reference in the design and implementation of Software solutions to automate the logistics, dispatchings, and weighings processes in various industrial sectors (for instance, the SLV Platform composed by SmartWeigh Solutions and SLV Solutions). Cachapuz's integrated weighing systems are part of the operation of weighing logistics processes in 63 countries from 5 continents. Regarding the cement sector, Cachapuz designed and developed a framework called SLV Cement¹ (an integral part of the SLV Solutions²) that focuses only on logistics challenges faced by cement companies. It represents dispatching and logistics flow control solutions that aim to automate all processes from the truck's arrival at the cement plants to the cement expeditions.

In our case study, the dispatch workflow process consists of the logistic flows of material (cement) in the cement industry, which contains several areas responsible for a specific task, as depicted in Fig. 3. This process includes all operations of entry, exit, and transfer of material (e.g., parking management, weighing operations, dispatching, entry/exit gates, bulk/bagged/pelletized cement loading), as illustrated in Fig. 4. Hence, cement is dispatched from the cement industry plant directly to customers.

¹<https://www.slvcement.com/>

²<https://www.cachapuz.com/slv-solutions/>

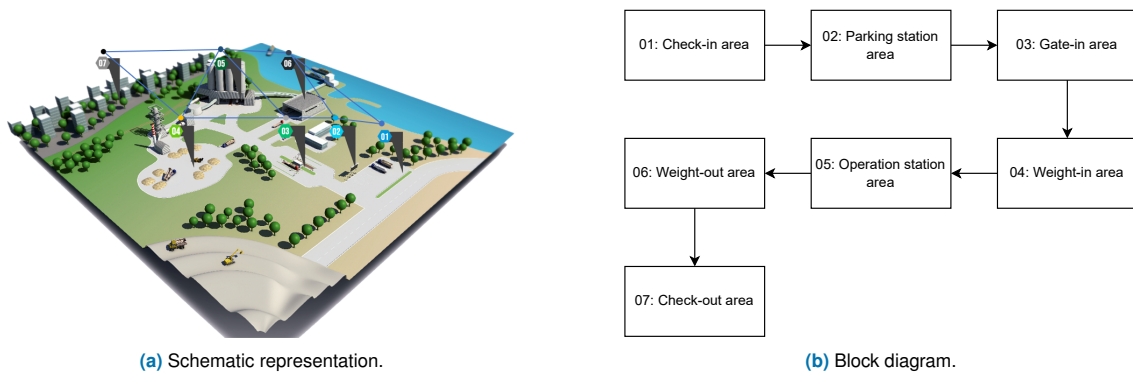


FIGURE 3. Material flow in a cement plant.

Firstly, the client (represented by the driver of the transportation vehicle) performs the check-in (area **01**) and heads to a parking station, waiting for entry to the factory (area **02**). Next, when authorized to enter, the client proceeds to the gate-in area (area **03**) and afterwards to the weight-in area in order to weigh the vehicle tare, i.e., vehicle empty (area **04**). At this point, the client is routed to an operating station composed of three different stations: bulk station (area **05.1**), automatic station and manual station (area **05.2**). At the bulk station, the transportation vehicles follow the cement silos for loading. Regarding the automatic station, the transportation vehicles go to the warehouse for loading using bagged or pelleted cement, and the manual station aims for loading using bagged cement. After loading with the ordered quantity, the weighing process is performed again (weight-out) in order to get the weight of the vehicle fully (area **06**). Finally, at the check-out area (area **07**) is verified the percentage deviations in weighings (difference between weight-in and weight-out). At this stage, the client proceeds to the gate-out and leaves the cement plant when the percentage deviation is between the defined threshold, otherwise is performed the inspection of the transportation vehicle. This process enables the transportation vehicle drivers for the loading operations entirely in self-service mode. In fact, all steps of this process are carried out for control and security purposes, ensuring that the customer is carrying the amount previously agreed and also that the safety weight to be carried by the transport vehicle is not exceeded.

The aforementioned process is considered crucial in the cement industry value chain. However, cement industry companies are constantly facing the problem of percentage deviations in the weighing processes (weigh-in and weigh-out) out of thresholds stipulated for security purposes (in our case study $]-\infty; -2] \cup [2; +\infty[$) compromising the entire dispatch workflow process and consequently leading to monetary loss, and logistic and security consequences. The expressed problem leads to several consequences, although, the most common is the blockage of the transportation vehicle inside

the factory for further inspection when performing the check-out, which in several circumstances takes a lot of time to be solved and consequently huge constraints during this process caused by long delays. As an aggravating factor, such blockage of the transportation vehicle inside of the factory may force to restart part of the process, i.e., unload the cement and redirect the transportation vehicle to the previously assigned operation station (area **05.1** or **05.2**) to repeat this stage of the process.

B. TECHNOLOGICAL ARCHITECTURE

The proposed data-driven technological architecture is composed of three main components, as depicted in Fig. 5.

- 1) **Machine Learning:** this component consists of deploying the selected ML model using a Docker container. We have developed a REST API using the Flask micro-framework, providing two main endpoints: an endpoint that allows training the selected ML and then creating a pickle³ of this model, and a second endpoint that aims to load the pickled model to afterwards be used to make predictions. In addition, all prediction results are saved in a structured query language (SQL) database.
- 2) **SLV Cement:** this platform is part of the SLV Platform and is proprietary of the Cachapuz - Weighing & Logistics Systems, Lda. It is responsible for automating the material flow in cement plants and aims to improve and optimize the dispatch procedures, allowing the monitoring and control of operations so that to lead to logistics and service level performance. In addition, this platform is integrated with SAP enterprise resource planning (ERP). All data used by the ML models are ingested from the SQL databases of this platform.
- 3) **Visualization:** this component allows the creation of data visualizations for the end-users using the PowerBI⁴ tool.

³<https://docs.python.org/3/library/pickle.html>

⁴<https://powerbi.microsoft.com/pt-pt/desktop/>

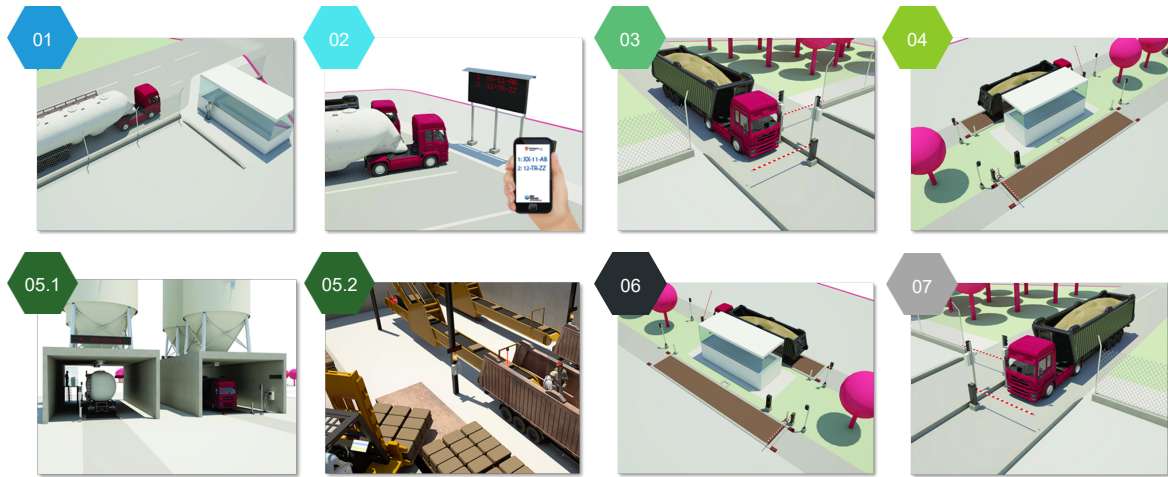


FIGURE 4. Steps of the analyzed dispatch workflow process.

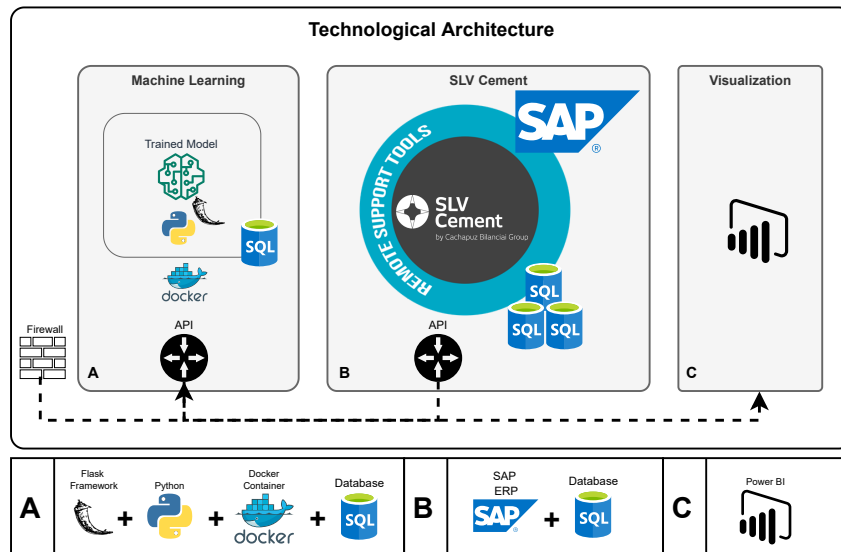


FIGURE 5. Technological architecture of the solution.

Figure 6 illustrates the exploratory diagram of the method applied for the prediction of weighing deviations in the dispatch workflow process.

C. MACHINE LEARNING PIPELINE

Fig. 7 illustrated the proposed ML pipeline composed of the following three main steps:

- 1) **Data Preparation:** this step comprises data process tasks such as data selection, data cleansing, data construction and formatting, which are more detailed in Section IV-D.
- 2) **Modeling and Evaluation:** several ML models are implemented and then evaluated its predictive power adopting evaluation metrics, as detailed in Sections IV-H.

- 3) **Deployment:** the selected ML model is deployed from a REST API developed using the Flask⁵ framework, running in a Docker⁶ container. During this process, an ML model is selected and trained, and subsequently, a pickle of the trained model is created to perform future predictions. Additionally, all predictions are stored in a SQL Database.

D. DATA PREPARATION

- 1) Weighing process data

The data used in this work are proprietary of the Cachapuz - Weighing & Logistics Systems, Lda, Portugal, collected from the SQL databases of the SLV Cement platform between December 2019 and March 2022, containing all the dispatch-

⁵<https://flask.palletsprojects.com/>

⁶<https://www.docker.com/>

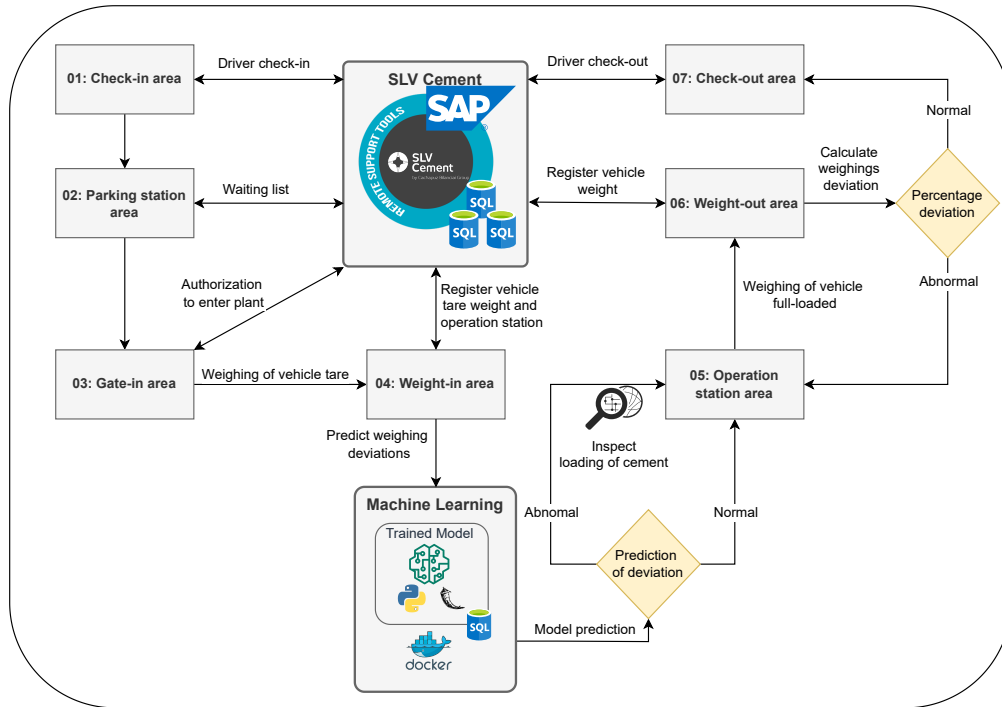


FIGURE 6. Exploratory diagram (dispatch workflow process and prediction of weighing deviations).

ing workflow process (all steps from the truck’s arrival at the cement plant to the cement expeditions). The collected data include a total of 45,000 examples with 22 features, as described in Table 2. These data were used as a basis for the feature engineering process, where new variables were generated using the business domain expert knowledge in order to improve the predictive power of the ML models (see Section IV-D2).

We performed the exploratory data analysis (EDA), and we found that the class distribution of our target output is imbalanced [18], i.e., the class distribution is biased or skewed as depicted in Fig. 8. We found that 91.2% of dispatch workflow occurs normally without abnormal weighing deviations (class “0”), and only 8.8% represents an alarmist situation, where trucks are blocked for further inspections due to such deviations in the weighing process (class “1”).

It is common sense that imbalanced classification represents a challenge for predictive modeling since severe class imbalance between majority and minority classes can bias the predictive performance of ML algorithms, especially regarding the minority class. Furthermore, in this context, the minority class is usually most interested and therefore the problem is more sensitive to misclassification for the minority class than the majority one. We address this problem by adopting the synthetic minority over-sampling technique (SMOTE) [19] in the training set in order to generate new artificial instances of minority class to balance the class distribution. It should be noted that the test sets are analyzed using their original class distribution values, thus are highly unbalanced. In addition, we also performed data cleansing

tasks in order to remove duplicated data and fill rows with missing values. We assess the relationship between variables through the correlation matrix, as illustrated in Fig. 9. Note that, there is a high correlation (correlation coefficient whose magnitude is greater than 0.50) between some of the dataset variables. Furthermore, after the feature engineering task, using the variable inflation factor (VIF), we performed the multicollinearity calculation between variables and gradually removed the variables with high multicollinearity until the level of the remaining variables’ multicollinearity drops significantly.

2) Feature Engineering

Feature engineering consists of creating new features from original ones by using domain knowledge in order to improve the prediction capability of ML models [30], [37]. Hence, we created a set of new features based on both the interpretation of data after the exploratory data analysis (EDA), and multiple brainstorming meetings with the domain expert from Cachapuz - Weighing & Logistics Systems, Lda - Portugal, as follows:

- **Historical percentage of vehicle blockages** - the percentage of vehicle blockages may influence future blockages, i.e., if a given transportation vehicle is frequently blocked inside the factory for further inspection, there is a high probability that it happens again.
- **Average of percentage deviation at operating stations weekly** - a weekly average of deviations regarding a given operating station.

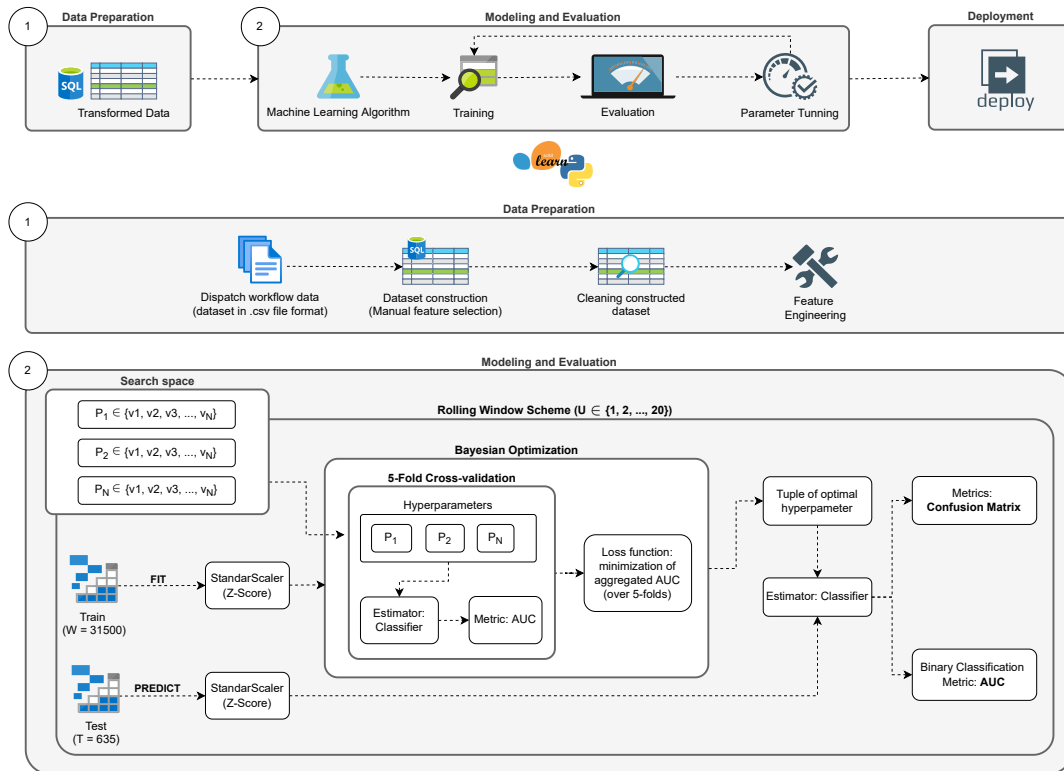


FIGURE 7. Machine Learning pipeline.

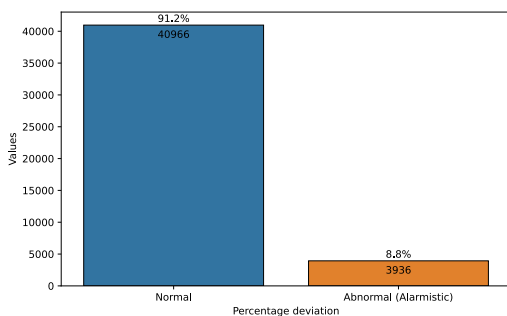


FIGURE 8. Distribution of output target.

- **Average of percentage deviation at operating stations hourly** - an hourly average of deviations regarding a given operating station.
- **Average of n deviation at operating stations** - the average of deviations regarding the last five weighings from a given operating station.
- **Hour of weighing (*)** - Hour of the weigh-in operation. There are certain hours of the day when alarmistic deviations occur frequently.
- **Day of weighing (*)** - Day of the weigh-in operation. The day of operation may influence future alarmistic deviations.

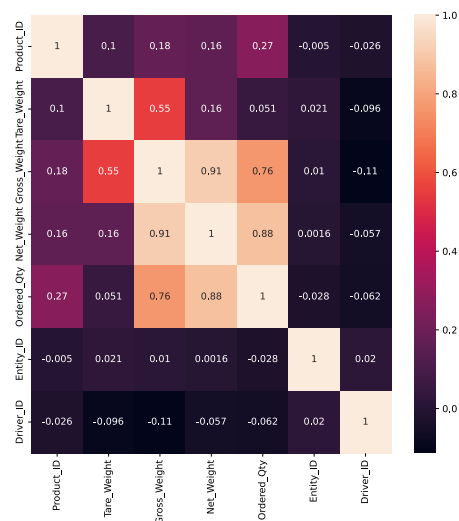


FIGURE 9. Correlation between variables.

- **Day of week of weighing (*)** - Day of the week related to the weigh-in operation. The day of the week regarding the operation may influence future alarmistic deviations.

TABLE 2. Summary of the weighing process data attributes.

Context	Attribute	Description (format, example)	Feature
Vehicle	Documentation	Type of documentation (text, '20190320')	No
	Type of vehicle	Configuration of truck (Axles) (text, 'Z002')	No
	Licence plate	Licence place of the truck (text, '01-MH-RD')	No
Product	Product number	Product ID (text, 'FH123201')	No
	Description	Description of the product (text, 'Cement bag')	No
Order	Quantity	Ordered quantity (number, 3000)	Yes
Weighing	Tare	Weight of truck tare (number, 4500)	Yes
	Gross Weight	Gross weight (number, 8000)	No
	Net weight	Net weight (number, 3500)	No
Operation	Station	Operating station (text, 'Station 1')	No
	Creation date	Date of record creation (date, '20211212')	No
	Entrance date	Date of truck entrance in the factory (date, '20211212')	No
	Tare Date	Date of weighing truck tare (date, '20211212')	No
	Start Operation date	Date of operation start (date, '20210202')	No
	End Operation date	Date of operation end (date, '20211212')	No
	Gross Date	Date of weighing truck fully (date, '20211212')	No
	State	State of the operation (text, 'C')	No
Exit date	Date of a truck exit of factory (date, '20211212')	No	
Entity & Driver	Entity number	Entity ID (text, 'XD-125-D')	No
	Driver number	Driver ID (text, 'SS02RD6')	No
	Driver name	Name of the driver (text, 'John Phillips')	No
Target	Deviation	Deviation from weighing process (ordered quantity vs. net weight) (number, 1)	Yes

- **Month of weighing (*)** - Month of the weigh-in operation. The month of operation may influence future alarmistic deviations.
- **Inspection period during a day** - Period of the day with an inspector for manual loading of cement to the transportation vehicles.

(*) After EDA from the data, we find patterns associated with these features.

E. FEATURE PREPARATION AND HYPERPARAMETER TUNING

1) Feature scaling

Feature scaling is a crucial data pre-processing task for good classification performance. It aims to scale or transform data in order to make an equal contribution to each feature. Certain ML algorithms such as k-nearest neighbors (k-NN), artificial neural networks (ANN), and support vector machines (SVM) are sensitive towards feature scaling. Among several feature scaling methods, Min-Max [0, 1] and Z-score methods were reported to be the best ones [2]. In this work, we adopted the Z-score method to be the most sensitive to outliers. Thus, the mean and standard deviation are calculated from the training set (oldest data) to then be used to re-scale the data (train and test sets) by using (1) in order to features have zero mean and unit variance.

$$x'_{i,n} = \frac{x_{i,n} - \mu_i}{\sigma_i} \quad (1)$$

2) Feature selection

Feature selection aims to select a subset of features by removing all unnecessary, irrelevant, and redundant data that may negatively affect the performance of ML models [29]. According to [29], two main approaches can be followed or a combination of both to accomplish this task, namely manual and automatic selection. In addition, the authors highlighted the importance of manual selection, but also the usefulness of automatic selection. In this work, we only performed a manual feature selection using business domain knowledge, since there was no need to resort to a further automatic feature selection process given that the manual selection resulted in a rather small number of features.

3) Hyperparameter tuning

Hyperparameter tuning represents a crucial task in every ML project and aims to define the optimal hyperparameters for a given ML model. In this work, we adopted Bayesian optimization using the HyperOpt library combined with k-fold cross-validation for hyperparameters optimization using only the training set. First, we define the search space and then, in the objective function, we consider the symmetrical

AUC metric ($-1 \times AUC$) as the criterion to be minimized over 10 iterations of the tree of parzen estimations (TPE) method [22].

F. MODELING: CLASSIFICATION ALGORITHMS

We tested six ML classification algorithms: decision tree, random forest, gradient-boosted tree, extreme gradient-boosting tree, support vector machine and multilayer perceptron.

1) Decision tree (DT)

A decision tree (DT) represents a popular method used for classification and regression tasks due to its ease of interpretation. It consists of a tree structure scheme (representing a set of rules) composed of a collection of decision nodes (input variables) connected by branches extending from root nodes to leaf nodes (decision outcomes or targets). Such a tree structure scheme express a general pattern of recursive partition/split, in which, starting at the root node, attributes are tested on decision nodes, resulting in multiple branches (which can be visualized as a set of IF-THEN statements) and similarly, each branch can lead to another decision node or a leaf node [26].

2) Random forest (RF)

Random forest (RF) is an efficiently and commonly used ML algorithm for classification and regression tasks, formulated by [23]. It consists of an ensemble of decision trees built by combining the bagging technique (also known as bootstrap aggregation) with random feature selection regarding the reduction of overfitting risk and achieving better predictive power [14]. The bagging technique consists of a general aggregation scheme, which aims to generate bootstrap samples from the original dataset using the CART method and decrease Gini impurity (DGI) as the splitting criterion [12], [13]. When building each tree, at each split, only a subset of features randomly selected (*mtry*) are considered candidates for splitting. In addition, the split is performed in order to maximise the CART criterion [13]. For the classification task, the final prediction of the ensemble is determined by majority voting [5].

3) Gradient-boosted tree (GBT)

Boosting consists of a technique formulated in [11], which aims to improve the accuracy of a predictive function by converting weak learners into strong learners in an iterative way. It applies weak learners in a sequential fashion to repeatedly re-weighted versions of the training data [1], [25]. The weight for the incorrectly classified examples is increased after each boosting iteration, whereas the weight for correctly classified ones is decreased [25]. Gradient boosting consists of building additive regression models by fitting in a sequential fashion a parameterized function (base learner) to pseudo-residuals (gradient of the loss function $L(y_i, F(x_i))$ being minimized) by least squares at each iteration [10]. Randomization can be introduced into the iterative procedure in order to improve

the approximation accuracy and execution speed of gradient boosting, improving the robustness against overfitting of the base learner (tree). [6] proposed the gradient boosting machine (GBM) using the connection between boosting and optimization [10].

4) Extreme gradient boosting (XGBoost)

The extreme gradient boosting (XGBoost) algorithm consists of an ensemble of decision trees based on gradient boosting proposed by [4] in order to be efficient and highly scalable [5], [14]. It introduces the regularization term into the objective function in order to prevent the overfitting phenomenon [4], [5]. Thus, the XGBoost build an additive expansion of the objective function through the minimization of the loss function, similarly to gradient boosting, which can be expressed as follows [5]:

$$L_{xgb} = \sum_{i=1}^N L(y_i, F(x_i)) + \sum_{m=1}^M \Omega(h_m) \quad (2)$$

where $L(\cdot)$ is a convex loss function and $\Omega(\cdot)$ represents the regularization function used to prevent overfitting by controlling the complexity of the model.

$$\Omega(h) = \gamma T + \frac{1}{2} \lambda \|w\|^2, \quad (3)$$

where T denotes the number of leaves nodes of the tree, w is the output scores of the leaves, and γ and λ represent the regularization parameters that determine the relative weight of the penalty term.

5) Support vector machine (SVM)

Support vector machine (SVM) is a powerful learning and popular algorithm for classification and regression purposes developed by [8], based on statistical learning theory [7]. It uses the nonlinear mapping (ϕ) to transform the input \mathbf{x} for a high m -dimensional space ($m > I$, where m represents the number of features) with kernel function ($K(x, x') = \sum_{i=1}^m \phi_i(x)\phi_i(x')$) to afterwards find the best linear separating hyperplane (see (5)) regarding a set of support vector points in the feature space [7]. Among several kernel functions, e.g. linear, polynomial or sigmoid, the most popular is the Gaussian kernel (4):

$$K(x, x') = \exp(-\gamma \|x - x'\|^2), \gamma > 0 \quad (4)$$

$$\hat{y}_k = w_0 + \sum_{i=1}^m w_i \phi_i(x_k) \quad (5)$$

The output target of the binary classification is given in the range $y \in [-1, 1]$ and the classification function:

$$f(x_k) = \sum_{j=1}^m y_j \alpha_j K(x_j, x_k) + b \quad (6)$$

were b and α_j represent the model coefficients.

6) multilayer perceptron (MLP)

Multilayer perceptron (MLP) is one of the most popular neural networks for pattern classification tasks. It consists of a feed-forward neural network composed of three different layers, which include an input layer, one or more hidden layers and an output layer fully interconnected forming the network topology, trained using the back-propagation algorithm [25]. Initially, each connection represents a random weight, then it is adjusted during the learning process. The input layer is intended to receive inputs from the outside world, as such the number of neurons in this layer is defined by the input data. The output layer provides the result of the predictions to the outside world, and therefore, the number of neurons is determined by the number of classes. The hidden layer aims to link the input layer to the output layer, extracting useful features and subfeatures from the input patterns with respect to the prediction output. Thus, the number of hidden layers and neurons in each hidden layer are both user-defined considering the problem under consideration [9], [26], [28]. Moreover, the hidden neurons use the sigmoid transfer function:

$$f(zi) = \frac{1}{(1 + \exp(-zi))} \quad (7)$$

$$zi = \sum_{j=1}^n w_j x_j \quad (8)$$

where x_j represent the activity of the j th input neurons and $f(zi)$ the activity of the i th neurons of hidden layer.

G. EXPLAINABLE ARTIFICIAL INTELLIGENCE

Over the last few years, artificial intelligence (AI) has increasingly been considered a key driver of value creation for companies. However, even with these unprecedented advances, several AI-based systems lack transparency due to their "black-box" nature [39], [54]–[56]. Indeed, black-box ML methods, such as SVM, ANN, DL, RF, and XGBoost, among others, are increasingly being used for addressing problems related to different areas of activity, providing powerful and accurate predictions [39], [41]. As such, these methods are very complex and consequently not directly explained or easily understood by humans. Nevertheless, in general, humans are sceptical about adopting techniques that are not directly interpretable, tractable and trustworthy, even worse for making important decisions [39], [41], [56]. Yet, explainable artificial intelligence (XAI) proposes to make AI more transparent [39], [55], [56]. In this work, we adopted the Shapley additive explanations (SHAP)⁷ for providing interpretability for the proposed ML models. This XAI approach was proposed by Lloyd Shapley and uses Shapley Values from the games theory to interpret the output of ML models [57]. Indeed, the Shapley value of a feature represents the difference between the average prediction value of samples considering and not considering this feature [14].

⁷<https://shap.readthedocs.io/>

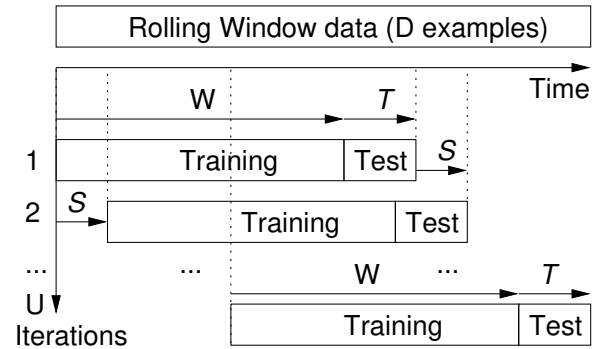


FIGURE 10. Rolling window scheme adapted from [36].

H. EVALUATION

We adopted a realistic and robust rolling window (RW) scheme for evaluating the classification models, as depicted in Fig. 10. This scheme is realistic because it produces a set of training and test iterations over time, simulating the real environment in which the ML model will be used. Furthermore, in contrast to the popular single hold-out train and test scheme, it is robust because it allows a set of predictions to be produced regarding each iteration, thus allowing multiple evaluations of the ML model over time.

This scheme works in a systematic way, as following described. Firstly, the model is trained using a fixed training window W which contains the oldest samples and afterwards makes predictions using the subsequent T samples from the first iteration of RW ($U = 1$). Then, the training window slides in S instances in the second RW iteration ($U = 2$), causing the replacement of the S oldest samples of the training window by the S recent ones. Therefore, a new model is fitted and then predicts the new subsequent T samples. As aforementioned, this process is systematic and repeats until the last RW interaction. The total number of RW iterations is calculated by using (9).

$$U = \frac{(D - (W + T))}{S} \quad (9)$$

1) Measuring model performance

The overall predictive performance of classification models is given by the AUC of receiver operating characteristic (ROC) analysis, also known as AUC or AUC-ROC [24]. The ROC analysis is obtained by considering the predictions as probabilities (p) for a binary class. The class is assumed true if $p > D$, where D is a decision threshold. The predicted class labels can be used to compute the confusion matrix regarding a fixed D . Fig. 11 depicts an example of the confusion matrix, which matches predicted outcomes with the actual values. It includes four main statistics for the binary classification task [28]:

- 1) True positives (TP) - number of positive class correctly classified;

- 2) True negatives (**TN**) - number of negative class correctly classified;
- 3) False positives (**FP**) - number of negative class incorrectly classified as positive class, and;
- 4) False negatives (**FN**) - number of positive class incorrectly classified as negative class.

		Prediction outcome	
		N	P
Actual value	N	TN	FP
	P	FN	TP

N - Negative; P - Positive

FIGURE 11. Confusion matrix for a binary classification task.

Moreover, we also considered other classification metrics which are obtained from the confusion matrix, as follows described [21], [26]:

- 1) True positive rate (TPR), recall, hit rate or sensitivity:

$$TPR = \frac{TP}{TP + TN} \quad (10)$$

- 2) False positive rate (FPR) or fall-out:

$$FPR = \frac{FP}{FP + TN} \quad (11)$$

The ROC curve is a two-dimensional graphical representation technique for visualizing, organizing, and selecting classifiers based on their performance. It aimed to summarize the trade-off between the TPR (y-axis) and FPR (x-axis) for several threshold points (D), which ranges from 0.0 to 1.0 [21], [24], [27], [28]. Moreover, the ROC curve is commonly used to measure the overall prediction accuracy of the model [27]. The AUC aims to measure the quality of the probabilistic classifier and is calculated by using (12). A random classifier has an AUC of 0.5 and a perfect classifier has an AUC of 1 [27].

$$\begin{aligned} AUC &= \int_0^1 \frac{TP}{TP + FN} d \frac{FP}{FP + TN} \\ &= \int_0^1 \frac{TP}{P} d \left(\frac{FP}{N} \right) \end{aligned} \quad (12)$$

V. EXPERIMENTS AND RESULTS

All experiments in this work were conducted using the scikit-learn library⁸ and code written in Python programming language, executed on a personal computer (Lenovo E590)

⁸<https://scikit-learn.org/>

with Intel Core i7-8565U processor, CPU 1.80-2.00GHz and 16GB RAM installed.

A. EXPERIMENTAL SETUP

In this study, we explore six ML classification models (RF, GBT, XGBoost, SVM, MLP, and DT) by adopting the rolling window (RW) scheme to evaluate these models, producing several training and test iterations and thus simulating a real environment. After consulting logistics domain experts, we configured the RW scheme considering the values $W = 31500$, $T = 635$ and $S = 635$ (thus, the test sets do not overlap), producing a total of $U = 20$ iterations. For each iteration of the RW, both train and test sets are scaled using the Z-score technique. Thus, we use the first iteration ($U = 1$) to calculate the mean and standard deviation on the features of the training set and then infer them to the features of the training and test sets of all iterations. Afterwards, we use only the training set for hyperparameter tuning adopting Bayesian optimization. For each tested model, we defined the hyperparameter space and the objective function to be minimized over 10 iterations (maximum number of evaluations) of the tree of parzen estimators (TPE) method [22]. Yet, in the objective function, we employed 5-fold cross-validation [59] to further calculate the AUC metric (loss or the criterion to be minimized). Indeed, the AUC metric provides several advantages, such as the quality values are not affected when the classification data is unbalanced and it is easily interpreted by humans (50% – random classifier, 60% – reasonable, 70% – good, 80% – very good, 90% – excellent, and 100% – perfect) [58].

Thus, for the DT model, we defined the hyperparameter search space under the ranges $max_depth \in \{2, 5, 10, 20, 30\}$ and $min_samples_split \in \{2, 6, 10\}$. The RF and GBT models considered the same hyperparameter space of the DT model, setting additionally the number of trees to train $n_estimators = 200$. Regarding the SVM model, we defined the search space $C \in \{0.01, 0.1, 0.5, 1.0, 2.0\}$ and $kernel \in \{“linear”, “poly”, “rbf”, “sigmoid”\}$. In the case of MLP, we defined a network with one hidden layer with H neurons determined using the heuristic $H = round(N/2)$ where N denotes the number of inputs and also trained with 500 epochs ($max_iter = 500$). We additionally considered $alpha \in \{0.0001, 0.05\}$, $solver = “sgd”$ and $learning_rate \in \{“constant”, “invscaling”, “adaptive”\}$. Lastly, for the XGBoost model, we defined the ranges $eta \in \{0.0, 0.25, 0.5, 0.75, 1.0\}$, $max_depth \in \{2, 5, 10, 20, 30\}$ and $max_bin \in \{10, 20, 40, 80, 100\}$.

Furthermore, we adopted the non-parametric Wilcoxon signed-rank test for paired samples [40] to calculate a pseudo-median of the AUC metric for each model explored over twenty RW iterations ($U = 20$) and determine whether are significantly different from one model to another.

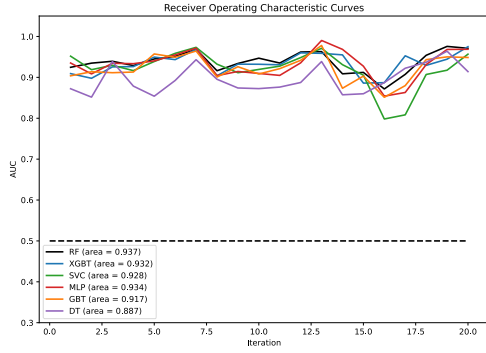


FIGURE 12. Evolution of AUC score over the RW iterations.

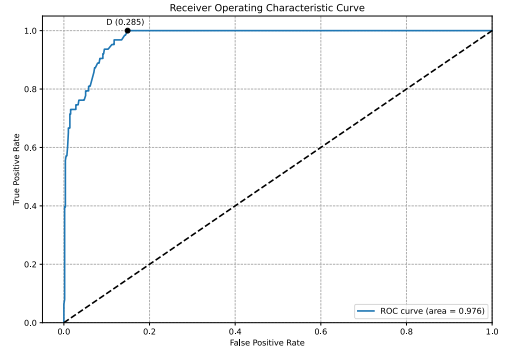


FIGURE 13. RF ROC curve at iteration $U = 19$ and selected decision threshold $D = 0.285$.

B. OVERALL PREDICTIVE PERFORMANCE

Table 3 presents the overall predictive power of tested ML models in terms of median AUC regarding 20 iterations of the RW scheme. The last two columns denote the median computational effort, in terms of the training and prediction time. The results show that the RF model provides the best predictive performance with a median AUC of 0.937, in contrast with the DT model which provides the smallest median AUC, although, it needs less time to train and predict. In addition, we applied the Wilcoxon signed-rank test [40] for the AUC significance, which indicated that the RF model is statistically significant regarding the GBT, SVM, and DT classification models.

Fig. 12 illustrates the evaluation of the AUC metric over the RW iterations $U \in \{1, 2, \dots, 20\}$. Furthermore, to demonstrate the quality of the obtained results and better measure the impact of the selected RF model, we selected the test set of iteration $U = 19$ to find and fix the best threshold (Th) D (see Fig. 13) and then extract the confusion matrix at iteration $U = 20$ using the fixed threshold ($D = 0.285$), as illustrated in Fig. 14. Table 4 describes the RF model prediction for the selected threshold D in terms of TP, TN, FP, and FN which allows calculating the TPR and the FPR metrics.

Fig. 15 illustrates the ranking of the top 11 most important features and their influence regarding the output classes for the selected RF model using SHAP for the RW iteration $U = 20$. Note that, the feature *Average_Station_Hourly* represents the most important when predicting percentage deviation in the weighing process, followed by the feature *Percentage_Blocks*. Moreover, eight of these most important features were created in the feature engineering process (see Section IV-D2).

Fig. 16 and 17 deepen the feature importance analysis, helping to understand the probabilities of features in terms of the class "0" (i.e., normal or acceptable percentage deviation). Thereby, both figures exemplify the influence of several attributes in the model's decision-making. For that, we applied SHAP to visualize the Force plot in two different weighing processes: in the first case, the RF model

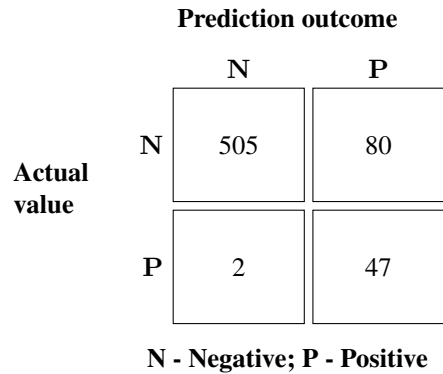


FIGURE 14. Confusion matrix of the RF model at RW iteration $U = 20$ when using $D = 0.285$.

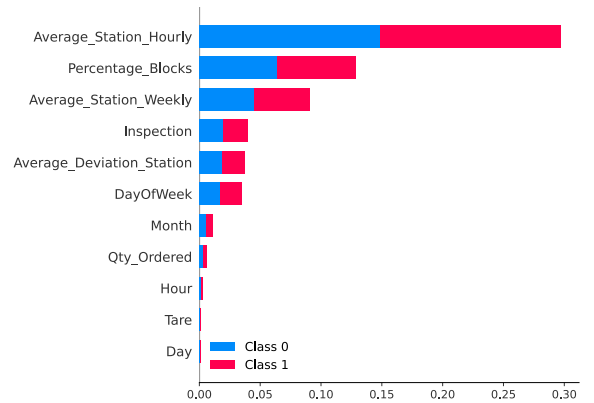


FIGURE 15. Feature importance for the RF model.

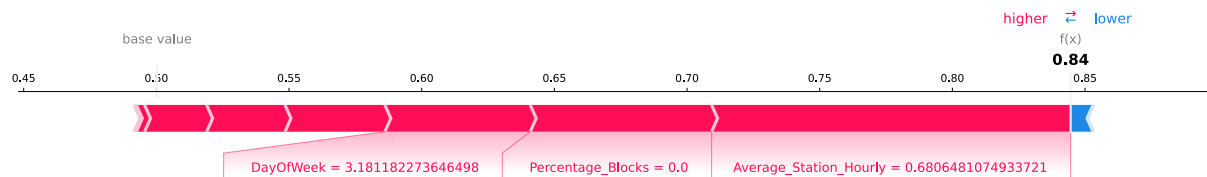
provides the result of prediction as a class "0", and for the second case, as a class "1". Fig. 16 illustrates a Force plot of a normal weighing and determines that the features *Average_Station_Hourly* and *Percentage_Blocks* contributed the most to the prediction of class "0", with a total probability of 0.84.

Fig. 17 illustrates the behaviour of the features in a sce-

TABLE 3. Comparison of the optimized ML models (best values in **bold**).

Models	AUC	Training time (s)	Predict time (s)
random forest (RF)	0.937*	14.71	0.031
gradient-boosted Tree (GBT)	0.917	177.71	0.015
extreme gradient-boosting (XGBoost)	0.932	10.89	0.001
support vector machine (SVM)	0.928	318.48	0.374
multilayer perceptron (MLP)	0.933	15.19	0.000
decision tree (DT)	0.887	0.18	0.000

* Statistically significant under paired comparison with GBT, SVM and DT.

**FIGURE 16.** Force plot: Impact of features on normal weighing (percentage deviations within the thresholds).**TABLE 4.** Prediction results of the RF model at RW iteration $U = 20$ when using $D = 0.285$.

Th (D)	TP	TN	FP	FN	TPR	FPR
0.285	47	505	80	2	95.91%	4.08%

nario where the result of a prediction was class “1” and shows how they impacted the probability of class “0”. Thus, we can note that features *Average_Station_Hourly* and *Average_Station_Weekly* contributed the most to decreasing the probability of class “0” and, on the other hand, feature *Percentage_Block* contribute the most to increasing such probability.

Fig. 18 depicts a dashboard implemented using the PowerBI tool in order to create visualizations for the end-users and therefore facilitate the integration of the proposed in the case-study company.

The left side of the developed dashboard shows information regarding the dispatching workflow process, namely the product ID associated with the process ID, dispatching date and the probabilities of class “0” and “1” from the RF model, providing useful information regarding the results of predictions. Moreover, it shows more detailed information regarding the operation, product and ordered quantity, as well information regarding the ranking of top more important features such as average deviation (average deviation at the station and average deviation at the station over time, hourly, daily and weekly), helping to explain the model prediction results.

VI. CONCLUSIONS

Industry 4.0 is significantly impacting the supply chain of organizations by helping to support and improve complicated

and dynamic processes as well as manage large-scale production and customer integration in order to achieve competitive advantage and improved organizational performance. In today’s globally competitive market, managing costs, manufacturing, and product deliveries represent key drivers for competitive advantage. In this context, we focus on improving the order fulfilment process through the improvement of the dispatch workflow process, more specifically in the process of loading cement into the transportation vehicle. Hence, the occurrence of anomalies (deviations) in such a process represents a complex problem that directly affects cement industry organizations in terms of security, service level, and monetary losses. Indeed, in this work, we propose an ML approach to predict weighing deviations in the dispatch workflow or vehicle dispatch process. Moreover, we proposed an ML pipeline for exploring six ML classification models and afterwards the selected model is deployed using the technological architecture (see Fig 5). Regarding the data preparation step of the proposed pipeline, we first extracted the data from the databases of the SLV Cement platform related to the aforementioned process. The data extraction activity resulted in a total of 45,000 records which were stored in CSV file format. Then, we perform data selection in order to select relevant features and also the cleansing tasks to remove inaccurate and incorrect records. Finally, in the features engineering task, we created new features based on the original ones using the domain expert knowledge and also the knowledge acquired from the data after performing the exploratory data analysis (EDA).

During the modeling stage, we adopted the RW scheme to evaluate six ML classification models: decision tree, random forest, gradient-boosted tree, extreme boosting tree, support vector machine, and multilayer perceptron. All the features

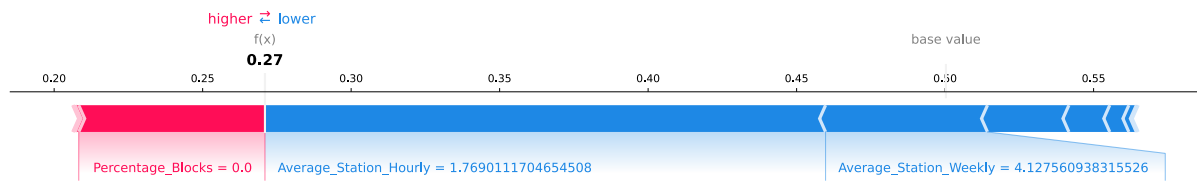


FIGURE 17. Force plot: Impact of features on abnormal or alarmistic weighing (percentage deviation without the thresholds).

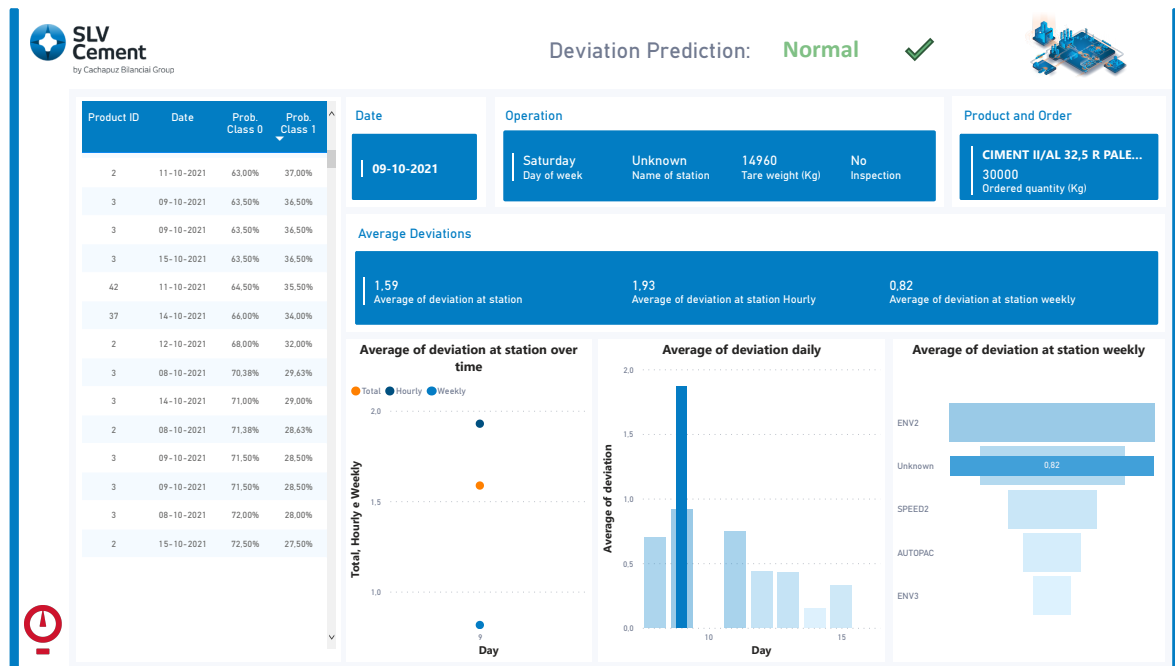


FIGURE 18. Dashboard: Predicting deviations of weighings in workflow dispatch process.

selected to compound the models' input sets are standardized using the z-score technique. We also adopted AUC as the evaluation metric to compare the predictive power of the explored models. Regarding the results, we find that the RF model provides better predictive power with a median AUC of 0.937 over the twenty iterations of the RW, followed by the GBT, XGBT, SVM, MLP, and DT models. However, in terms of computational effort, the DT model requires the least amount of training time, approximately 0.18 seconds, whereas the RF model requires 14.71 seconds to train. The SVM model requires the most time to train (approximately 318.48 seconds). Furthermore, after applying the Wilcoxon signed-rank test, we found that the RF model is statistically significant over GBT, SVM, and DT. The RF model is selected as it stands out as the best classifier in terms of prediction power. To better evaluate the performance of the selected model in a realistic setting, we set the iteration $U = 19$ to search for the best threshold (D) of the ROC curve. Thus, we obtained and fixed the threshold $D = 0.285$ and

subsequently it is used in iteration $U = 20$ to obtain the confusion matrix. From the confusion matrix, we calculated the true positive and false positive rates, resulting in $TPR = 95.91\%$ and $FPR = 4.08\%$. In addition, we explainable knowledge from the RF model by using the SHAP method, demonstrating the influence of each feature in the prediction outcomes. Hence, we provide the top ranking of features and demonstrate that eight of the eleven features considered were created in feature engineering tasks (see Section IV-D2). Moreover, we find that *Average_Station_Hourly* (created in the feature engineering task) represents the most influential feature for the RF model. Lastly, the proposed model (RF model) is deployed in a micro-service which is connected to the SLV platform, using its data to train and predict anomalies in the bag's loading process. We also build a Dashboard using the PowerBI tool, which aims to make easier visualization of model predictions, as well as several useful pieces of information for a better comprehension of the prediction outcomes.

The proposed ML approach provides operational and financial advantages for the organization. Regarding the operational advantages, it allows getting information in advance regarding probable blockage of transportation vehicles inside of the factory at the check-out stage and consequently enables monitoring and inspecting the entire process in order to prevent long delays to solve the problem or restarting the entire process. Moreover, security questions such as the violation of federal regulations/norms of maximum gross vehicle weight (e.g., overweight transportation vehicles) can be avoided. The financial advantage is associated with the possibility of avoiding sending orders in quantities greater than the order. In future work, we plan to train the explored ML classifiers with a large historical dataset, explore techniques to monitor and estimate the performance of ML models (confidence-based performance estimation and direct loss estimation) post-deployment (without access to targets) and detect data drift using tools such as NannyML⁹, measure the impact of a model misclassification on the logistic performance of the concerned company, explore other classifiers and also AutoML tools for benchmarking purposes, and create new features to improve the prediction performance of the proposed ML models.

REFERENCES

- [1] T. Hastie, R. Tibshirani & J. Friedman, *The Elements of Statistical Learning*, 2nd ed., New York, NY, USA: Springer, 2009, pp. 1-745.
- [2] D. Singh & B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol.97, Part B, pp. 1-23, Dec. 2020.
- [3] CSCMP, "Supply Chain Operations Reference (SCOR) model," *Council Of Supply Chain Management Professionals*, pp. 1-24, 2010.
- [4] T. Chen & C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proc. of The 22nd ACM SIGKDD*, San Francisco, CA USA, 2016, pp. 785-794.
- [5] C. Bentéjac, A. Csörgő & G. Martínez-Muñoz, "A Comparative Analysis of Gradient Boosting Algorithms," *Artificial Intelligence Review*, vol. 54, pp. 1937-1967, Mar. 2021.
- [6] J. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," *The Annals Of Statistics*, vol. 29, no. 5, pp. 1-39, Oct. 2001.
- [7] P. Cortez, "Data Mining with Multilayer Perceptrons and Support Vector Machines," in *Data Mining: Foundations And Intelligent Paradigms*, vol. 24, D.E.Holmes & L.C. Jain, Berlin, Heidelberg, DE: Springer pp. 9-25, 2012.
- [8] V. Vapnik, *The Nature of Statistical Learning Theory*, 2nd ed., New York, NY, USA: Springer, 2000, pp. 1-314.
- [9] M. Rafiq, G. Bugmann & D. Easterbrook, "Neural network design for engineering applications," *Computers & Structures*, vol. 79, no. 17, pp. 1541-1552, Jul. 2001.
- [10] J. Burez & D. Van den Poel, "Handling class imbalance in customer churn prediction," *Expert Systems With Applications*, vol. 36, no. 3, pp. 4626-4636, Apr. 2009.
- [11] R. Schapire, "The Strength of Weak Learnability," *Machine Learning*, vol. 5, pp. 197-227, Jun. 1990.
- [12] G. Biau & E. Scornet, "A random forest guided tour," *Test*, vol. 25, pp. 197-227, Apr. 2016.
- [13] R. Couronné, P. Probst & A. Boulesteix, "Random forest versus logistic regression: A large-scale benchmark experiment," *BMC Bioinformatics*, vol. 19, no. 270, pp. 1-14, Jul. 2018.
- [14] R. Fatahi, H. Nasiri, E. Dadfar & S. Chehreh Chelgani, "Modeling of energy consumption factors for an industrial cement vertical roller mill by SHAP-XGBoost: a "conscious lab" approach," *Scientific Reports*, vol. 12, no. 7543, pp. 1-13, May 2022.
- [15] A. Vieira *et al.*, "Using simulation to model the logistic operations of a company of the cement industry," *FME Transactions*, vol. 47, no.4, pp. 683-690, 2019.
- [16] B. Noche & T. Elhasia, "Approach to Innovative Supply Chain Strategies in Cement Industry: Analysis and Model Simulation," in *Proc. - Social And Behavioral Sciences*, vol. 75, pp. 359-369, Apr. 2013.
- [17] M. Barbosa *et al.*, "Managing supply chain resources with Big Data Analytics: a systematic review," *Int. Journal Of Logistics Research And Applications*, vol. 21, no. 3, pp. 177-200, 2018.
- [18] N. Japkowicz & S. Stephen, "The class imbalance problem: A systematic study," *Intelligent Data Analysis*, vol. 6, no. 5, pp. 429-449, Oct. 2002.
- [19] N. Chawla *et al.*, "SMOTE: Synthetic minority over-sampling technique," *Journal Of Artificial Intelligence Research*, vol. 16, pp. 321-357, Jun. 2002.
- [20] J. Gonçalves, M. Sameiro Carvalho & P. Cortez, "Operations research models and methods for safety stock determination: A review," *Operations Research Perspectives*, vol. 7, 2020.
- [21] Y. Sun, A. Wong & M. Kamel, "Classification of imbalanced data: A review," *Int. Journal Of Pattern Recognition And Artificial Intelligence*, vol. 23, pp. 687-719, 2009.
- [22] J. Bergstra *et al.*, "Algorithms for hyper-parameter optimization," in *25th Annual Conference On Neural Information Processing Systems*, Granada, Spain, 2011, pp. 1-9.
- [23] L. Breiman, "Random forests," *Machine Learning*, pp. 5-32, 2011.
- [24] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, pp. 861-874, 2006.
- [25] C. Krauss, X. Do & N. Huck, "Deep neural networks, gradient-boosted trees, random forests: Statistical arbitrage on the S&P 500," *European Journal Of Operational Research*, vol. 259, pp. 689-702, 2017.
- [26] D. Larose, "Discovering Knowledge in Data: An Introduction to Data Mining," 2nd ed., USA: John Wiley & Sons, 2014, pp. 336.
- [27] S. Li & X. Zhang, "Research on orthopedic auxiliary classification and prediction model based on XGBoost algorithm," *Neural Computing And Applications*, vol. 32, pp. 1971-1979, Apr. 2020.
- [28] A. Dwivedi, "Performance evaluation of different machine learning techniques for prediction of heart disease," *Neural Computing And Applications*, vol. 29, pp. 685-693, May 2018.
- [29] I. Witten, E. Frank, M. Hall & C. Pal, "Data Mining: Practical Machine Learning Tools and Techniques," *Data Mining: Practical Machine Learning Tools And Techniques*, pp. 1-621, 2016.
- [30] P. Domingos, "A few useful things to know about machine learning," *Communications Of The ACM*, vol. 55, pp. 78-87, Oct. 2012.
- [31] D. Simchi-Levi, P. Kaminsky, & E. Simchi-Levi, *Designing and managing the Supply Chain: Concepts, Strategies, and Case Studies*, USA: McGraw-Hill Companies, 2000, pp. 1-354.
- [32] S. Chopra, & P. Meindl, *Supply Chain Management: Strategy, Planning, and Operation*, 6th ed., UK: Person Education Limited, 2016, pp. 1-516.
- [33] K. Croxton, "The Order Fulfillment Process," *The International Journal Of Logistics Management*, vol. 14, pp. 19-32, 2003.
- [34] F. Lin & M. Shaw, "Reengineering the order fulfillment process in supply chain networks," *Int. Journal Of Flexible Manufacturing Systems* vol. 10, pp. 197-229, 1998.
- [35] P. Chapman *et al.*, "CRISP-DM -Cross-Industry Standard Process for Data Mining - 1.0 Step-by-step data mining guide," *CRISP-DM Consortium*, pp. 76, 2000.
- [36] N. Oliveira, P. Cortez & N. Areal, "The impact of microblogging data for stock market prediction: Using Twitter to predict returns, volatility, trading volume and survey sentiment indices," *Expert Syst. Appl.*, vol. 73, pp. 125-144, 2017.
- [37] A. Brintrup *et al.*, "Supply chain data analytics for predicting supplier disruptions: a case study in complex asset manufacturing," *International Journal Of Production Research*, vol. 58, pp. 3330-3341, 2020.
- [38] H. Marshall & G. Murphy, "Factors affecting the accuracy of weighbridge systems," *Int. Journal Of Forest Engineering*, vol. 14, pp. 67-79, 2003.
- [39] A. Adadi & M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52138-52160, 2018.
- [40] M. Hollander, D. Wolfe & E. Chicken, *Nonparametric statistical methods*, 3rd ed., USA: John Wiley & Sons, 2013, pp. 1-848.
- [41] A. Barredo Arrieta *et al.*, "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82-115, 2020.

⁹<https://www.nannyml.com/>

- [42] M. Halimic & W. Balachandran, "Kalman filter for dynamic weighing system," in Proc. of The IEEE Int. Symposium on Ind. Electronics, Athens, Greece, 1995, vol. 2, pp. 786-791.
- [43] W. Lin & L. Huang, "Design of Port Unattended Weighbridge System Based on Internet of Things," in 3rd Int. Conf. AMCCE, Dalian, China, 2018, pp. 194-198.
- [44] S. Kim, J. Lee, M. Park & B. Jo, "Vehicle signal analysis using artificial neural networks for a bridge weigh-in-motion system," *Sensors*, vol. 9, pp. 7943-7956, 2009.
- [45] O. Gungor, I. Al-Qadi & J. Mann, "Detect and charge: Machine learning based fully data-driven framework for computing overweight vehicle fee for bridges," *Automation In Construction*, vol. 96, pp. 200-210, 2018.
- [46] Q. Liu *et al.*, "Deep learning model comparison for vision-based classification of full/empty-load trucks in earthmoving operations," *Applied Sciences*, vol. 9, pp. 4871, 2019.
- [47] W. Yan *et al.*, "Probabilistic machine learning approach to bridge fatigue failure analysis due to vehicular overloading," *Engineering Structures*, vol. 193, pp. 91-99, 2019.
- [48] T. Akter & S. Hernandez, "Truck industry classification from anonymous mobile sensor data using machine learning," *Int. Journal Of Transportation Science And Technology*, vol. 11, pp. 522-535, 2022.
- [49] S. Park *et al.*, "Diagnosis of problems in truck ore transport operations in underground mines using various machine learning models and data collected by internet of things systems," *Minerals*, vol. 11, pp. 1128, 2021.
- [50] N. Tahaei *et al.*, "Machine learning of truck traffic classification groups from weigh-in-motion data," *Machine Learning With Applications*, vol. 6, pp. 100178, 2021.
- [51] Z. Yao *et al.*, "Deep learning-based prediction of piled-up status and payload distribution of bulk material," *Automation In Construction*, vol. 121, pp. 103424, 2021.
- [52] Y. Zhou *et al.*, "Novel methodology for identifying the weight of moving vehicles on bridges using structural response pattern extraction and deep learning algorithms," *Measurement*, vol. 168, pp. 108384, 2021.
- [53] M. Deniz *et al.*, "Single Image based Volume Estimation for Dump Trucks in Earthmoving using Machine Learning Approach," *EPiC Series In Built Environment*, vol. 3, pp. 380-388, 2022.
- [54] M. Ribeiro, S. Singh & C. Guestrin, "Why should i trust you?" Explaining the predictions of any classifier," *Proc. 22nd ACM SIGKDD, San Francisco, CA, USA*, pp. 1135-1144, 2016.
- [55] R. Peres *et al.*, "Industrial Artificial Intelligence in Industry 4.0 - Systematic Review, Challenges and Outlook," *IEEE Access*, vol. 8, pp. 220121-220139, 2020.
- [56] M. Sahakyan, Z. Aung & T. Rahwan, "Explainable artificial intelligence for tabular data: A survey," *IEEE Access*, vol. 9, pp. 135392-135422, 2021.
- [57] S. Lundberg & S. Lee, "A unified approach to interpreting model predictions," in Proc. of the 31st Int. Conf. NIPS, Long Beach, CA, USA, vol. 30, 2017.
- [58] P. Pereira, P. Cortez, & R. Mendes, "Multi-objective grammatical evolution of decision trees for mobile marketing user conversion prediction," *Expert Systems with Applications*, vol. 168, pp. 114287, 2021.
- [59] G. Baryannis, S. Dani, & G. Antoniou, "Predicting supply chain risks using machine learning: The trade-off between performance and interpretability," *Future Generation Computer Systems*, vol. 101, pp. 993-1004, 2019.
- [60] K. Zuting *et al.*, "A synchronized strategy to minimize vehicle dispatching time: a real example of steel industry," *Advances in Manufacturing*, vol. 2, pp. 333-343, 2014.
- [61] C. Wang *et al.*, "Simulation-based optimisation of logistics distribution system for an assembly line with path constraints," *International Journal of Production Research*, vol. 52(12), pp. 3538-3551, 2014.
- [62] A. Potter, B. Yang & C. Lalwani, "A simulation study of despatch bay performance in the steel processing industry," *European Journal of Operational Research*, vol. 179(2), pp. 567-578, 2007.

TABLE 5. List of acronym.

Acronym	Description
SC	supply chain
OFP	order fulfilment process
ML	machine learning
RF	random forest
SHAP	Shapley additive explanations
SCM	supply chain management
RW	rolling window
XAI	explainable knowledge
DL	deep learning
BDA	big data analytics
ANN	artificial neural network
B-WIM	bridge weigh-in-motion
SVM	support vector machine
FEQS	field earthmoving quantity statistics
SCOR	supply chain operation reference
ERP	enterprise resource planing
SQL	structured query language
EDA	exploratory data analysis
SMOTE	synthetic minority over-sampling technique
VIF	variable inflation factor
ROC	receiver operating characteristic
AUC	area under ROC curve
TPE	tree of parzen estimations
DT	decision tree
GBT	gradient-boosted tree
XGBoost	extreme gradient-boosting tree
MLP	multilayer perceptron
U	number of rolling window iterations
D	size of dataset
W	fixed training set size
T	fixed size of testing set
S	window slide size



JÚLIO BARROS received the M.Sc. in Telecommunication and Informatics Engineering from the University of Minho, in 2016. From 2016 to 2018, he worked as a researcher (software developer) in the P25: IGPM (Integrated Global Project Management of Industrialization Projects) R&D project at the same university. He is currently a Ph.D. candidate in Advanced Engineering Systems for Industry and Data scientist at DTX - Digital Transformation CoLab, Guimarães, Portugal.

He is also an Invited Assistant Professor at the Department of Information Systems, University of Minho and a member of the ALGORITMI/LASI Research Centre. His main research interests lie in Machine Learning and Decision Support Systems (see <https://jlb.dsi.uminho.pt/>).

APPENDIX



research interests include Machine Learning and Data Visualization.

FRANCISCO CUNHA received the M.S.c. degree in Engineering and Management of Information Systems from the University of Minho, Portugal, in 2022. During 2021, he worked as a Data Visualization researcher for EngageLab and Lab2PT. From the end of 2021, he started work as Data Scientist at Cachapuz - Weighing & Logistics Systems, Lda in the R&D project NeWeSt. Currently, he is a data scientist at YOURDATA HD Digital Campus, Metyis group. His current



Business Area for both national and international markets. He is also leading R&D teams, which work on projects based on IoT, Augmented Reality, Computer Vision and Industry 4.0 concepts.

CÂNDIDO MARTINS received the B.S. degree in Systems Engineering and Computer Science and an MBA in Management, Marketing and Communication. His professional career is linked to Cachapuz Bilanciai Group, a global market reference in the implementation of innovative and highly technological solutions in the logistics, dispatching and weighing areas. He is currently the Executive & Innovation Manager of Cachapuz and is responsible for the Software and Automation



series of plasma processes for surface activation and deposition of multifunctional thin films. He is specialized in the research and development of smart materials for integration in different sensing applications. Paulo Pedrosa is currently the CTO of Advanced Materials at DTx: Digital Transformation CoLab where he leads the Functional and Sensitive Materials Group.

PAULO PEDROSA received his M.Sc. in Materials Engineering from Universidade do Minho, Portugal, in 2008 and in 2014 completed his Ph.D. in Biomedical Engineering at the University of Porto, Faculty of Engineering, Portugal. In his Ph.D. he developed novel concepts of biomedical electrodes to be used in EEG applications. In his post-docs, both at FEMTO-ST Institute, France, and Centre of Physics, Minho University, Portugal, he developed innovative EEG hydrogels and a



ence, Machine Learning and Modern Optimization, has appeared in Journal of Heuristics, Decision Support Systems, Information Sciences and others (see <https://pcortez.dsi.uminho.pt/>).

PAULO CORTEZ is Full Professor at the Department of Information Systems, University of Minho, Portugal. He is also assistant director of the ALGORITMI R&D centre. From 2012 to 2015, he was Vice-President of the Portuguese Association for Artificial Intelligence (www.appia.pt). Currently, he is Associate Editor of the journals Decision Support Systems (Elsevier) and Expert Systems (Wiley). His research, within the fields of Decision Support, Data Science,

...