CIRP Journal of Manufacturing Science and Technology 44 (2023) 116–125

Contents lists available at ScienceDirect

CIRP Journal of Manufacturing Science and Technology

journal homepage: www.elsevier.com/locate/cirpj

Towards predictive analytics in internal logistics – An approach for the data-driven determination of key performance indicators☆

Max Wuennenberg [a,*], Konstantin Muehlbauer [b], Johannes Fottner [a], Sebastian Meissner [b]

[a] Chair of Materials Handling, Material Flow, Logistics, Technical University of Munich, Boltzmannstrasse 15, 85748 Garching bei Muenchen, Germany
[b] Technology Centre for Production and Logistics Systems, Landshut University of Applied Sciences, Braeuhausgasse 33, 84130 Dingolfing, Germany

ARTICLE INFO

Article history:
Available online xxxx

Keywords:
Data maturity
Data science
Internal logistics
Key performance indicators
Predictive analytics
Process analysis

ABSTRACT

Data-driven methods can leverage opportunities to optimize internal logistics systems. Performance metrics can be gathered by Machine Learning. However, missing information is a significant obstacle. Therefore, additional data is necessary. This article presents a procedure model for process analysis, from database preprocessing to gathering process insights, focusing on predictive analytics. A control theory-driven approach categorizes the data, being the guideline for the procedure model. Assistance is provided in selecting data processing methods and obtaining valuable insights. A case study validates the procedure model with a performance analysis by predicting breakdown occurrence, comparing k-nearest neighbors, decision tree algorithms and neural networks.

Introduction

Initial situation and motivation of the topic

Internal logistics systems (ILS) are the link between value-adding activities within manufacturing companies. Executing operations such as conveying or storage, successful production processes rely on the effectiveness of ILS. These systems can contain sub systems such as continuous conveyors, storage, and retrieval systems or automated guided vehicles [1]. Performance measurement systems are important tools that enable the objective-oriented evaluation of business processes. ILS can be analyzed using performance measurement systems, but due to their increasing heterogeneity and dynamics, it becomes more and more difficult to deduce key performance indicators (KPIs) manually. Changes in the product portfolio, a high variance in demand, and the volatility of supply chains lead to decentralized systems whose process chains incorporate various logistics functions [2]. A frequently used approach to analyze physical enterprise processes is the value stream method (VSM). However, this method can often only deal insufficiently with the aforementioned challenges [3]. Considering this, data science opens up the opportunity to enable automated, product family-overarching analysis concepts and thereby complement the traditional, analog VSM. Moreover, a specification of the method for the particular application scenario of ILS allows for even more objective-oriented analysis concepts [4]. The logistics-oriented VSM 4.0 is one such specification and is therefore able to deliver versatile process data from ILS [5]. In order to turn those results into actual benefits from the practitioner's perspective, it needs to be clarified which data can be analyzed in what manner and what results can thus be obtained. Setting up a feasible data structure within the ILS development process is an important challenge, especially with regard to transformability during the operating phase [6].

State of the art

VSM encompasses methods for process mapping, analysis, and improvements. Its purpose is a visualization of the process based on recordings and measurements that were taken on-site. Hence, standardized symbols and performance metrics have been developed [7]. One main disadvantage is that only the situation at the very moment of the process mapping is considered. Also, only one product family is analyzed for each value stream and the whole method focusses on production processes rather than ILS processes. Performance metrics are recorded manually, using stop watches or

☆ This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

* Corresponding author.
E-mail addresses: max.wuennenberg@tum.de (M. Wuennenberg), konstantin.muehlbauer@haw-landshut.de (K. Muehlbauer), j.fottner@tum.de (J. Fottner), sebastian.meissner@haw-landshut.de (S. Meissner).

https://doi.org/10.1016/j.cirpj.2023.05.005
1755-5817/

counting stock. This yields a high human working effort to map the process [3].

Several extensions have been proposed that aim for a compensation of these drawbacks. One of those focusses on the consideration of ILS processes. This includes specific basic functions of logistics and an adaptation of the respective performance metrics. Also, those metrics are considered as distribution functions in order to cope with the high variability of ILS processes [8]. Another approach puts the emphasis rather on Industry 4.0 issues and the digitization of processes. Like in the conventional VSM, both material and information flow are considered, but the information flow is analyzed and described on a more detailed level, e.g., also considering storage media. However, the process mapping is executed in the same manner as in the conventional VSM [4].

Logistics-oriented VSM 4.0 combines the advantages of the two aforementioned VSM extensions. Both the specifications of ILS and Industry 4.0 are taken into account. Also, several product variants can be considered in one value stream, and the digital collection of data points is enabled. This is achieved by building process boxes where both physical parameters as well as data sources, degree of automation, and tracking points are depicted. The method also contains the evaluation of the generated data in the shape of a data maturity model, with further applications like data mining (DM) or machine learning (ML) in mind. This helps practitioners with the choice of an appropriate data science method based on the existing data infrastructure. [5].

Apart from the manual process mapping, there are other approaches that focus on the identification of optimization potentials within processes, directly based on data-driven methods. One example for this is the parametrization of a simulation model with ML. Based on production data, important parameters are deduced automatically which helps simulation engineers to save time [9]. Furthermore, this simulation model can be used for process optimization in combination with business process modeling [10]. That approach does, however, not come with a systematic method to obtain a feasible data structure and to deduce sensible analysis steps from these findings. Without such a data structure, it remains difficult to properly parametrize a simulation model. Another approach deals with the prediction of ILS processes using ML with the objective of supporting the system planning [11]. Also, there is an approach where machine data is used to identify optimization potentials – under the use of a performance measurement system, an ML model and the additional implementation of sensor technology [12]. With specific regard to a process-centric examination of ILS, process mining is used to analyze and optimize storage systems using both structure- and time-based analyses [13]. The focus is however limited to the consideration of specific issues rather than trying to cover the broad variety of data sources.

Objective

There is a gap between the traditional VSM and its extensions for the ILS domain on the one hand and recent publications concerning data science on the other hand: It is important to obtain several categories of process data, and it is also important to have sophisticated data analysis methods at hand. But there is a lack at the intersection point where it needs to be clarified which type of analysis can be addressed using a particular category of data. Although this gap in the state of the art has been identified in previous work, it has not yet been solved. Thus, the potential of data analyses such as predictive analytics can be determined following a suitable procedure model to follow at that intersection point. This article presents such a link by introducing a categorization framework for process data and a subsequent association of the data. Subsequently, a conceptual procedure model for the determination of meaningful analysis methods based on the existing database (see Fig. 1) is introduced. The approaches presented in this article address the following two research questions (*RQ*):

1. How can input and output data for data-driven analyses in the ILS domain be categorized?
2. Which steps are necessary in a procedure model to implement data-driven ILS analyses based on real-world process data?

Thereby, the problem of a black box between physical processes and the desired analysis can be solved. Data science allows for a process analysis even without an analytical relationship between inputs and outputs at hand. The paper is structured as follows: *Design of a procedure model for the analysis of internal logistics systems* contains an assessment of the state of the art in ILS and digital process analysis. Approaches from the control theory domain are adapted to develop a data categorization framework. Subsequently, guidelines for data-driven analysis concepts are proposed, based on the maturity of the respective data to be analyzed. In *Case study: application in an internal logistics scenario*, the findings are applied to a case study with the scope of a predictive analytics application. The obtained results are evaluated and critically discussed in *Discussion*. The paper concludes with *References*.

Design of a procedure model for the analysis of internal logistics systems

Overview of internal logistics processes

ILS can be composed of numerous sub-processes, each one of which is usually linked to one basic internal logistics function. These functions cover – among others – conveying, sorting, storage, picking, and packaging. Exemplarily, conveying, storage, and picking are covered in more detail in the remainder of this article: Conveying describes the transportation of goods or persons by technical means specifically in a locally limited area [14]. These technical means can cover stationary systems components such as roller conveyors or belt conveyors, as well as vehicles like forklift trucks or tugger trains. Secondly, storage covers all intentional stays of goods within an ILS process [14]. This is usually accomplished with designated storage areas and systems, such as high racks. Picking, on the other hand, is the customer-specific consolidation of articles for a particular task including their merging and provision for dispatch [14]. Especially with regards to data analysis, an important characteristic of these processes is their degree of automation: Forklift trucks, tugger trains, etc. are either operated by people (thus, they are only automated to a small degree), or they operate in a fully automated manner, e.g., automated guided vehicles. Stationary conveying systems also often possess a high degree of digitalization, whereas picking processes require such complex tactile operations that they are still mostly manual. Storage systems can be operated *manually*, with *mechanized* devices such as forklift trucks, or in a fully *automated* manner with stacker cranes or automated small parts storage systems [15].

Fundamentals of data science

The transformation of data into information and finally into knowledge is a process that has increasingly gained the focus of enterprises, especially in recent years [16]. Insights from the intelligent processing of data can help to manage the growing complexity within production and logistics systems [17]. Core techniques include the application of artificial intelligence (AI), ML, and DM [18]. AI aims to approximate the behavior of computers to that of humans and thereby solve real-world problems more efficiently and effectively than humans [19]. ML attempts to learn automatically on the basis of (historical) data, e.g. by recognizing patterns in these data [20]. ML can be further divided into different types: *supervised*

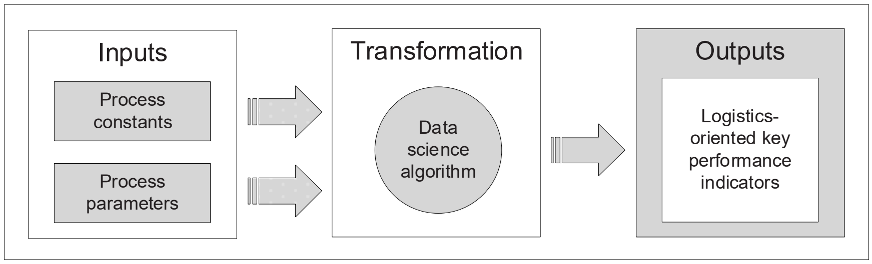

Fig. 1. Scheme of the presented approach.

learning, *unsupervised learning*, and *reinforcement learning*. Supervised learning needs previously labeled data so that the prediction can be assessed based on the accuracy with which labels are predicted. If there are however no predefined labels, unsupervised learning models can make assumptions on their own [21]. DM can be described as a process where knowledge is derived from data and then visualized [22]. This includes the collection, processing, analysis, and the visualization and interpretation of the available data [23,24].

Findings from the process of DM can be grouped into four different types: *descriptive*, *diagnostic*, *predictive,* and *prescriptive analysis*. These four types of data analytics cover a range from visualization to the detection of correlations to the prediction of future insight as well as the derivation of recommendations for action based on predictions [23–25]. Apart from the complexity of the application, the required data maturity is also a decisive aspect for the application of the respective type of data analytics [5].

Digital process data and key performance indicators

ILS are increasingly affected by digitization. Particularly in day-to-day business, a lot of data from different information systems is generated. Widely used information systems within production and logistics systems are enterprise resource planning systems (ERP), manufacturing execution systems (MES), or material flow computers (MFC) [26]. Through the bidirectional exchange of information, these digital systems are not only control instruments, but also fulfill a documentation purpose in the form of data logs and tables. The multitude of digital systems as well as the complex distribution of information in different notation and structure lead to a quantity of data that is difficult to process. Nevertheless, different data sources that appear in similar structures (albeit being stored in distributed locations) can be identified within the widespread information systems [27]. Based on these findings, intelligent and efficient processing of this data can make processes or entire systems more productive. A wide variety of KPIs or entire performance measurement systems can be used to evaluate the performance of logistics processes [28,29]. KPIs enable a comparability of the performance capabilities between processes, systems, and entire companies due to their largely standardized calculation. They are widely used in different forms within business processes as well as in the planning and control of production and logistics systems. However, the calculation and determination of KPIs is usually still done manually or semi-automatically using tools like spreadsheet programs. An essential requirement for the efficiency of a KPI system is the automated recording and evaluation of KPIs [28].

In this paper, special KPIs are taken into account, which can be determined automatically in many companies. The relevant KPIs for ILS, such as throughput time, throughput, inventory, and adherence to schedules, can serve as a starting point. In combination, these KPIs form the four relevant logistics targets [15]. Information systems that record transaction data from logistics processes are widely used in ILS and are therefore in the focus of the automated KPI calculation in this article. However, these KPIs can rarely be determined on the basis of data, as the information is often not recorded at all or only manually.

Machine learning algorithms

There is a multitude of different ML algorithms and new ones are continuously added due to ongoing research. Nevertheless, most algorithms can be divided into different classes. In the scope of this work, two main algorithm classes from the family of supervised learning can be distinguished which are regression and classification. Classification describes a procedure to identify the correct class of an input on different possible classes. Classification models can be built using different methods, such as simple thresholds, regression methods, or other ML techniques such as k-nearest neighbors (KNN), decision trees or artificial neural networks (ANN). The model learns from training data to classify the category or class of an input feature. The classifier can be a binary classifier, i.e. a choice between two classes, or a multi-class classifier, i.e. a choice between several classes [22,30]. In enterprises with a specification on ILS processes, the expertize in data-based analysis methods is mostly still in an early stage [31]. For this reason, in the following paragraph, only the most important and common ML algorithms are discussed. There are however related works that deal with further possibilities for algorithm selection in particular [21,32].

KNN describes a method in which elements are attributed to a group of other elements that share relatively similar feature values. That is, the predicted label is averaged between the labels of the k elements which are closest, according to all feature values. KNN can be used for both regression problems and classification problems. In this case, continuous values are grouped around the closest categorical values [30]. A decision tree is an ML model that builds a series of decisions based on different variables to get a result. The different feature variables are sorted according to their entropy so that the structure for a generalized yet utmost accurately predicting tree can be derived. Random Forest is a collection of decision trees to improve the prediction of all single decision trees [30]. Regressions create the possibility to predict values of continuous data within a series of values [30]. The aim is to extract a relationship within the available data in order to derive predictions about future developments on the basis of these findings. However, regression models can also be used to perform classification tasks. They rely on the assumption of an underlying mathematical function that the behavior of values follows. Hence, there are exemplary methods such as the polynomial regression (a special case of which is the linear regression) or the logistic regression [22]. ANN are another algorithm class that can perform both classification and regression tasks, especially when the accuracy of the previously mentioned ML models is no longer sufficient. A well-known representative of this class is the multilayer perceptron algorithm (MLP). The structure of an ANN consists of input, output, and hidden layers in between, which are all interconnected. The depth of the ANN's computational power can differ according to its number of layers [21,30,33].

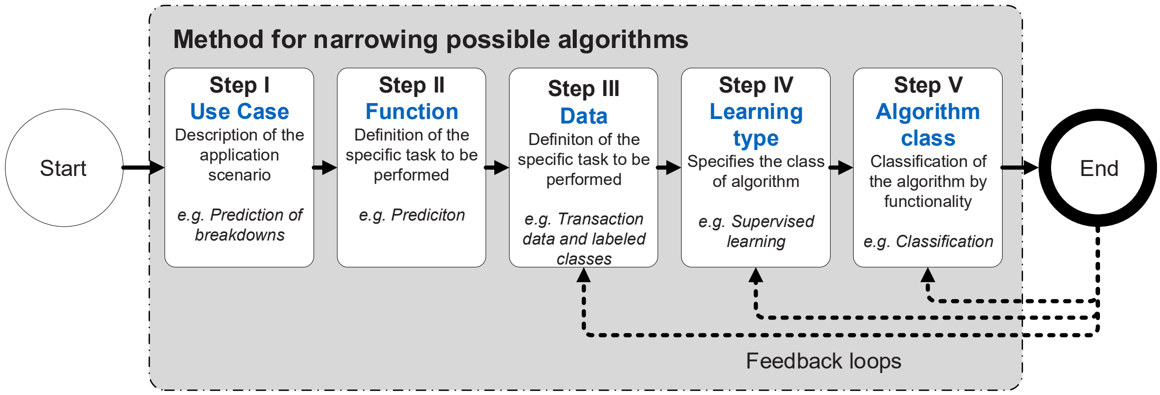

Fig. 2. Procedure for the selection of an algorithm.
Adapted from [34].

Based on the learning types and algorithm classes described above, these findings are also useful to describe a process for the rough selection of a suitable algorithm ([34], see Fig. 2). According to the widely used DM approaches, the first step is to describe the use case as good as possible in order to identify the relevant problem (step I: *Use Case*) [23,24]. Subsequently, the necessary function of the algorithm can be defined (step II: *Function*). In addition, the next step is to consider the available data (step III: *Data*). In this phase, it has to be checked which learning type can be addressed on the basis of the first points of the procedure (step IV: *Learning type*). The decision depends to a large extent on the available data structure. The results can be used to determine whether the provided data is labeled or non-labeled. Finally, the algorithm class can be identified and thus the selection of algorithms can be narrowed down (step V: *Algorithm class*).

Statistical evaluation of performance

The first step in a data science application is the exploration of the data. From that point, it can already be possible to determine first patterns or anomalies in the context of DM. At the beginning, the number of data points and the possible values for each data field as well as missing values can be determined. The most basic differentiation that can be made is ordinal and categorical data fields. Ordinal fields contain entries that can be compared by putting them in an order whereas this is not possible for categorical values. Ordinal values can, depending on the chosen scale, even allow for further statistical evaluation such as arithmetic mean, median, and standard deviation. Categorical values, on the other hand, can only be evaluated according to the frequency with which certain categories occur (a statistical measure for this being the mode). Furthermore, some data fields with ordinal values can be put into a relation by computing the correlation between them, which can serve as a hint for useful learning models: Correlating values might influence each other, so some of them possibly form parts of the input vector, while others have the potential to become elements of the output vector [35]. Apart from that, the exploration of data is a necessary step in order to compute the maturity of a given data set. This maturity assessment can then be used to determine if DM, ML, or AI algorithms can even be applied sensibly for a particular database [5].

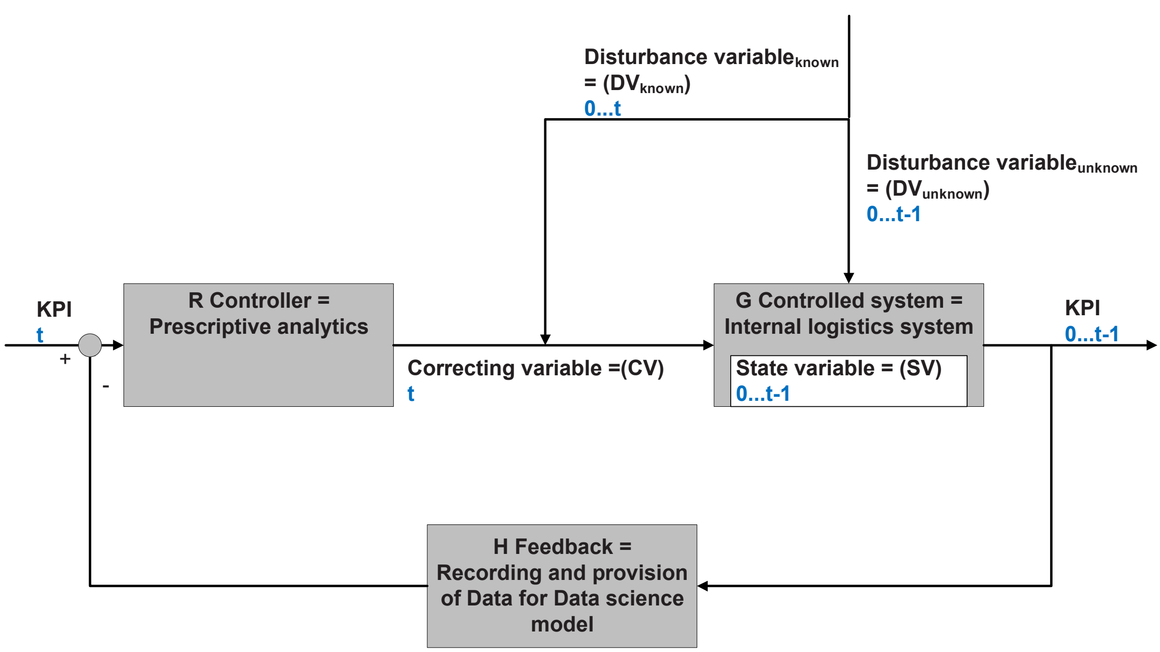

Fig. 3. System description according to a closed loop control.
Adapted from [36].

Data types	**Transaction data**	**Master data**	**Key Performance Indicator (KPI)**	
Timepoint	0...t-1	t	t	0...t-1
Definition	**Process variables** (= Process parameters) ≙ **State variables & Disturbance variables** $_{unknown}$	**Process constants** ≙ **Correction variables & Disturbance variables** $_{known}$	**Controlled variables** ≙ **Actual values**	**Target variables** ≙ **Command variables**
Examples	**Process variables, Process parameters, State variables** • Article identification number (ID) • Article type • Activity type • Conveyor ID • Number of articles • Position (localization) • Source and sink of transport task • Task ID • Timestamp • …	**Process constants, Correction variables** • Conveyor capacity • Loading scheme (pure / mixed) • Lot sizes • Number of workers • Organization type (e.g. predefined routes, direct deliveries) • Ressources (e.g., forklift trucks, workers) • Storage type / conveyor type • Type of load unit • Working times (e.g.breaks and shift times • …	**KPI, Target variables, Command variables, Controlled variables, Actual values** • Breakdown durations • Breakdowns and reliability • Capital commitment • Error ratio • Out-of-stock ratio • Turnover ratio • Utilization of available storage space • Utilization rate • Stock range • Stock level • Throughput • Throughput time • Timeliness • …	
	Disturbance variables $_{unknown}$ • Status of breakdowns • Quality-related data • …	**Disturbance variables** $_{known}$ • Article costs • Size and weight of articles • …		

Fig. 4. Categorization of system and process data.

When applying ML or AI approaches, the self-optimization is an important characteristic of the algorithm. Thus, a metric is necessary that this optimization can rely on. One common possibility for that is the *root mean squared error* (RMSE), which considers the deviation between actual and calculated value for every data point but weighs large deviations higher than small ones. At the same time, positive and negative deviations do not cancel each other out. However, RMSE is only suitable for ordinal data, not for categorical data. In the latter case, *precision* and *recall* are among the most common evaluation metrics. They compare true and false estimations of the model: While precision measures how many of the classifications are correct (ratio between true positives and overall number of elements attributed to the class), recall takes the proportion of classifications that are successfully detected (ratio between true positives and the overall mightiness of the class). The combination of those two metrics is called F-score [33].

A closed-loop control model for internal logistics

Fig. 3 shows the principle of a closed loop control system adapted to the performance measurement in ILS [36]. A vector of one or several *KPI* works as input and output of the system. That is, it contains the relevant information to evaluate the actual performance of the system (actual values) and compare it to the desired outcome (command variables). In terms of data science, *KPI* represents the labels of the data. The ILS itself is depicted as the controlled system *G*. This means that certain inputs (i.e., logistics objects) are transformed to outputs by changing their shape and / or their location, following external commands. Hence, *G* represents the physical world within this closed loop control. A procedure model for the optimization of the system (either manual or using automated, AI-driven algorithms) works as the controller *R*. The controller *R* yields the correcting variable *CV* that is used as input for *G*. Apart from *CV*, the behavior of *G* is also influenced by known and unknown disturbance variables Kn_{own} and $DV_{unknown}$. These represent (un)predictable alterations of the process conditions that are caused by external and internal influences. In order to enable the functionality of the controller as a regulating element in the process chain, it does not only rely on the command variables (*KPI*) as its input, but also compares them to the actual values and thereby determines the deviation between the two. This is made possible due to the feedback element *H*. In the domain of this paper, *H* is represented by the recording and provision of data for a data science model that *R* can work with. For each variable that appears in Fig. 3, the blue numbering indicates the time period(s) in which the variable is considered. That is, at a point in time *t*, for the command variables, only the value for current period is considered whereas for the disturbance variables, previous values are also relevant. However, only the known disturbance variables are also present in the current cycle – $DV_{unknown}$ is just available in periods *0* to *t - 1*, which covers all past periods since the beginning of the data recording.

Fig. 4 gives an overview on how to transfer the terminology of data management, particularly in the ILS domain, to the presented control theory terminology: Transaction data covers the so-called process variables or process parameters, such as sources and sinks for transportation tasks, task numbers, and status reports of the system. Depending on whether these values are actively set or caused by the process behavior, these variables can either be categorized as state variables or unknown disturbance variables. Both types of variables are related to a specific process instance, like a particular transportation unit. On the other hand, the master data of a system can be interpreted as process constants such as material flow resources (e.g., forklift trucks or workers), shift times, or size and weight of articles. Again, these variables can be categorized – depending on the influence that process operators have on them – as correcting variables or as known disturbance variables. These variables contain information that is not related to a certain process instance but is system-overarching at least over a certain period of time. Finally, KPIs (such as throughput time, out-of-stock ratio, or capital commitment) serve as the controlled variables and thus the actual values of the ILS process. At the same time, they are usually assigned to desired outcomes and can therefore also be interpreted as target variables or command variables. Regarding the data science domain, while KPIs serve as data labels, all input metrics contain the features for a data science model [5]. As a conclusion, it can be stated

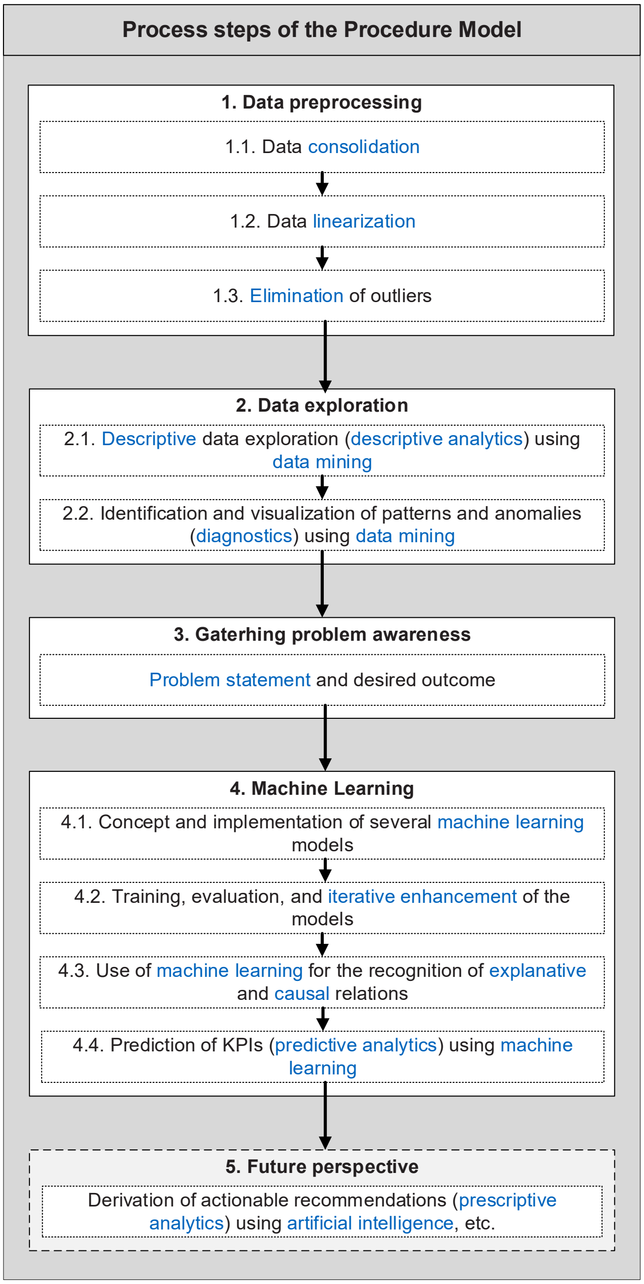

Fig. 5. Complete procedure model with individual steps.

that a sensible categorization of the available data is an enabler for subsequent data analyses.

Analysis steps

Consolidating the explained aspects of data science leads to a conceptual procedure model for process analysis. It is depicted in Fig. 5, where all necessary steps of the procedure model are consolidated: At first, all process data is preprocessed (step 1). In particular, this starts with a consolidation of the data in step 1.1. That is, data from software systems such as ERP and WMS is imported and merged into one data structure. Here, the concept of process variables and process constants based on the different data sources is initially applied. Step 1.2 is the linearization of data. This is essentially a conversion of non-numerical ordinal data into numerical values. In step 1.3, the data is further processed which covers the identification and elimination of outliers to improve the quality of generated analyses. Assuming that the reason for the occurrence of such outliers is clarified, there are several criteria to exclude them from the considered data. For example, outlier classification can be executed as follows: Considering the median value m and the interquartile distance d, values y from the data set with $|y - m| > 1.5{\cdot}d$ are eliminated [37]. By eliminating outliers, the prediction quality within the considered data generally grows, whereas it shrinks when considering the entire data set. Subsequently, a data exploration (step 2) helps to identify data fields with relationships that hold the potential for beneficial analytical insights. Suitable approaches cover, for instance, visualizations such as histograms or boxplots [38]. Additionally, a correlation matrix can help to quantify these relations. In step 2.1, DM algorithms allow for a more precise, descriptive data exploration, so that patterns and anomalies can be identified in step 2.2 (one example for such algorithms being k-means clustering) [39]. Both of these steps help to identify the effect that disturbance variables can have on the overall process, especially taking into account the possibility to control the process based on external decisions. Step 3 then contains the crucial decision for a problem posture and a desired outcome. That is, the subsequent data analyses need to be related to defined objectives. In the concept shown in Fig. 4, this relates to the choice of relevant *KPI*. For the determination of quantitative relations between data fields (which is a condition for predictive analytics), one or several ML models are implemented in step 4, following the control theory-related approach for the categorization of data fields. At first, promising ML models must be chosen (step 4.1). This also contains the conversion of continuous values into discrete data points in case a classification is desired rather than a regression. Suitable approaches cover, for instance, KNN, decision trees, or ANN, as has been explained in the previous subsection. These models are then trained and evaluated (step 4.2). The evaluation is executed by comparing the models from different algorithm classes, using relevant quality metrics such as the F-score, RMSE, correct estimates, or recall (see "*Statistical evaluation of performance*"). Step 4.3 leads to the recognition of quantitative relations, which might also possess the potential of being causal. In step 4.4 (which is the last step that is considered in this article's case study application scenario), the future development of KPIs can be predicted. In the step 5, prescriptive analytics finally focuses on deriving recommendations for action based on the prediction with regard to planning and control as well as process optimization. There are various possibilities for this, such as the use of rule-based systems, the use of a simulation mode, or the use of AI algorithms.

Case study: application in an internal logistics scenario

Description of the case study

The application of a case study is based on data from an automated storage system of a manufacturing company with a high number of variants. The storage system is the central point of all ILS processes within the plant. In addition to individual parts for assembly, semi-finished products from mechanical production are also stored in this warehouse. The warehouse has a capacity of around 15,000 storage spaces. Standardized containers are used for each storage location, and containers differ in terms of their segmentation so that several types of items can be transported in one. Thus, the number of different material numbers per container can vary from one to eight. The data of the system is transferred from an ERP system to the control system of the automated storage system and back again via electronic bookings. In addition, data on breakdowns is stored directly in the system. A material transaction in the system creates an entry in the ERP system using transport orders with several transport order positions. This makes it possible to identify the times of storage and retrieval (timestamps) as well as various

Table 1
Section of the data and presentation of the data structure of the case study.

Transportation order	Transportation order position	Sink	Source	Material No.	Time	Date	Quantity in units	Loading aid number	Operator ID
4590281	1	XZ3	259	39 4122	12:14:25	2021–08–22	100	1654	ID-KL
4590281	2	XZ2	259	61 9540	12:15:13	2021–08–22	150	1882	ID-KL
4590281	3	XZ6	259	34 9412	12:16:10	2021–08–22	50	1794	ID-KL
4593465	1	100	VY33	47 0481	12:22:46	2021–08–22	700	737	ID-GE
4596891	1	VY9	202	33 2409	07:14:33	2021–08–23	200	2198	ID-HD
….	….	….	….	….	….	….	….	….	….

information such as material numbers or quantities (These findings are the result of step 1 and 2 in the presented procedure model). The data set used contains 216 different working days in one year (explorative findings like this representing step 3). The structure of the data is shown in Table 1. The scope of the data set covers relevant phenomena such as shift times and scheduled interruptions of the production calendar, albeit potentially overlooking some aspects for which the consideration of at least an entire year or even longer periods might be necessary. Since the operators regularly execute test runs within the system, certain working days show unusually high numbers of breakdowns merely due to these tests. However, by applying the aforementioned criterion to eliminate values based on the median and the interquartile distance, these outliers could be removed in accordance with corporate process experts before applying the data analysis.

Results of the application

Fig. 6 shows the results of the application of this procedure model within the case study after preprocessing and data exploration (step 1). The number of breakdowns that occurred on the individual working days has been chosen as output variable (KPI) after data exploration and description / visualization (steps 2 and 3) using correlation matrices in combination with k-means clustering for DM. A high occurrence of breakdowns can lead to unexpected delays and generally increases the throughput time, which is why a solid knowledge about the proneness towards breakdowns of the system is desirable. A classification approach has been chosen, and the breakdown numbers were classified in eight different classes so that the bandwidth gap between class thresholds is growing by the power of two: 0, 10, 20, 40, 80, 160, 320, 640. Since the grouping of values is getting increasingly sparse within higher breakdown numbers, a similar frequency of values in each class can be reached. Other ways of determining thresholds (e.g., following a linear spacing) delivered a less even distribution of data points among the classes. This can lead to a deterioration of the achieved accuracy because the influence of certain classes is over-represented in comparison to others. The other (input) variables can be seen in the captions of all subplots: working day, number of transportation tasks (retrieval), number of transportation tasks (storage), number of positions in transportation tasks (retrieval), number of positions in transportation tasks (storage), number of transportation tasks (retrieval), number of conveying units (retrieval), number of conveying units (storage), number of articles (retrieval), number of articles (storage), working time in hours, number of workers. Furthermore, the data set was searched for outliers. Although potential candidates for outliers could be identified judged by their number of breakdowns, no explanations for this behavior could be determined. For this reason, it was decided that these values should not be excluded from the optimization algorithms. Several ML algorithms have been tested in order to generate predictions for the breakdown occurrence (random forest decision tree, conventional decision tree, MLP

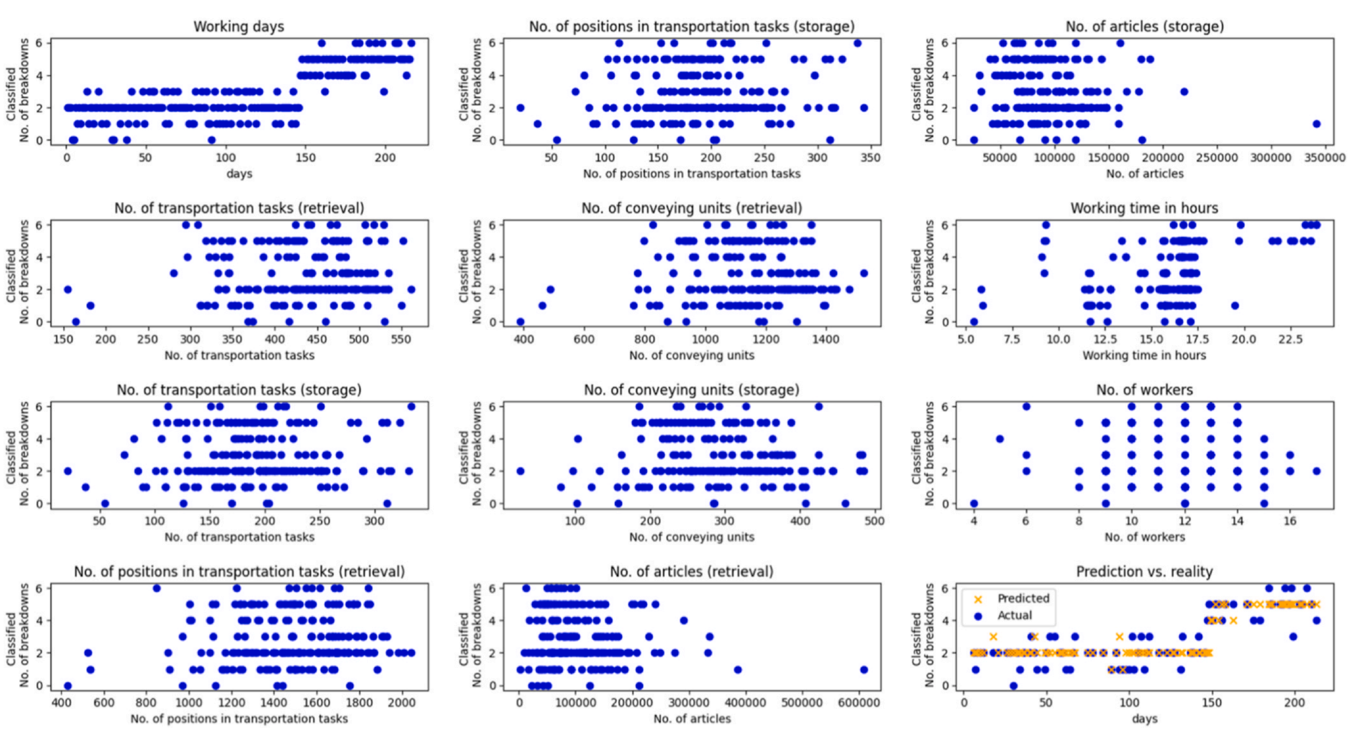

Fig. 6. Relationship between input and output data fields (The x-axis of each blue subplot represents the data field in the caption of the respective subplot, whereas the y-axis always represents the predicted breakdown occurrence class) The plot in the lower right corner compares actual and predicted value – the classified number of breakdowns – for each considered day. The prediction was generated using a gradient boosting decision tree classifier.

Table 2
Comparison between different prediction algorithms (for 87 data entries): the boldfaced values indicate the best results for each quality criterion.

Algorithm type	F-Score	RMSE	Correct estimates
Random forest decision tree	0.575	0.809	50
Conventional decision tree	0.494	0.884	43
multilayer perceptron artificial neuronal network	0.402	1.742	35
k-nearest neighbors	0.310	2.087	27
Gradient boosting decision tree	**0.586**	**0.719**	**51**

(ANN), KNN, gradient boosting decision tree), eventually leading to the choice of the gradient boosting decision tree whose results can also be seen in Fig. 6 (i.e., step 4.1–4.3 in the procedure model). The training data contains 70% of the values, while the test data contains the remaining values. It should be noted that the split between training and test data has been done randomly and not temporally. Thus, there is no specific point in time from the perspective of which all training values are in the past and all test values are in the future. In the lower right corner, the comparison between predicted and actual values for the most accurate prediction model are shown (these comparisons representing step 4.4).

The quantitative results of the application of five different classification models can be seen in Tables 2 and 3. All of those models were implemented with algorithms from the *Python* toolbox *Scikit-Learn*, and optimized with a hyperparameter *GridSearch*. As can be seen in Table 2, the *gradient boosting* decision tree classifier reaches the lowest RMSE, as well as the highest F-Score. At the same time, it reaches the most correct estimates. The two other decision tree models do not perform as good, but they still outclass the ANN and the KNN classifier. Apart from these classifiers, it would have also been possible to use regressor models. Those predict continuous values rather than discrete (categorical) ones, but they reach similar prediction quality. However, for the given data set, discrete values are more meaningful which is why this work focusses on the application of classifiers. This is due to the presence of nonlinear behavior and categorical data fields. In Table 3, the recall for each algorithm is depicted per breakdown frequency class. In this application, recall is more important to be considered than precision since it is less problematic to wrongly predict a breakdown than to neglect a breakdown in the predictions. It should be noted that the sum of all recalls per row is lower for the “preferable” algorithm type (*gradient boosting)* in comparison to the decision tree classifier, albeit having a higher F-Score, a lower RMSE, and more correct estimates. The reason why this is possible is that the individual classes do not all possess the same number of elements. Thus, a high recall in a class with many elements has a higher contribution to an overall good F-score than a high recall in a sparsely filled class. This phenomenon explains why, from an overall perspective, the gradient boosting decision tree is the preferable algorithm class for the defined problem with the given data set: class 3 and class 6 are the ones with the most entries, and the classifier reaches a recall of 1.

Discussion

Interpretation

The development and evaluation of the procedure model for ILS process data categorization and analysis answers the two research questions of this article: First, an approach for data categorization based on the closed loop model of control theory allows to categorize input metrics, based on whether they can be influenced or not, and whether they are specific for a certain process instance or not. On the other end of the presented conceptual procedure model are the output metrics, which leads to the second research question. The procedure model presented in this article contains a ten-step method from data consolidation to KPI prediction, with an eleventh step to be further detailed and tested in future research work.

The application of this conceptual procedure model in an ILS case study reaches an F-Score of almost 60%, which makes it considerably better than plain guessing (guessing can reach 50% in a classification with two classes – the more classes there are, the more difficult guessing becomes). That is, because of the complex relations between all involved process variables, there is no analytical model available upfront which could be applied instead. There are several ways of how a data set can be split into training and test data. In this work, a randomized train-test data split has been applied for all applications. The application of other data split methods such as a temporal split is still an open topic. In addition to that, a sophisticated detection and elimination of outliers might have the potential to improve the obtained results.

When assessing the internal validity of the generated results, the question arises if there could be further relevant factors, which influence the breakdown occurrence in this ILS process. Available data fields were considered in this model, but it is still possible that other (causal) influence factors can just not be tracked with the available data resources. Compared to existing approaches with a stronger focus on a more limited application scenario such as [9], there is still potential in terms of accuracy. Hence, further progress in process digitization yields the potential to increase the accuracy of the prediction. However, as it was mentioned, the model already delivers predictions that are significantly better than plain guessing. Assuming that no analytical model is available, this yields a considerable potential for practitioners. From the perspective of a process scientist, the goal is the consideration of data fields for the input and output that are inherently dependent. This does not diminish the meaningfulness of the generated results, but rather ensures that these results can at some point lead to process optimizations (which can only be effective if an input metric does have an influence on an output metric).

Regarding external validity, the generalizability of the procedure model needs to be critically discussed. The control theory approach for data categorization can be applied to the entire ILS domain as there is no specification for a certain process type. When executing the ten analysis steps however, along with explorative analyses of the data, it must also be ensured that as much process understanding as possible can be gathered. Without it, a sound choice of

Table 3
Comparison between different prediction algorithms: recall for each class of breakdown frequency. Boldfaced numbers indicate the highest recall per class. The headline of the table explains the classification into different solution classes by the number of breakdowns.

Algorithm Type	Class 1: 0	Class 2: 1–10	Class 3: 11–20	Class 4: 21–40	Class 5: 41–80	Class 6: 81–160	Class 7: 161–320
Random forest decision tree	0	0.14	0.94	0	0.29	0.81	0
Conventional Decision tree	0	**0.21**	0.66	**0.3**	**0.57**	0.44	**0.75**
Multi-layer perceptron ANN	0	0	**1**	0	0	0	0
KNN	0	0.14	0.54	0.2	0	0.25	0
Gradient boosting Decision tree	0	0	**1**	0	0	**1**	0
Test data	*1*	*14*	*35*	*10*	*7*	*16*	*4*

input-output relations cannot be made, especially not under the aspect of causality issues.

Limitations

From a critical point of view, it needs to be stressed that the developed procedure model relies highly on the validity of the data that is used for the data science application. This issue can be partially addressed by applying a data maturity framework: a low maturity means that the developed procedure model is not or only partially applicable. However, the challenge remains that features, and labels need to be chosen in an appropriate manner. If there is an analytical relationship between input and output data, the added value of data science is questionable. If that is not the case and the behavior of the system is considered a black box, it must be taken into account that a correlation between data fields does not necessarily mean there is causality as well. Hence, a correlation analysis alone is not always enough to ensure that the constructed model is fully suitable for training and use of AI models in the field of prescriptive analytics. They rely highly on causality, as there is otherwise no possibility to actively influence the behavior of the system. In other words, the controller element of the closed loop control cannot deliver proper correcting variables (in this article's case study, the execution of step 5 would be necessary to generate correcting variables).

Although the results that can be achieved with the presented ML algorithms already perform significantly better than plain guessing, there is still room for further improvements by considering more sophisticated analysis approaches. Since this article focusses on a feasible specification for the ILS domain rather than perfecting the choice of an optimal algorithm, these algorithms are out of its scope.

As for the influence of disturbance variables, the validity of data is important as well. Not every disturbing influence is obvious at the beginning. When there are important factors which remain unconsidered, the quality of predictions (not to mention prescriptions) is significantly lower.

Finally, since the developed procedure model addresses the entire ILS domain, further testing of its analytic part in other scenarios – including additional basic logistics processes – is necessary to assess its overall applicability in industrial use cases from a practitioner's perspective. Additional benefits could be reached by automating the procedure model so that less manual actions are required to execute the individual steps.

Conclusion and outlook

The method for a categorization of input and output performance metrics in this article enables a structured approach when tackling an ILS process analysis. Starting from a database that consists of data with a sufficient maturity, a practitioner is guided to categorize the data by their influence on the overall process. The model of a closed control loop can therefore be adapted to ILS. From that point, the right data foundation has been set up to determine which data-driven use scenarios are possible with the given data. This can cover descriptive analytics such as pattern recognition, predictive analytics, and finally prescriptive analytics. Thus, the presented procedure model could, for instance, also play a role in developing a decision support system for ILS operators. The more sophisticated an intended use case is, the more thoroughly must be planned which metrics serve as inputs and outputs of the model. The procedure model presented in this article allows to cover a wide range of process types within the ILS domain.

For future research directions, a more exhaustive testing of the procedure model with different data sets in various process environments is a necessary idea. Their complexity could be increased gradually, also by incorporating several source systems with the arising challenge that numerous data fields need to be merged. Also, different DM, ML, and AI approaches could be tested, also with alterations in their parameters, to extend the procedure model in this direction.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Fottner, J., Clauer, D., Hormes, F., Freitag, M., Beinke, T., Overmeyer, L., Gottwald, S.N., Elbert, R., Sarnow, T., Schmidt, T., Reith, K.B., Zadek, H., Thomas, F., 2021, Autonomous Systems in Intralogistics – State of the Art and Future Research Challenges. BVL – Bundesvereinigung Logistik, 14:2.

[2] Dörnhöfer, M., Günthner, W.A., 2017, A Research and Industry Perspective on Automotive Logistics Performance Measurement. IJLM, 28/1: 102–126.

[3] Forno, A.J.D., Pereira, F.A., Forcellini, F.A., Kipper, L.M., 2014, Value Stream Mapping: A Study about the Problems and Challenges Found in the Literature from the past 15 Years about Application of Lean tools. The International Journal of Advanced Manufacturing Technology, 72/5–8: 779–790.

[4] Meudt, T., Metternich, J., Abele, E., 2017, Value Stream Mapping 4.0: Holistic Examination of Value Stream and Information Logistics in Production. CIRP Annals, 66/1: 413–416.

[5] Muehlbauer, K., Wuennenberg, M., Meissner, S., Fottner, J., 2022. Data Driven Logistics-oriented Value Stream Mapping 4.0: A Guideline for Practitioners. In: Proceedings of the 18th IFAC Workshop on Control Applications of Optimization, Gif-sur-Yvette, Frankreich. 18.07.2022–22.07.2022.

[6] Wuennenberg, M., Vollmuth, P., Xu, J., Fottner, J., Vogel-Heuser, B., 2022. Transformability in Material Flow Systems: Towards an Improved Product Development Process. In: Proceedings of the 1st International Symposium on Industrial Engineering and Automation. Bozen-Bolzano, Italy. 21.6.2022–22.6. 2022.

[7] Rother, M., Shook, J., 2018. Learning to See: Value-stream Mapping to Create Value and Eliminate Muda, Version 1.5; 20th Anniversary Edition. Lean Enterprise Inst, Boston, p. 134.

[8] Knössl, T., 2013, Logistikorientierte Wertstromanalyse. Günthner WA, Boppert J, (Eds.) Lean Logistics. Springer Berlin Heidelberg, Berlin, Heidelberg: 135–144.

[9] Vernickel, K., Brunner, L., Hoellthaler, G., Sansivieri, G., Härdtlein, C., Trauner, L., Bank, L., Fischer, J., Berg, J., 2020, Machine-Learning-Based Approach for Parameterizing Material Flow Simulation Models. Procedia CIRP, 93:407–412.

[10] Wuennenberg, M., Wegerich, B., Fottner, J., 2022. Optimization of Internal Logistics using a combined BPMN and Simulation Approach. In: Proceedings of the 36th ECMS International Conference on Modelling and Simulation ECMS 2022. Ålesund, Norway. 30.05.2022–03.06.2022. Pirrot, Saarbrücken, pp. 13–19.

[11] Knoll, D., Prüglmeier, M., Reinhart, G., 2016, Predicting Future Inbound Logistics Processes Using Machine Learning. Procedia CIRP, 52:145–150.

[12] Ungermann, F., Kuhnle, A., Stricker, N., Lanza, G., 2019, Data Analytics for Manufacturing Systems – A Data-Driven Approach for Process Optimization. Procedia CIRP, 81:369–374.

[13] Knoll, D., Reinhart, G., Prüglmeier, M., 2019, Enabling Value Stream Mapping for Internal Logistics Using Multidimensional Process Mining. Expert Systems with Applications, 124:130–142.

[14] Hompel, M., ten Schmidt, T., Dregger, J., 2018, Materialflusssysteme. Springer Berlin Heidelberg, Berlin, Heidelberg.

[15] Wiendahl, H.-P., Reichardt, J., Nyhuis, P., 2014. Handbuch Fabrikplanung: Konzept, Gestaltung und Umsetzung wandlungsfähiger Produktionsstätten, 2., überarb. und erw. Aufl. ed. Hanser, München, p. 628.

[16] Tao, F., Qi, Q., Liu, A., Kusiak, A., 2018, Data-Driven Smart Manufacturing. Journal of Manufacturing Systems, 48:157–169.

[17] Choudhary, A.K., Harding, J.A., Tiwari, M.K., 2009, Data Mining in Manufacturing: A Review Based on the Kind of Knowledge. Journal of Intelligent Manufacturing, 20/5: 501–521.

[18] Schuh, G., Reinhart, G., Prote, J.-P., Sauermann, F., Horsthofer, J., Oppolzer, F., Knoll, D., 2019, Data Mining Definitions and Applications for the Management of Production Complexity. Procedia CIRP, 81:874–879.

[19] Russell, S.J., Norvig, P., 2016, Artificial Intelligence: A Modern Approach, third ed. Global Edition. Pearson, Upper Saddle River: 1132.

[20] Domingos, P., 2012, A Few Useful Things to Know about Machine Learning. Communications of the ACM, 55/10: 78–87.

[21] Goodfellow, I., Bengio, Y., Courville, A., 2017, Deep Learning. MIT Press Ltd, Cambridge, Mass: 800.

[22] Han, J., Pei, J., Kamber, M., 2012, Data Mining: Concepts and Techniques. third ed. Morgan Kaufmann; Elsevier, Amsterdam, Boston, Heidelberg, Amsterdam, Boston, Heidelberg: 703.

[23] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., Wirth, R., 2000. CRISP-DM 1.0: Step-by-Step Data Mining Guide.

[24] Fayyad, U., Piatetsky-Shapiro, Gregory, Smyth, P., 1996, From Data Mining to Knowledge Discovery in Databases. AI Magazine, 17/3: 37–54.

[25] Witten, I.H., Frank, E., Hall, M.A., 2011, Data Mining: Practical Machine Learning Tools and Techniques. Elsevier.
[26] Schuh, G., Stich, V., 2012, Produktionsplanung und -steuerung 1. Springer Berlin Heidelberg, Berlin, Heidelberg.
[27] Knoll, D., Prüglmeier, M., Reinhart, G., 2017, Materialflussanalyse mit ERP-Transportaufträgen: Automatisierte Ableitung und Visualisierung von Materialflüssen in der Produktionslogistik. Werkstattstechnik Online, 107/3: 129–133.
[28] Dörnhöfer, M., Schröder, F., Günthner, W.A., 2016, Logistics Performance Measurement System for the Automotive Industry. Logistics Research, 9:1.
[29] Syska, A., 1990, Entwicklung einer Vorgehensweise zur Bildung von betriebsspezifischen Logistik-Kennzahlensystemen. Springer, Berlin. 209 blz.
[30] Rebala, G., Ravi, A., Churiwala, S., 2019. An Introduction to Machine Learning. Springer International Publishing, Cham.
[31] Usuga Cadavid, J.P., Lamouri, S., Grabot, B., Pellerin, R., Fortin, A., 2020, Machine Learning Applied in Production Planning and Control: A State-of-the-art in the Era of industry 4.0. Journal of Intelligent Manufacturing, 31/6: 1531–1558.
[32] Bishop, C.M., 2009, Pattern Recognition and Machine Learning, Corrected at 8th Printing 2009 ed.. Springer, New York, NY: 738.
[33] Joshi, A.V., 2020. Machine Learning and Artificial Intelligence. Springer International Publishing, Cham.
[34] Böttcher, B., Klemm, D., Velten, C., 2017. Machine Learning im Unternehmenseinsatz: Künstliche Intelligenz als Grundlage digitaler Transformationsprozesse. Crisp Research.
[35] Cleve, J., Lämmel, U., 2014, Data Mining. De Gruyter Oldenbourg, München: 306.
[36] Leigh, J.R., 2012. Control Theory: A Guided Tour, third ed. The Inst. of Engineering and Technology, London, p. 444.
[37] Krzywinski, M., Altman, N., 2014, Visualizing Samples With Box Plots. Nature Methods, 11/2: 119–120.
[38] Grus, J., 2015. Data Science from Scratch: First Principles with Python, first ed. O'Reilly, Beijing, Köln, p. 311.
[39] Neubert, A., 2020, Clustering for Monitoring Logistical Processes in General Cargo Warehouses. Freitag M, Haasis H-D, Kotzab H, Pannek J, (Eds.) Dynamics in Logistics. Springer International Publishing, Cham: 431–440.