JOURNAL of
MATERIALS and MECHATRONICS:A

e-ISSN 2717-8811
JournalMM, 2022, 3(2), 240-256
https://doi.org/10.55546/jmm.1161542

Araştırma Makalesi / Research Article

A Method to Classify Steel Plate Faults Based on Ensemble Learning

Erkan Caner OZKAT[1]*

[1] Recep Tayyip Erdogan University, Faculty of Engineering and Architecture, Department of Mechanical Engineering, Rize, Turkey, ORCID ID: https://orcid.org/0000-0003-0530-5439, erkancaner.ozkat@erdogan.edu.tr

Geliş/ Received: 13.08.2022; **Kabul / Accepted:** 01.10.2022

ABSTRACT: With the industrial revolution 4.0, machine learning methods are widely used in all aspects of manufacturing to perform quality prediction, fault diagnosis, or maintenance. In the steel industry, it is important to precisely detect faults/defects in order to produce high-quality steel plates. However, determining the exact first-principal model between process parameters and mechanical properties is a challenging process. In addition, steel plate defects are detected through manual, costly, and less productive offline inspection in the traditional manufacturing process of steel. Therefore, it is a great necessity to enable the automatic detection of steel plate faults. To this end, this study explores the capabilities of the following three machine learning models Adaboost, Bagging, and Random Forest in detecting steel plate faults. The well-known steel plate failure dataset provided by Communication Sciences Research Centre Semeion was used in this study. The aim of many studies using this dataset is to correctly classify defects in steel plates using traditional machine learning models, ignoring the applicability of the developed models to real-world problems. Manufacturing is a dynamic process with constant adjustments and improvements. For this reason, it is necessary to establish a learning process that determines the best model based on the arrival of new information. Contrary to previous studies on the steel plate failure dataset, this article presents a systematic modelling approach that includes the normalization step in the data preparation stage to reduce the effects of outliers, the feature selection step in the dimension reduction stage to develop a machine learning model with fewer inputs, and hyperparameter optimization step in the model development stage to increase the accuracy of the machine learning model. The performances of the developed machine learning models were compared according to statistical metrics in terms of precision, recall, sensitivity, and accuracy. The results revealed that AdaBoost performed well on this dataset, achieving accuracy scores of 93.15% and 91.90% for the training and test datasets, respectively.

Keywords: Machine Learning, Classification, Ensemble Methods, Fault Detection, Artificial Learning.

*Sorumlu yazar / Corresponding author: erkancaner.ozkat@erdogan.edu.tr

Bu makaleye atıf yapmak için /To cite this article

Ozkat, E. C. (2022). A Method to Classify Steel Plate Faults Based on Ensemble Learning. Journal of Materials and Mechatronics: A (JournalMM), 3(2), 240-256.

1. INTRODUCTION

A product must be manufactured to meet the defined permissible upper and lower limits of each attribute and any deviation from these specified limits is considered a defect (Ozkat et al., 2017a; Liu et al., 2019; Kahveci et al., 2022). Without an effective monitoring and control strategy, today's complex manufacturing systems tend to produce faulty parts more frequently. These defects, defined by undesirable system dynamics, can lead to serious consequences such as a reduction in production, great economic losses, and unwanted downtimes (Ozkat et al., 2017b; Bektas et al., 2019; Gao et al., 2020). Early detection of defects and fault diagnosis is an important task in manufacturing to enhance the quality of the product and optimize the cost. With the integration of computer technology into production systems with Industry 4.0, the human factor has been minimized at every stage of production and it has enabled production to take place faster, with low cost and with a low margin of error (Xu et al., 2018; Kurt, 2019; Alkan & Bullock, 2021).

Steel is one of the most widely used materials in most engineering applications due to its strength, ductility, and recyclability (Lennox et al., 2000; Backman et al., 2019). One of the major challenges for the entire steel industry is the quality assurance of steel during manufacturing since steel goes through many different manufacturing processes from casting to drawing, pressing, to rolling (Widodo & Yang, 2007; Nkonyana et al., 2019). As a result, there are usually several kinds of defects on the steel plate that need to be localized and classified.

Traditionally, statistical processing control (SPC) methods have been deployed for monitoring quality during production, but they cannot predict the actual values relevant to product quality or estimate when the failure will occur. Because of this situation, machine learning (ML) models have been applied in a wide range in the field of manufacturing during recent years to solve real-world classification problems (Bektas et al., 2018; Ceryan et al., 2021; Özkat et al., 2021). The classification aims to accurately assign the sample to one of the predefined classes.

One issue regarding the steel quality control problem is the lack of large-scale, high-quality, industry-level, open-source datasets (Zhang et al., 2020). Due to the limited datasets, the well-known steel plate failure dataset has been used in many studies. The dataset is provided by the Semeion, Research Centre of Sciences of Communication, via Sersale 117, 00128, Rome, Italy (Buscema et al., 2010, Kaggle, 2017). The aim of many research works using this dataset is to correctly classify faults in steel plates employing the traditional ML models, neglecting the applicability of the developed ML models to real-world problems. With the integration of good and accurate models into manufacturing, defective steel plates can be identified as plates as early in the manufacturing process as possible, saving time and cost. However, manufacturing is a dynamic process, and constant adjustments and improvements are made. If the ML models cannot be updated with new data, they will quickly become obsolete and suffer a decline in accuracy. Therefore, it is necessary to establish artificial learning processes that determine the best model based on the arrival of new information.

The learning process often includes four main stages: data preparation, dimension reduction, model development, and model selection. Dimension reduction deals with the elimination of non-critical features without significant loss of information from the original dataset. This stage helps reduce computation time and develop a simpler structure in the machine learning model. Moreover, hyperparameter optimisation, which is one of the steps of the model development stage, is an important part of achieving a more accurate and updatable model. Regarding the studies using the steel plate fault dataset, the following ML models, namely: logistic regression (LR) (Fakhr and

Elsayad, 2012; Simić et al., 2014; Kharal, 2020; Gamal et al., 2021), support vector machine (SVM) (Simić et al., 2014; Tian et al., 2015; Nkonyana et al., 2019; Srivastava, 2019; Gamal et al., 2021; Tasar, 2022), k-nearest neighbour (kNN) (Srivastava, 2019; Gamal et al., 2021; Tasar, 2022), naive Bayes (NB) (Kazemi et al., 2018; Gamal et al., 2021), decision tree (DT) (Fakhr and Elsayad, 2012; Chen, 2018; Kazemi et al., 2018; Srivastava, 2019; Gamal et al., 2021; Tasar, 2022), random forest (RF) (Chen, 2018; Nkonyana et al., 2019; Srivastava, 2019; Kharal, 2020; Gamal et al., 2021; Tasar, 2022), neural network (NN) (Fakhr and Elsayad, 2012; Simić et al., 2014; Zhao et al., 2015; Kazemi et al., 2018; Nkonyana et al., 2019; Gamal et al., 2021; Tasar, 2022) have developed to address the fault classification problem. However, among all these ML models, studies involving hyperparameter optimization are rarely addressed (Tian et al., 2015; Zhao et al., 2015; Nkonyana et al., 2019; Kharal, 2020), while studies involving dimension reduction using feature selection step are not available. Instead, some studies have reduced the number of target classes (Zhao et al., 2015; Chen, 2018; Kazemi et al., 2018; Kharal, 2020; Gamal et al., 2021; Tasar, 2022). As a matter of fact, the description of the currently used dataset already states that the seventh target class is not unique. In addition, it is clearly stated in some studies that the data set is not divided into training and test data sets, and the success of the developed ML model is calculated over the entire data set (Srivastava, 2019; Tasar, 2022).

In contrast to the previous studies, this article provides a systematic modelling approach that includes the normalization step in the data preparation stage to reduce the effects of outliers, the feature selection step in the dimension reduction stage to develop an ML model with short computation time, and hyperparameter optimization step in the model development stage to increase the accuracy of the ML model. Furthermore, the ensemble machine learning model, which is a recent trend in the classification problems to overcome the individual drawbacks of each ML model (Yang et al., 2021; Pham et al., 2022; Xiong et al., 2022), was utilized to provide intelligent multi-class diagnostics for steel plates. The basic purpose of implementing ML models is to help operational decision-makers to organise effective and efficient manufacturing. The classification performances of the proposed models were computed using the following four statistical metrics: precision, recall, sensitivity, and accuracy score.

The rest of the paper is organized as follows: Section 2 presents a brief overview of some of the studies conducted using the provided dataset. Section 3 introduces the methodology used in the presented study in detail. Section 4 provides the results and is followed by concluding remarks in Section 5.

2. RELATED STUDIES ON THE STEEL PLATE FAULT DATASET

Classification is the process of finding which classes that new data belong to in a given dataset. The steel plate fault diagnosis dataset has been widely studied in machine learning for automatic pattern recognition. In this regard, relevant literature is summarized below.

Fakhr and Elsayad, (2012) employed a decision tree with boosting, a multi-perception neural network with pruning and logistic regression with step forward models, and tested their effectiveness using accuracy, specificity, and sensitivity. According to their results, the decision tree with boosting algorithm has achieved a remarkable performance with 97.25 and 98.09% accuracy on training and test sets. Similarly, Kazemi et al., (2018) studied decision tree, multi perception neural network, Bayesian network and ensemble random forest models. The data set is partitioned into 70% training

and 30% testing. It was indicated that the decision tree was superior to other models with reaching an accuracy score of 95.66 in both training and test.

Simić et al., (2014) utilized a remarkable approach by hybridizing random forest and bagging algorithms, called as Treebagger, and compare this novel algorithm against support vector machine, logistic regression, and multi perception neural network classification algorithms. The dataset was divided into training and test by the ratio of 70:30 in percentage, respectively. It was demonstrated that the Treebagger outperformed the other models in both training and test. However, the time required to create the tree bagger model reached up to four minutes, which is the most time-consuming task among others. A recent study conducted by Chen, (2018) reported that Adaboosting, another hybrid approach, could achieve 100% and 88.57% accuracy in the training and test set, respectively. In this study, ten-fold cross-validation was applied to each machine learning model, and other fault class was eliminated from the dataset in which the number of classes became 6.

Another important aspect of machine learning methodology is hyperparameter optimization. It is possible to further improve the performance of the model by choosing an optimal combination of hyperparameters that minimizes a predefined loss function. For example, Tian et al., (2015) utilized genetic algorithm (GA), grid search (GS) and particle swarm optimization (PSO) optimization methods to obtain the optimum hyperparameters for the support vector machine classification model. In addition, the classification accuracy was improved by normalizing all features. The implementation of GS, GA, and PSO in SVM yielded accuracy scores of 94.6%, 95.2%, and 88% for training, and 77.7%, 77.2%, and 78% for testing, respectively. Similarly, Zhao et al., (2015) integrated the local outlier factor (LOF) anomaly detection method with a back-propagation neural network to classify steel plate faults. Levenberg–Marquardt was employed to obtain optimal hyperparameters. As a result, the average training and test accuracy scores were 94.68% and 88.05%, respectively.

Kharal, (2020) performed the classification of faults on steel surfaces by applying optimization. It was found that the features in this dataset were imbalanced and to handle this problem undersampling, oversampling and synthetic minority oversampling technique methods were employed to balance the dataset. The best classification performance was obtained from an optimized random forest with 10-fold cross validation. Gamal et al., (2021) utilized the most common classification machine learning algorithms namely, decision trees, k-nearest neighbour, random forest, support vector machine, naive Bayes, logistic regression, and multi-layer perceptron neural network. The dataset was divided in half (50:50%) as training and test sets, and 10-fold cross validation was applied to each machine learning model. It was demonstrated that the accuracy scores of the listed machine learning methods were 91.14%, 82.86%, 93.29%, 86%, 59%, 88.29% and 73.86%, respectively. The lowest score was achieved using the naïve Bayes model, whereas the highest score was obtained using the random forest model. More interestingly, it was reported that the faults such as stains were easily classified, but other faults class could not be easily classified using any of these machine learning models. In addition, some faults such as Z scratch, and K scratch could be classified with less error depending on the performance of the machine learning.

Srivastava, (2019) showed that the random forest algorithm using a 20-fold cross validation achieved 79.23 % accuracy with 0.203 root mean square error on contrary to the k-nearest neighbour, decision tree, support vector machine, and deep neural network. In a tremendously similar and the most recent study conducted by Tasar, (2022) investigated the performance of linear discriminant, k-nearest neighbour, decision tree, support vector machine, random forest, and deep neural network machine learning model without applying data partitioning and feature selection. The accuracy score of each model was found as 90.136%, 91.7880%, 93.013%, 93.287%, 95.479%, and 96.986%,

respectively. The essential idea in machine learning is to test the performance of the developed model on a dataset completely independent of the data used during model development. In the presented, the data set was not divided into two as training and testing, and any feature selection method was not employed, but the cross-validation technique was applied for the developed models. In such a modelling method, it was thought that the models memorized the data set rather than learning. Therefore, the reliability of the obtained results is open to discussion.

To sum up, Table 1 groups the listed publications in terms of the feature selection step in the dimension reduction stage and the hyperparameter optimization step in the model development stage to increase the accuracy of the ML model.

Table 1. Comparison table for related studies on ML models for steel plate fault diagnosis

Feature Selection \ Hyperparameters Optimization	No	Yes
No	Fakhr and Elsayad, 2012 Simić et al., 2014 Chen, 2018 Kazemi et al., 2018 Srivastava, 2019 Gamal et al., 2021 Tasar, 2022	Tian et al., 2015 Zhao et al., 2015 Nkonyana et al., 2019 Kharal, 2020
Yes	N/A	Proposed in this study

3. METHODOLOGY FOR MULTICLASS FAULT DIAGNOSIS IN STEEL PLATES

The methodology flow used in the presented work is illustrated in Figure 1. The fishbone diagram consists of four main stages which are (i) data preparation, (ii) dimension reduction, (iii) model development (iv) model selection. The detailed information required for each stage is given in the following sections.

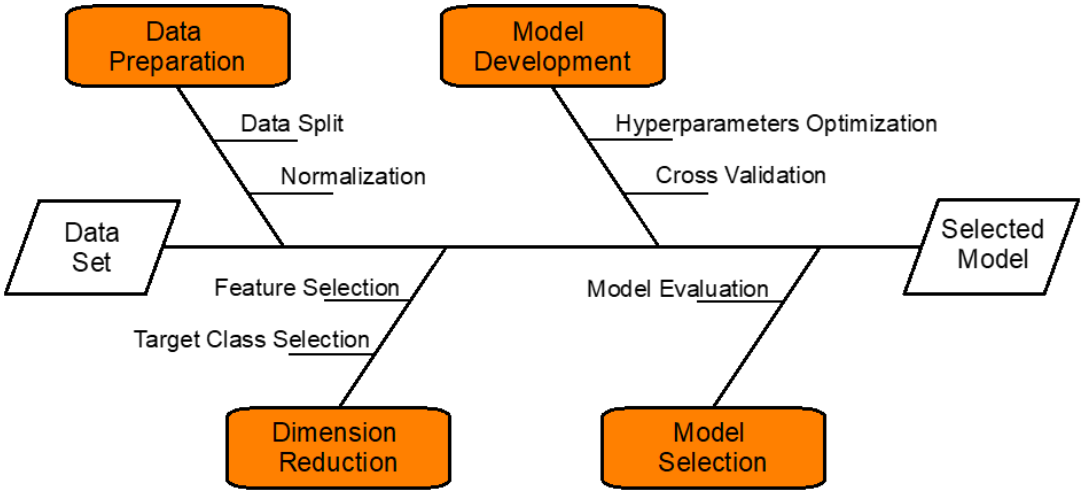

Figure 1. The methodology flow used in this study

3.1 Dataset Description

The dataset used in this study is a public dataset and it can be accessed through the online data science platform Kaggle (Kaggle, 2017). The dataset contains 1941 experiments. Each experiment has 27 independent features and one of 7 different types of classified steel surface defects, which are Dirtiness, Stains, Pastry, Z-Scratch, K-Scratch, Bumps, and Other Faults. The failure types and corresponding sample numbers are listed in Table 2, and detailed information on the 27 independent features is given in Table 3.

Table 2. Types of faults and sample sizes

Fault class	Failure types	Sample numbers
1	Dirtiness	55
2	Stains	72
3	Pastry	158
4	Z-Scratch	190
5	K-Scratch	391
6	Bumps	402
7	Other Faults	673

Table 3. Independent features of the faulty steel plates data set

Number	Feature	Type	Number	Feature	Type
1	X Minimum	Numerical	15	Edges Index	Numerical
2	X Maximum	Numerical	16	Empty Index	Numerical
3	Y Minimum	Numerical	17	Square Index	Numerical
4	Y Maximum	Numerical	18	Outside X Index	Numerical
5	Pixels Areas	Numerical	19	Edges X Index	Numerical
6	X Perimeter	Numerical	20	Edges Y Index	Numerical
7	Y Perimeter	Numerical	21	Outside Global Index	Numerical
8	Sum of Luminosity	Numerical	22	Log of Areas	Numerical
9	Minimum of Luminosity	Numerical	23	Log X Index	Numerical
10	Maximum of Luminosity	Numerical	24	Log Y Index	Numerical
11	Length of Conveyer	Numerical	25	Orientation Index	Numerical
12	Type of Steel A300	Categorical	26	Luminosity Index	Numerical
13	Type of Steel A400	Categorical	27	Sigmoid of Areas	Numerical
14	Steel Plate Thickness	Numerical	-	-	-

3.2 Data Preparation

3.2.1 Data split

The most crucial item of data preparation in machine learning methodology, which should not be neglected, is the separation of data into training and test sets. Training data is utilized to learn the patterns in the data and develop the machine learning model. Testing data is employed to test the model on unseen data to validate the results across different models. The factor to be considered here is the separation rate. The training set should not only contain a large amount of data to enable learning, but also a small amount of data so that the pattern in the data is not memorized. For this purpose, the dataset has been randomly split into 80% training data and 20% test data.

3.2.2 Normalization

Another step that should be applied in the data preparation stage in order to improve the machine learning performance is normalization. The purpose of this step is to move the values of only the numerical data into a common scale, without distorting the differences in the value ranges of the numerical data. The Z-score normalization technique was applied in this study, each numerical feature

was centred and scaled by the corresponding weighted mean and standard deviation. The method of Z-score normalization is given in Equation 1.

$$x_{i,j}^{*} = \frac{x_{i,j} - \mu_{x,i,j}}{\sigma_{x,i,j}} \tag{1}$$

where i is the index of the feature, j is the index of the sample, $x_{i,j}$ is the ith feature of the jth sample, $x_{i,j}^{*}$ is the ith normalized feature of the jth sample, $\mu_{x,i,j}$ is mean and $\sigma_{x,i,j}$ is standard deviation of the ith feature of the jth sample. It is very important to emphasize that normalization was first applied to the training dataset. The mean and standard deviation of each feature obtained in the training dataset was utilized to normalize the test dataset. The reason behind this approach is that test data should be independent of the training data in terms of information content.

3.3 Dimension Reduction

3.3.1 Target class selection

Traditionally, dimension reduction has been applied to features in machine learning, but in this study, it has been applied to both target classes and features. The reason for this is that target class 7 (i.e., Other Faults), as stated in the dataset definition, is not a specific kind of fault but a combination of several faults that are different fault from 1 to 6. It is quite difficult to determine samples of the 7th failure class as samples belonging to this class do not share certain features, also it is difficult to find dominant features to train. In addition, some features in class 7 may have similar properties to features in other classes. There are 673 samples of Other Failures that are not clearly classified. For this reason, as in some studies in the literature (Tian et al., 2015; Kazemi et al., 2018; Gamal et al., 2021), class 7 was excluded from the data set, as it would significantly affect the modelling.

3.3.2 Feature selection

It is not correct to use all the features in the model development stage. Since some features will cause errors and the developed model can diverge; hence, it is of great importance to detect and remove these features that are not relevant to the target class. Many methods are utilized to determine the features used in model development, among which the widely accepted method is Principal Component Analysis (PCA). Since it transforms high-dimensional data into low-dimensional data using the computationally simple linear algebra method, as it enables machine learning methods to converge faster when trained on the principal components rather than the original dataset. The main steps to be followed in PCA are standardization of the data, calculation of the covariance matrix and finding the eigenvalues and eigenvectors for the covariance matrix, respectively. While the eigenvector determines the principal component, the eigenvalue determines the magnitude corresponding to this principal component. Dimension reduction is performed by determining the eigenvectors corresponding to the eigenvalues with a magnitude above the pre-defined threshold value.

3.4 Model Development

Classification in machine learning is the problem of determining which class set a new observation belongs to, based on a training set containing examples of the known classes. Ensemble machine learning methods combine multiple ML models to obtain better predictive performance than could be obtained from any of the ML model alone. The final classification depends on the combined outputs of the individual models. The common ensemble classification techniques include boosting,

bagging and random forest. This section explains about different machine learning algorithms used in this study. All the proposed models were developed in the MATLAB R2022a environment. During model development, random seeds were used at training in hyperparameters optimization that performs reproducibility of models.

3.4.1 Hyperparameters optimization

Adaptive Boosting (Adaboost) is an ensemble model in which different weak classifiers are trained on the same training set and then these weak classifiers are combined to create a stronger classifier with a certain weight. This weak classifier can be any algorithm such as decision tree, and k near neighbour. In addition, the weight of a classifier is calculated according to the accuracy error that the model will make during the training phase. The boosting algorithm adopts an iterative approach which will tend to give more weight to misclassified samples in the hope that the next model will be more accurate. In this study, the weak classifier is selected as decision tree and three hyperparameters, maximum number of splits, number of learners and learning rate. These hyperparameters were varied in the range of 1-1014, 10-500, and 0-1, respectively.

Bootstrap Aggregation (Bagging) is another ensemble model which aims to create a set of classifiers having the same importance unlike boosting. The bagging and boosting algorithms appear the same, but the way to train the base classifier is completely different. For instance, given a data set containing “n” samples, select randomly one point from the training dataset and repeat this selection “N” times without replacement, eventually resulting in a new dataset in which some samples may appear several times while others may never appear. With this process, different training datasets and therefore different classifiers are created. Since the training method of each base classifier is independent and identical, an equal weighted strategy is used to vote by the classifier. Each model will vote on the outcome of the prediction and the overall output will be the class that has received the most votes. In this study, the weak classifier is selected as decision tree and three hyperparameters, maximum number of splits, number of learners and number of predictors to sample. These hyperparameters were varied in the range of 1-1014, 10-500, and 1-15, respectively.

The random forest model is an ensemble method that operates by building several decision trees trained on randomly sampled training data using the bootstrap sampling method. For each decision tree, a dataset is created by the bootstrap procedure. Constructing a large number of trees and aggregating them reduces the overfitting problem of a single tree and thus improves the generalization ability of the random forest. Two hyperparameters, minimum number of leaf sizes, and number of trees need to be optimized to obtain a good model. These hyperparameters were varied in the range of 1–500 and the minimum number of leaf sizes was chosen using the Bayesian method.

3.4.2 Cross validation

Cross validation is a method that ensures that the developed machine learning model is independent of the separation of the data set into training and test sets. Typically, the training dataset is divided into k parts. The machine learning model is trained on k−1 parts of the data, and the rest of the data is used for validating the model. This process is repeated k times to reduce the variance. The k-fold cross-validation method gets its name from this process. The results of each k-fold can then be averaged to produce a single result of the machine learning model. In the presented work, 10-fold cross-validation was applied to each developed model.

3.5 Model Selection

The developed machine learning models aim to determine the failure types that a steel plate may have using the selected features as inputs. Since it is a classification problem, the models

developed are evaluated using the following statistical metrics: precision, recall, sensitivity and accuracy. These metrics explain how well a target class is predicted or how bad if a prediction has missed the class. In addition, these metrics are defined using the confusion matrix. It has two dimensions matrix; the rows of the matrix represent samples of the actual classes and the columns represent the samples of the class predicted by the machine learning model. Generally, the confusion matrix is a result of the binary classification problem, which has only two classes to be classified, preferably one positive class and one negative class. However, the presented work is a multi-class classification problem that classifies samples into one of six classes, and no generalized formulae are provided for calculating the precision, recall, specificity, and overall accuracy of the model, having many classes to consider. Let us suppose that N, i and j represent the number of samples, the actual and predicted classes, respectively. An example of the multi-class confusion matrix is given in Table 4.

Table 4. An example of the multi-class confusion matrix

Predicted Classes / **Actual Classes**	**Class 1**	***Class 2***		**Class j**
Class 1	N_{11}	N_{12}	⋯	N_{1j}
Class 2	N_{21}	N_{22}	⋯	N_{2j}
⋮	⋮	⋮	⋱	⋮
Class i	N_{i1}	N_{i2}	⋯	N_{ij}

The numbers of true positive (TP), true negative (TN), false positive (FP), and false negative (FP) for each class i are computed in Equations 2-5., respectively (Markoulidakis et al., 2021). In addition, the formulation of the following metrics used in this study, precision (P), recall (R), and specificity (S) for each class i and the accuracy score (A) are given in Equation 6-9., respectively (Markoulidakis et al., 2021).

$$TP_i = N_{i,i} \tag{2}$$

$$TN_i = \sum_{\substack{j=1 \\ j\neq i}}^{n} \sum_{\substack{k=1 \\ k\neq i}}^{n} N_{jk} \tag{3}$$

$$FP_i = \sum_{\substack{j=1 \\ j\neq i}}^{n} N_{ji} \tag{4}$$

$$FN_i = \sum_{\substack{i=1 \\ i\neq j}}^{n} N_{ij} \tag{5}$$

$$P_i = \frac{TP_i}{TP_i + FP_i} \tag{6}$$

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{7}$$

$$S_i = \frac{TN_i}{TN_i + FP_i} \tag{8}$$

$$A = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n} \sum_{j=1}^{n} N_{ij}} \tag{9}$$

4. RESULTS AND DISCUSSION

The quality and quantity of the dataset have a huge impact on the performance of machine learning models. In this regard, the features and target classes of the dataset were initially thoroughly examined. As stated in Table 2, the type of steel is a categorial feature so it is important to examine the errors that occur according to the steel type before data splitting. Figure 2 illustrates the radar plot of fault classes by steel type. Upon closer inspection, only one sample of K Scratch and Stains was found for type A300 steel. Additionally, Dirtiness is also infrequent for this type of steel. On the other hand, Z Scratch is a relatively rare class of failure for steel type A400. Therefore, these samples along with samples that contain other faults class were excluded from the data set in order to improve the performance of the machine learning methods before data splitting.

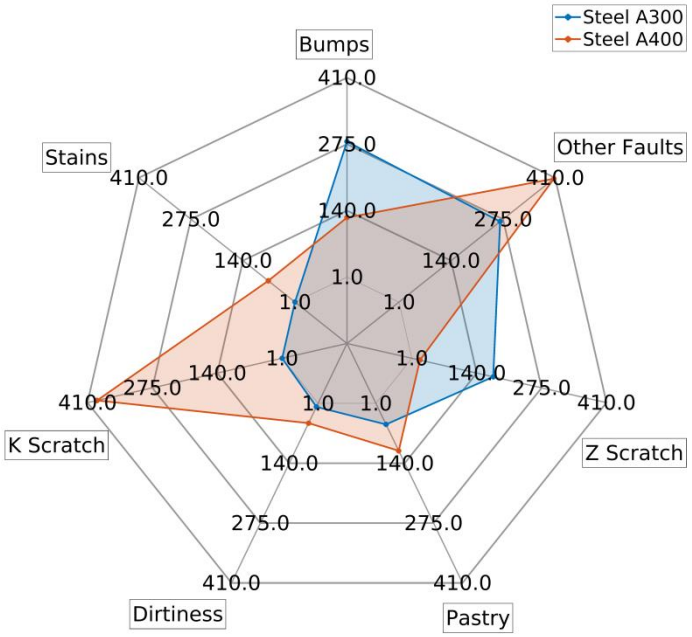

Figure 2. Fault classes according to the steel type

The dataset initially contains 1941 samples, after excluding samples with other faults class, along with samples with lesser fault classes when fault classes are classified according to the steel type, 1239 samples remained in the dataset. The dataset was randomly split into 80% training data (992 samples) and 20% test data (247 samples) before normalization. In the presented work, 'Type of Steel' and 'Outside Global Index' were considered as categorial features. The dimension reduction was conducted to the remaining features by ranking the metric and eliminating features that did not reach a certain score. In order to determine the most important feature influencing the target classes, PCA was conducted, and the importance score is presented in Figure. 3. It is evident that the elbow point is observed at 'X Maximum' feature which implies that the number of important features is 13,

and adding the two categorical features, the most important features obtained for modelling are 'Sum of Luminosity', 'X Minimum', 'Orientation Index', 'Minimum of Luminosity', 'Empty Index', 'Outside X Index', 'Y Minimum', 'Edges Y Index', 'Y Perimeter', 'Length of Conveyer', 'X Perimeter', 'X Maximum', 'Type of Steel' and 'Outside Global Index'.

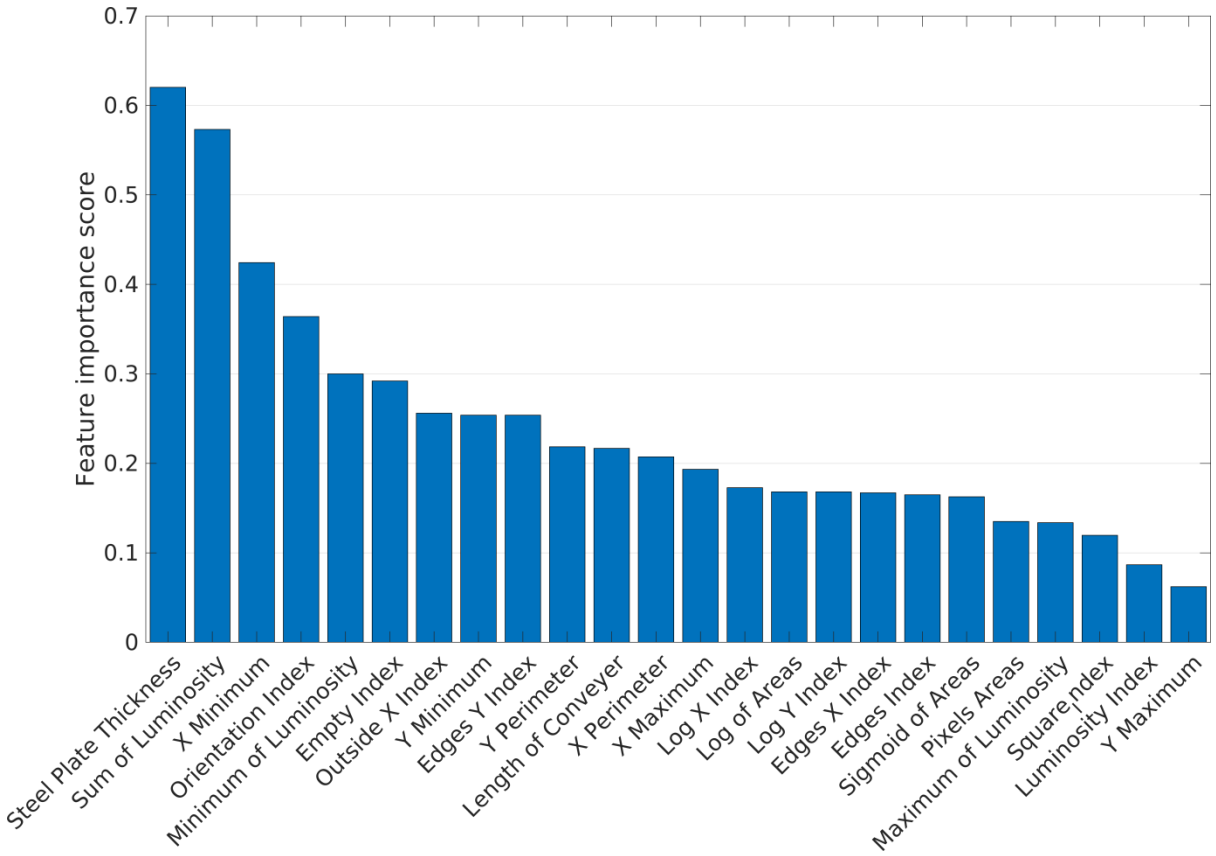

Figure 3. Feature importance ranking for dimension reduction

The optimum model was observed in the Adaboost model (i.e., ML1) with a learning rate of 0.9851, number of learners was 311, and maximum number of splits was 59. Similarly, the values of optimal hyperparameters for the Bagging model (i.e., ML2) were 174 for the number of learners, 345 for the maximum number of splits, and 15 for the number of predictors to sample. Moreover, the Random Forest model (i.e., ML3) performed best when the number of trees was 300, and the minimum number of leaf size was 1. The hyperparameters of each machine learning model were obtained during the training phase with 10-fold cross validation and, then, the trained models were utilized to predict the test set. Since the purpose of this study is to classify steel plate faults utilizing a machine learning model, it is important to determine the best models among the models developed.

Three statistical metrics namely, precision, recall and specificity are introduced to access the performance of the models developed. The results of these metrics for the training and test datasets are given in Tables 4 and 5, respectively. Moreover, the results are given in percentages, which means that a value close to 100 represents a good correlation between actual and predicted classes.

In brief, precision indicates the classifier's certainty of correctly predicting a particular class. In other words, it is the ratio of the number of TP to the total positive prediction including TP and FP. Therefore, FP, which is the cost of the model, are represented as part of precision. Once the models are accessed based on the precision value, the model with the highest precision would be chosen as the final model. On the other hand, recall and specificity represent the number of correct positive (TP) and negative (TN) predictions out of the total true positive and negatives, respectively. They assess the usefulness of the model in a single class. In this regard, the results clearly confirm that Adaboost (i.e., ML1) is the most effective model in fault detection with respect to all the

performance measures. The second-best effective model for detecting the faults in steel plates is the Bagging (i.e., ML2) model.

In Table 5, it can be observed that the highest precision is obtained using the Adaboost (i.e., ML1) model. With the ML1 model, a score of 100% was obtained in Dirtiness, K-Scratch and Z-Scratch, fault classes, and all the classes predicted as positive were found to be positive. For Recall, the highest scores were obtained with the ML1 model in the Bumps and Dirtiness fault classes, with the Bagging (i.e., ML2) model in the K-Scratch and Pastry fault classes, and with the Random Forest (i.e., ML3) model in the Stains and Z-Scratch fault classes. Accordingly, it was determined that the values estimated as TP in the fault classes related to each model were estimated correctly at the highest level. For specificity, the highest value in the Pastry fault class was obtained with the ML3 model, but the highest value in other fault classes was obtained with the ML1 model. Thus, it was determined that the values estimated as TN by the ML1 method were estimated correctly at the highest level.

Table 5. Precision, Recall and Specificity results for each fault class obtained from the training dataset

Fault Class	Training dataset								
	Precision (%)			Recall (%)			Specificity (%)		
	ML1	ML2	ML3	ML1	ML2	ML3	ML1	ML2	ML3
Bumps	87.826	87.021	91.447	94.099	91.615	86.335	93.731	93.433	96.119
Dirtiness	100	93.939	82.927	89.189	83.784	91.892	100	99.791	99.267
K-Scratch	100	98.722	95.652	98.397	99.038	98.718	100	99.412	97.941
Pastry	79.487	78.992	74.194	73.228	74.016	72.441	97.225	97.11	96.301
Stains	96.364	94.444	91.525	94.643	91.071	96.429	99.786	99.679	99.466
Z-Scratch	100	97.761	95.775	97.826	94.928	98.551	100	99.649	99.297

ML1: Adaboost, **ML2**: Bagging, **ML3**: Random Forest Ensemble models

The values of the statistical metrics, precision, recall, and specificity of each model were computed using the test dataset and presented in Table 6. According to the presented results, in particular, the ML1 model has come to the fore, similar to the results obtained with the training set for the recall and specificity values, while the ML2 and ML3 models have been put forward for the precision value, and this situation is evident for the stain fault class. However, the average precision is the highest for the ML1 model. Dirtiness has the highest precision, recall and the specificity values for the ML1 model. To sum up, the Adaboost (i.e., ML1) model is the most suited for detecting these fault classes.

Table 6. Precision, Recall and Specificity results for each fault class obtained from the test dataset

Fault Class	Test dataset								
	Precision (%)			Recall (%)			Specificity (%)		
	ML1	ML2	ML3	ML1	ML2	ML3	ML1	ML2	ML3
Bumps	88.608	89.474	95.522	87.5	85	80	94.611	95.21	98.204
Dirtiness	100	88.889	72.727	88.889	88.889	88.889	100	99.58	98.739
K-Scratch	97.468	98.701	93.902	98.718	97.436	98.718	98.817	99.408	97.041
Pastry	79.310	77.143	73.529	74.194	87.097	80.645	97.222	96.296	95.833
Stains	93.750	100	100	100	100	100	99.569	100	100
Z-Scratch	94.444	94.286	89.474	100	97.059	100	99.061	99.061	98.122

ML1: Adaboost, **ML2**: Bagging, **ML3**: Random Forest Ensemble models

In the current problem precision and accuracy are the key metrics to compare the models. Generally, the accuracy score is a measure of how often the model predicts correctly. The Adaboost model provides accuracy scores of 93.15% and 91.90% for training and testing, respectively. Similarly, the Bagging model provides accuracy scores of 91.83% and 91.90% for training and testing, respectively. In the same way, the Random Forest model provides accuracy scores of 90.93% and 90.28% for training and testing, respectively. According to the test set, which is the unseen data set, the accuracy scores of Adaboost and Bagging are the same. Since the precision of Adaboost is better than other models for both training and test sets, it is found to be better to utilize the Adaboost model in the classification of faults.

The multiclass confusion matrix for the steel plate failure classification derived using the Adaboost model for both the training and test datasets is illustrated in Figure 4. The results of the statistical metrics presented for the ML1 model given in Tables 5 and 6 were calculated using the values given in Figure 4. The results in Figure 4 are parallel to the results in Tables 5 and 6.

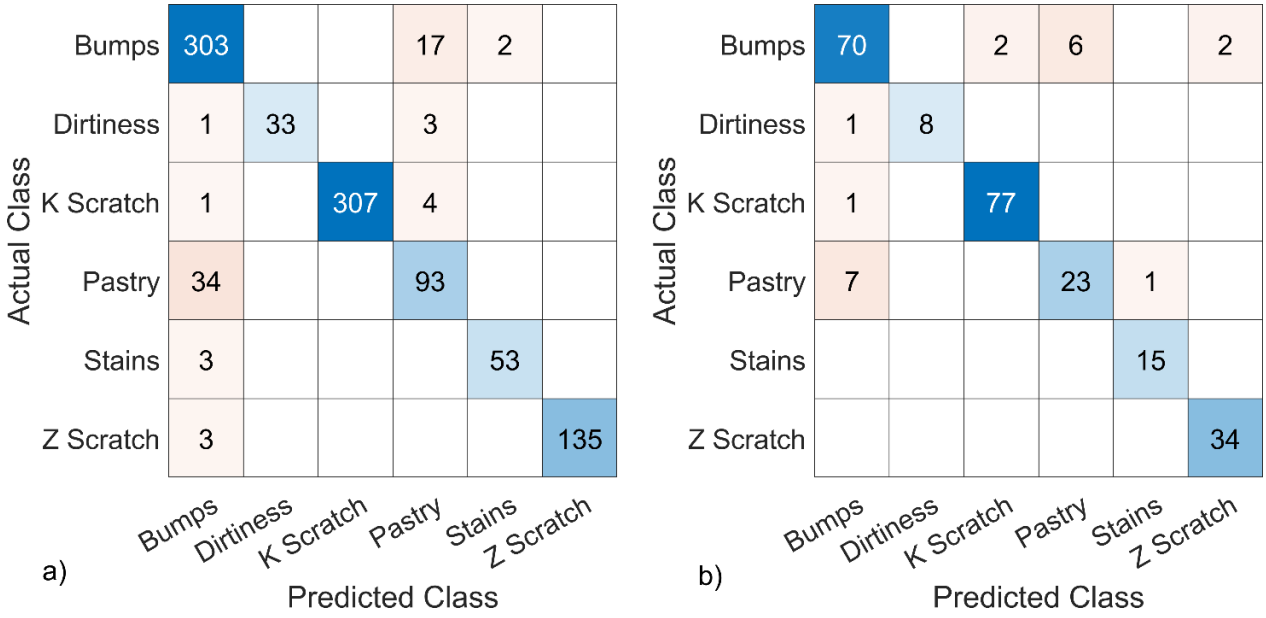

Figure 4. Multiclass confusion matrix for steel plate fault classification obtained from a) training dataset, b) test dataset

The performances of the developed models are compared with the related studies in the literature using the steel plate fault dataset in Table 7. The performances of the developed machine learning models were compared according to statistical metrics in terms of precision, recall, sensitivity and accuracy. However, in most of the studies, the majority of the aforesaid metrics are not given, instead only the accuracy score can be easily obtained. In some studies, the data set is also not divided into two as training and testing. Therefore, the values that cannot be found are marked as Not Given (NG).

Table 7. Comparison of the performance of the ML models with respect to related studies

References	ML Model	Training Dataset Metrics				Training Dataset Metrics			
		A (%)	P (%)	R (%)	S (%)	A (%)	P (%)	R (%)	S (%)
Fakhr and Elsayad, (2012)	DT	98.09	NG	NG	NG	97.25	NG	NG	NG
	NN	79.14	NG	NG	NG	74.79	NG	NG	NG
Tian et al., (2015)	GS-SVM	94.6	NG	NG	NG	77.7	NG	NG	NG
	GA-SVM	95.2	NG	NG	NG	77.2	NG	NG	NG
	PSO-SVM	88	NG	NG	NG	78.0	NG	NG	NG
Zhao et al., (2015)	NN	94.67	NG	NG	NG	88.05	NG	NG	NG
Chen, (2018)	DT	93.57	NG	NG	NG	85.43	NG	NG	NG
	RF	100	NG	NG	NG	90.29	NG	NG	NG
	Adaboost	100	NG	NG	NG	88.57	NG	NG	NG
	Bagging	96.30	NG	NG	NG	90.00	NG	NG	NG
Srivastava, (2019)	DT	NG	NG	NG	NG	76.04	NG	NG	NG
	RF	NG	NG	NG	NG	79.9	NG	NG	NG
	kNN	NG	NG	NG	NG	71.35	NG	NG	NG
	Adaboost	NG	NG	NG	NG	78.41	NG	NG	NG
	SVM	NG	NG	NG	NG	74.90	NG	NG	NG
Nkonyana et al., (2019)	SVM	NG	NG	NG	NG	73.6	NG	NG	NG
	NN	NG	NG	NG	NG	69.6	NG	NG	NG
	RF	NG	NG	NG	NG	77.8	NG	NG	NG
Kharal, (2020)	LR	89.13	NG	NG	NG	70.56	NG	NG	NG
	RF	94.18	NG	NG	NG	91.25	NG	NG	NG
Gamal et al., (2021)	DT	NG	NG	NG	NG	91.14	91.29	NG	91.14
	RF	NG	NG	NG	NG	91.29	91.86	NG	91.29
	SVM	NG	NG	NG	NG	86.00	74.57	NG	86.00
	LR	NG	NG	NG	NG	88.29	86.71	NG	88.29
Tasar (2022)	kNN	91.78	86.00	87.15	87.05	NG	NG	NG	NG
	SVM	93.28	88.02	89.28	88.44	NG	NG	NG	NG
	DT	93.01	88.47	89.30	88.82	NG	NG	NG	NG
	RF	95.47	92.09	93.17	92.37	NG	NG	NG	NG
	NN	96.98	94.75	95.54	94.87	NG	NG	NG	NG
This study	Adaboost	93.15	93.95	91.23	98.46	91.90	92.26	91.55	98.21
	Bagging	91.83	91.81	89.08	98.18	91.90	91.42	92.58	98.26
	RF	90.93	88.59	90.73	98.07	90.28	87.53	91.38	97.99

5. CONCLUSION

This article presents a comparative study of ensemble machine learning models such as Adaboost, Bagging, and Random Forest for multiclass fault classification of steel plates. The classification performance of the proposed models is presented using statistical metrics precision, recall, sensitivity and accuracy. The dataset provided in this study contains 27 independent features and 7 different failure classes. The dimension reduction method PCA was applied before training machine learning models. It is found that removing 12 insignificant features improves the performance of models. In addition, the other failure class, which was one of the failure classes, was excluded from the dataset since it was not a specific kind of failure class, but it was a combination of several classes. In conclusion, it can be said that the innovations and main contributions of this article are two-folds. First, this work discusses existing ML models using a steel plate failure dataset, and then, this work provides a comprehensive modelling approach to construct a computationally cheaper and more accurate ML model by applying the feature selection and hyperparameter optimization steps. Thus, it is thought that the developed model can more accurately adapt to sudden changes during production by determining the best model based on new information. Furthermore, in contrast to the previous studies, this article provides a systematic modelling approach that includes the normalization step in the data preparation stage to reduce the effects of outliers, the feature selection step in the dimension reduction stage to develop an ML model with a short computation time, and hyperparameter optimization step in the model development stage to increase the accuracy of the ML model. As the basic conclusion, this work determines that the Adaboost model is the most suitable model for fault detection problems. It can achieve high accuracy compared with the Bagging and Random Forest models. The Adaboost model achieves accuracy scores of 93.15% and 91.90% for training and test datasets, respectively. The second-best model is the Bagging model, which shows accuracy scores of 91.83% and 91.90% for training and test datasets, respectively. In conclusion, this study shows that ensemble machine learning models have the ability to accurately classify faults that occur during manufacturing, and they can replace manual inspection in decision support systems despite potential problems in practice.

7. CONFLICT OF INTEREST

Authors approve that to the best of their knowledge, there is not any conflict of interest or common interest with an institution/organization or a person that may affect the review process of the paper.

8. AUTHOR CONTRIBUTION

Erkan Caner OZKAT has the full responsibility of the paper about determining the concept of the research, data collection, data analysis and interpretation of the results, preparation of the manuscript and critical analysis of the intellectual content with the final approval.

9. NOMENCLATURE

A	accuracy score	NB	naive Bayes
DT	decision tree	NN	neural network
FN	false negative	P	precision
FP	false positive	PCA	principal component analysis
GA	genetic algorithm	PSO	particle swarm optimization
GS	grid search	R	recall
i	the index of actual class	RF	random forest
j	the index of predicted class	RMSE	root mean square error
kNN	k-nearest neighbour	S	specificity
LOF	local outlier factor	SPC	statistical processing control
LR	logistic regression	SVM	support vector machine
ML	machine learning	TN	true negative
N	number of sample	TP	true positive

10. REFERENCES

Alkan B., Bullock, S., Assessing operational complexity of manufacturing systems based on algorithmic complexity of key performance indicator time-series. Journal of the Operational Research Society 72(10), 2241-2255, 2021.

Backman J., Kyllönen V., Helaakoski H., Methods and tools of improving steel manufacturing processes: Current state and future methods. IFAC-PapersOnLine 52(13), 1174-1179, 2019.

Bektas O., Jones J. A., Sankararaman S., Roychoudhury I., Goebel K., Reconstructing secondary test database from PHM08 challenge data set. Data in Brief 21, 2464-2469, 2018.

Bektas O., Jones J. A., Sankararaman S., Roychoudhury I., Goebel K., A neural network framework for similarity-based prognostics. MethodsX 6, 383-390, 2019.

Buscema M., Terzi S., Tastle W., A new meta-classifier. Annual Meeting of the North American Fuzzy Information Processing Society (NAFIPS), Toronto, ON, Canada, July 12-14, 2010, pp: 1-7.

Ceryan N., Ozkat E. C., Korkmaz Can N., Ceryan S., Machine learning models to estimate the elastic modulus of weathered magmatic rocks. Environmental Earth Sciences 80(12), 1-24, 2021.

Chen J., The Application of tree-based ML algorithm in steel plates faults identification. Journal of Applied and Physical Sciences 4(2), 47-54, 2018.

Fakhr M., Elsayad A. M., Steel plates faults diagnosis with data mining models. Journal of Computer Science 8(4), 506-514, 2012.

Gamal M., Donkol A., Shaban A., Costantino F., Di G., Patriarca R., Anomalies detection in smart manufacturing using machine learning and deep learning algorithms. In Proceedings of the International Conference on Industrial Engineering and Operations Management, Rome, Italy, August 2-5, 2021, pp: 1611-1622.

Gao Y., Gao L., Li X., Yan X., A semi-supervised convolutional neural network-based method for steel surface defect recognition. Robotics and Computer-Integrated Manufacturing 61, 101825, 2020.

Kaggle, A., Faulty Steel Plates, Research Center of Sciences of Communication, https://www.kaggle.com/datasets/uciml/faulty-steel-plates, (Retrieved August 8, 2022), 2017.

Kahveci S., Alkan B., Musab H, A., Ahmad B., Harrison R., An end-to-end big data analytics platform for IoT-enabled smart factories: A case study of battery module assembly system for electric vehicles. Journal of Manufacturing Systems 63, 214-223, 2022.

Kazemi M. A. A., Hajian S., Kiani N., Quality Control and Classification of Steel Plates Faults Using Data Mining. Applied Mathematics Information Sciences Letters 6(2), 59-67, 2018.

Kharal A., Explainable artificial intelligence based fault diagnosis and insight harvesting for steel plates manufacturing. arXiv preprint arXiv:2008.04448, 2020.

Kurt R., Industry 4.0 in terms of industrial relations and its impacts on labour life. Procedia computer science 158, 590-601, 2019.

Lennox B., Montague G., Marjanovic O., Detection of faults in Batch Processes: Application to an industrial fermentation and a steel making process. Water Science and Technology, 2000.

Liu Y., Gao H., Guo L., Qin A., Cai C., You Z., A data-flow oriented deep ensemble learning method for real-time surface defect inspection. IEEE Transactions on Instrumentation and Measurement 69(7), 4681-4691, 2019.

Markoulidakis I., Rallis I., Georgoulas I., Kopsiaftis G., Doulamis A., Doulamis N., Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem. Technologies 9(4), 81, 2021.

Nkonyana T., Sun Y., Twala B., Dogo E., Performance evaluation of data mining techniques in steel manufacturing industry. Procedia Manufacturing 35, 623-628, 2019.

Ozkat E. C., Franciosa P., Ceglarek D., Laser dimpling process parameters selection and optimization using surrogate-driven process capability space. Optics & Laser Technology 93, 149-164, 2017a.

Ozkat E. C., Franciosa P., Ceglarek D., Development of decoupled multi-physics simulation for laser lap welding considering part-to-part gap. Journal of Laser Applications 29(2), 022423, 2017b.

Özkat E. C., Makine Öğrenmesi Metodolojisi Kullanılarak Yüksek Hızlı Rulmanlarda Sağlık Göstergesinin Belirlenmesi. Avrupa Bilim ve Teknoloji Dergisi (22), 176-183, 2021.

Pham T. A., Tran V. Q., Developing random forest hybridization models for estimating the axial bearing capacity of pile. Plos one, 17(3), e0265747, 2022.

Simić D., Svirčević V., Simić S., An approach of steel plates fault diagnosis in multiple classes decision making. In International Conference on Hybrid Artificial Intelligence Systems, Salamanca, Spain, June 11-13, 2014, pp: 86-97.

Srivastava A. K., Comparison analysis of machine learning algorithms for steel plate fault detection. International Research Journal of Engineering and Technology 6(4), 1231-1234, 2019.

Tasar B., Comparison Analysis of Machine Learning Algorithms for Steel Plate Fault Detection. Düzce Üniversitesi Bilim ve Teknoloji Dergisi 10(3), 1578-1588, 2022.

Tian Y., Fu M., Wu F. Steel plates fault diagnosis on the basis of support vector machines. Neurocomputing 151, 296-303, 2015.

Widodo A., Yang B. S., Support vector machine in machine condition monitoring and fault diagnosis. Mechanical systems and signal processing 21(6), 2560-2574, 2007.

Xiong J., Pang Q., Cheng W., Wang N., Yong Z., Reservoir risk modelling using a hybrid approach based on the feature selection technique and ensemble methods. Geocarto International 37(11), 3312-3336, 2022.

Xu L. D., Xu E. L., Li L. Industry 4.0: state of the art and future trends. International journal of production research 56(8), 2941-2962, 2018.

Yang K., Yu Z., Chen C. P., Cao W., Wong H. S., You J., Han G., Progressive hybrid classifier ensemble for imbalanced data. IEEE Transactions on Systems, Man, and Cybernetics: Systems 52(4), 2464-2478, 2021.

Zhang X., Kano M., Tani M., Mori J., Ise J., Harada K., Prediction and causal analysis of defects in steel products: Handling nonnegative and highly overdispersed count data. Control Engineering Practice 95, 104258, 2020.

Zhao Z., Yang J., Lu W., Wang X., Application of local outlier factor method and back-propagation neural network for steel plates fault diagnosis. In The 27th Chinese Control and Decision Conference (2015 CCDC), Qingdao, China, 23-25 May, 2015, pp: 2416-2421.