Machine Learning-based Prediction of Surplus Material in Intelligent Production Processes

Yingjun Ji*, Ziyan Zhao*, Shixin Liu, and Xiaoyue Yong

Abstract—Industry 4.0 technologies have driven traditional manufacturing to intelligent manufacturing. With massive production data, machine-learning-based methods have been widely used to predict and control a production process intelligently. They can help manufacturers achieve efficiency improvement and cost reduction. In this work, we present a data-driven method to predict surplus materials in a cold rolling process in intelligent steel production systems. A cold-rolled steel coil is a common kind of steel products that is easy to generate surplus materials during its production. Avoiding or reducing the generation of surplus materials is desired for steel enterprises since it seriously affects their profits. However, its complex production processes make it difficult to identify the causes of surplus materials. The issue of predicting surplus material has not yet been well-addressed. In this work, a surplus material prediction problem is first proposed and solved based on a combination of statistical analysis and data-driven methods. We aim to find the key parameters that cause the generation of surplus material and predict whether surplus materials would be generated during a production process. Based on an industrial dataset, several machine learning methods are developed to build a prediction model that is able to meet actual requirements of steel coil production processes. The experimental results show that, among them, extreme gradient boosting and logistic regression methods are highly reliable with the best performance. In addition, an explicit expression obtained by logistic regression can provide practitioners with excellent guidance in their practical applications.

***Index Terms*—surplus material prediction; industrial data analysis; feature selection; data-driven methods; intelligent steel production system**

I. Introduction

Steel manufacturing is a traditional and typical process industry [1, 2]. Since the development of industrial informatization in steel manufacturing enterprises, they have accumulated industrial big data. Driven by the concept of Industry 4.0, steel manufacturing enterprises are transforming from informatization to intellectualization. How to extract effective information from industrial big data to guide actual production processes is an important challenge. This work focuses on a surplus material prediction problem arising from practical steel coil production processes and presents machine learning-based methods to handle it.

A cold-rolled coil is a common type of products in steel enterprises. For easy storage and transportation, steel sheets and strips are generally produced and rolled into coils. The production of cold-rolled coils is arranged according to their order weight. Typically, the weight requirement of a customer order is in a certain range, e.g., 8-15 tons. If a cold-rolled coil product cannot meet the weight requirement, e.g., weight unmatched or quality dissatisfactory, it would be treated as a surplus material. If a steel coil product is judged to be a surplus material, it can only be sold as spot goods whose price drops by 15% to 20% per ton. Moreover, if the coil cannot be sold even under such conditions, it has to be returned to a furnace as a raw material for steelmaking. This obviously increases production and inventory costs, reduces the profit rate of a company, and consumes additional energy. Thus, avoiding or reducing the generation of surplus materials during a production process of cold-rolled coils is an important issue that steel enterprises need to address.

Many existing studies focus on predicting product quality and mechanical properties of steel coils in a cold rolling process. Nam et al. [3] present an on-line model based on finite element method to achieve a roll force profile prediction in a cold rolling process. Sanz-Garcia et al. [4] propose a GA-SVR approach for predicting temperature in a continuous annealing furnace. Lalam et al. [5] use artificial neural networks (ANN) to predict yield strength and ultimate tensile strength of coils in a galvanizing process. Lu et al. [6] use data-driven methods to analyze and predict mill vibration. The presence of surplus materials may be affected by the quality of semi-finished products in several sub-processes of a cold rolling process. However, no studies have been conducted on the prediction of surplus material generation in literature to our best knowledge.

The new problem concerned in this work is named as a surplus material prediction problem (SMPP). It is to predict whether a steel coil is a surplus material. It means that an SMPP is treated as a binary classification problem. The key to solving this issue lies in finding the features related to the generation of surplus materials and controlling their related variables. However, hundreds of features are involved in the production process of cold-rolled coils, and their types are

Manuscript received April 3, 2025; revised April 14, 2025; accepted April 18, 2025. This article was recommended for publication by Associate Editor Liang Qi upon evaluation of the reviewers' comments. The work is financially supported in part by the Liaoning Revitalization Talents Program under Grant XLYC2002041, in part by Science and Technology Project of Liaoning Province under Grant 2024-BSLH-205, and in part by Youth Program of the Liaoning Provincial Department of Education under Grant JYTQN2023366.

Y. Ji is with the Faculty of Information, Liaoning University, Shenyang 110036, P. R. China (e-mail: jiyingjun@lnu.edu.cn).

Z. Zhao, and S. Liu are with the State Key Laboratory of Synthetical Automation for Process Industries, and the College of Information Science and Engineering, Northeastern University, Shenyang 110819, P. R. China (e-mail: zhaoziyan@mail.neu.edu.cn, and sxliu@mail.neu.edu.cn).

X. Yong is with Shanghai Baosight Software Co., Ltd, Shanghai 201203, P. R. China (e-mail: yongxiaoyue@baosight.com)

* Yingjun Ji and Ziyan Zhao contributed to the work equally and should be regarded as co-first authors. *Corresponding Author: Yingjun Ji*

multiple. Although production data for these features can be collected from a steel enterprise, it is still difficult to process them and analyze their impacts on the generation of surplus materials. Besides, the actual production data of steel coils are normally incomplete and irregular, which increases the difficulty in extracting valid information from them.

In recent studies, data-driven machine learning methods have drawn increasing attention for solving the problems that cannot be modeled by explicit mathematical equations. Depending on the characteristics of problems, they can be classified as supervised and unsupervised learning ones [7]. The concerned SMPP as a binary classification problem can be addressed with the former. Extreme gradient boosting (XGBoost) [8] is a well-known supervised learning method that constructs a powerful classifier by stepwise adding decision trees. Besides, it can rank the importance of the features in its prediction model. Therefore, it is applicable to solve some classification problems as well as for feature selection. Yan *et al.* [9] propose a modified XGBoost method to predict the fatigue strength of steel. Katirci *et al.* [10] use XGBoost to predict the ZnNi alloy coating thickness and Ni % amount in the coating. Logistic regression (LR) [11] is an effective choice for solving binary classification problems, which aims to model the probability that a sample belongs to a particular category. Alatarvas *et al*. [12] use an LR-based classifier to predict inclusion state in molten steel. Besides, some other machine learning methods, such as support vector machines (SVM) [13, 14], naive Bayes (NB) [14], quadratic discriminant analysis (QDA) [15], k-nearest neighbor algorithm (KNN) [10], ANN [16], are also commonly used to solve prediction problems in different industrial scenarios.

Although data-driven methods have been widely used in some prediction problems of steel production, few studies considering the whole production process of cold-rolled steel coils have been reported, especially on surplus material prediction. Previous studies on surplus material focus on production scheduling and optimization, which do not predict its occurrence. In this work, we propose a three-stage data-driven approach to identify the core features and predict the generation of surplus materials depending on their actual production data. This work aims to make the below contributions:

1) A novel surplus material prediction problem arising from a practical steel coil manufacturing process is first studied.

2) A three-stage approach is proposed to solve an SMPP by pre-processing industrial data, conducting feature selection with both statistical analysis and machine learning methods, and predicting if surplus materials are generated.

3) The experiments conducted on an actual industrial dataset show the effectiveness of the proposed three-stage approach. The performance of some commonly used machine learning methods on solving SMPPs are compared and the ones with superior prediction accuracy and interpretability are recommend to practitioners.

The remainder of the work is organized as follows. An SMPP of cold-rolled coils in steel production systems is presented in Section II. The proposed three-stage approach for solving it is described in Sections III-V. Specifically, data cleaning and feature selection methods are described in Sections III and IV, respectively. XGBoost, LR, and some competitive prediction methods are introduced in Section V. The experimental results based on actual production data are shown in Section VI. Conclusions and future work are discussed in Section VII.

II. What is the Problem?

The general layout of a cold rolling process is shown in Fig. 1. The incoming material is generally transformed into a finished cold-rolled coil product after multiple complicated sub-processes, such as pickling, cold rolling, annealing, galvanizing, finishing, cutting, and coiling. Each sub-process may impact the weight of a finished product that is the key to deciding whether surplus material is generated. The quality of its semi-finished products is also an important factor to surplus material. According to our investigations in steel enterprises, three main reasons for surplus material generation are summarized:

1) Some varieties of cold-rolled coils, e.g., coated strips, need to go through an additional coating process before coiling. In this process, if the coating layer is not uniform or the coating material has poor quality, the produced coils may suffer from weight and quality issues. Thus, they could not meet the specification requirement of customer orders and have to be regarded as surplus materials.

2) Finishing mill is generally equipped with a detecting machine that can detect defects on products at its exit. The defect determination results of different detecting machines have certain differences, which leads to different sizes of excised defects. If too much (or too little) portion of a coil is excised, its remaining weight cannot meet a customer's order and has to be classified as surplus material.

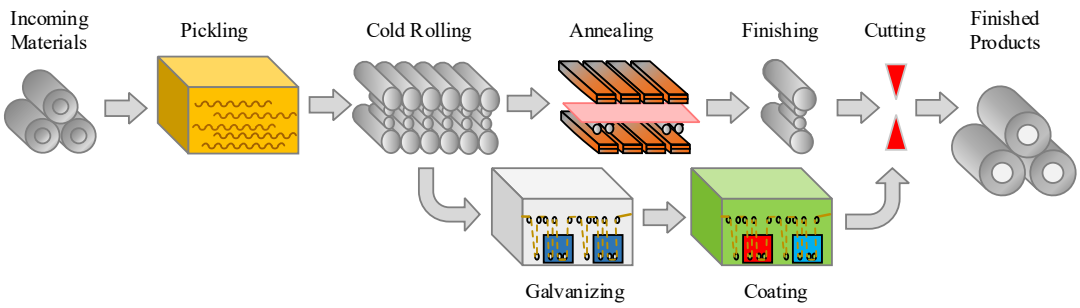

Fig.1. The layout of a cold rolling production line.

3) The final cutting process is also an important factor. In some cutting units, their cutting model is relatively simple. Instead of cutting coils according to the defects on them, the separation is simply performed according to some simple rules such as proportion. After cutting the defective part of a cold-rolled coil, the residual part becomes surplus material.

According to the above description, a cold rolling process consists of a series of sub-processes. Each of them includes many process parameters (features). These features have different levels of influence on the generation of surplus material. We aim to select those essential ones and identify their impacts. Based on actual production data, the concerned SMPP is to predict if a surplus coil would appear under a selected set of features. We regard SMPP as a binary classification problem, i.e., the case that a surplus coil appears is true/positive, while the one that no surplus coil appears is false/negative. For convenience, we call the output binary feature (i.e., surplus coil appears or not) as the response.

The raw data collected from a production line are often incomplete and irregular. They cannot be directly used to build a prediction model. It is essential to preprocess them before their use. Otherwise, the performance of the prediction model would be poor and overfitting may occur. Such an unreliable model would not be acceptable in practical applications. When constructing a prediction model, it is also crucial to determine the number of features to be used. The number of features affects the effectiveness of the model and its applicability in industrial scenarios [17]. This work proposes a three-stage approach to deal with an SMPP, as shown in Fig. 2. At the first stage, data cleaning is executed to screen and condition raw data. Then, feature selection at the second stage is to identify core features that have important influences on the generation of surplus materials. According to different data types and characteristics, we use statistical analysis-based and ML-based methods, respectively. At the third stage, different methods are applied to build models for surplus material prediction based on the selected features.

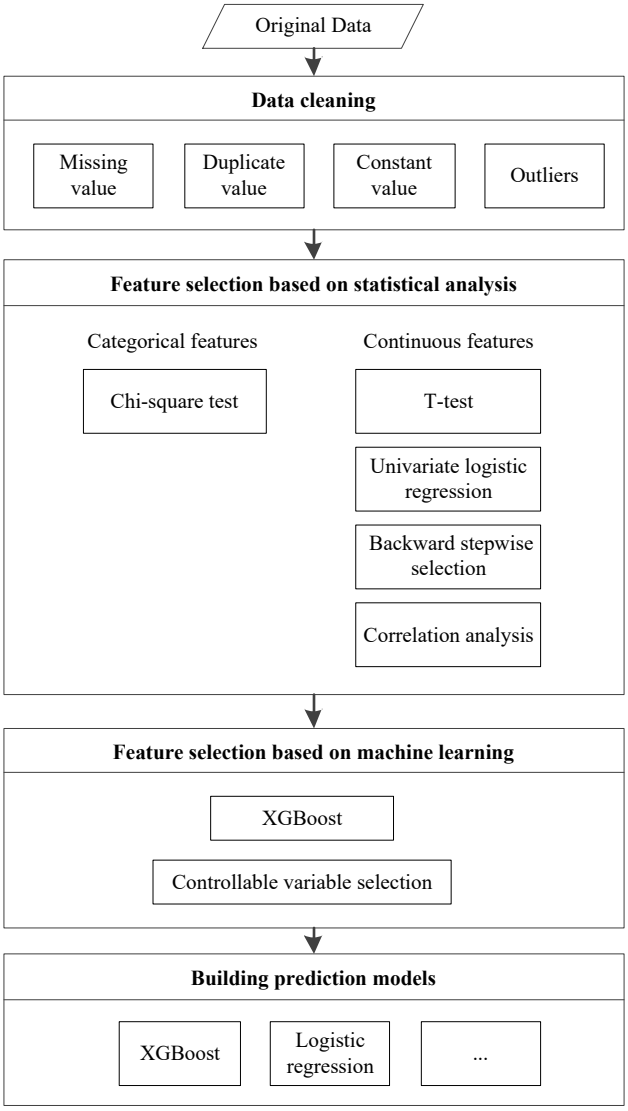

Fig. 2. Flowchart for solving the concerned SMPP.

III. Data Cleaning

The dataset used in this work is collected from a cold-rolled coil production line of a steel enterprise located in southeastern China. Because of the incompleteness and irregularity of the raw data, cleaning is necessary to make it satisfy the requirements of feature selection and surplus material prediction. Considering the requirements for available data, four types of irregular data, i.e., missing, duplicate, and constant values as well as outliers are identified. Based on their characteristics, we analyze their causes in steel coil production and introduce corresponding treatments.

1) Missing value

In the production of steel coils, two scenarios may result in missing values: one is the loss of production records. For example, during a rolling process, a data acquisition module fails to collect rolling data at a certain moment due to sudden abnormality. A loss of rolling data then occurs at this point. The other represents that the operating state of a unit is not available. For example, a certain variety of steel coil does not need to go through a flying shear in a cold rolling process. Then, the corresponding record of the coil in the feature (i.e., the operating state of the flying shear) is missing, which means that the coil is not cut on the flying shear. We judge the absence of a feature by column. If the missing value of a feature in a column exceeds a preset threshold θ_1, the entire column is deleted.

2) Duplicate value

Duplicate values mean that there are two or more features in a raw dataset whose values are the same on all samples. In this case, duplicate ones have the same effect on the response (i.e., whether a steel coil is a surplus material). Two scenarios may result in the generation of duplicate values. One is that two features have different meanings, but they take exactly equal values. For example, p_1 and p_2 are two sequential sub-processes in a cold rolling process, and p_1 precedes p_2. The exit thickness of p_1 is equal to the entrance thickness of p_2. Although they present different meanings, their values are equal and have the same effect on a prediction of surplus material. Therefore, they are regarded as duplicate features. The other scenario is that the raw dataset is a combination of datasets from multiple sub-processes of a cold rolling process. A certain feature (e.g., identity information of steel coils) may

be contained in several datasets from these sub-processes. Merging these datasets leads to the duplicate features. For duplicated features, we retain only one of them.

3) Constant value

In the production of steel coils, some units operate in one setting mode for most of the time, and some even have only one constant operating mode. Hence, their corresponding features produce mostly constant values. Based on the percentage of repeated values of each feature in actual production data, we can obtain a distribution histogram of the features under different percentages, as shown in Fig. 3. Note that 6 features whose percentages are greater than 100% on the horizontal axis indicate that their percentages are equal to 100%, i.e., their entire samples take a constant value. In analyzing the relationship between different features and the response, if the values of a feature are the same in most samples (i.e., the same value count exceeds a certain threshold), we consider that the feature has little or no effect on the response. In this work, we represent such threshold as θ_2. If more than θ_2 samples of a feature have the same value, this feature is deleted from the dataset.

4) Outliers

An outlier is a sample whose one or more features deviate markedly from others' [18]. In actual production, a part of the raw data is manually recorded by workers, which has a high probability of producing incorrect ones. Besides, anomalies of the data acquisition system such as an overflow of a recorded time data can also result in an outlier. The presence of outliers can negatively affect the accuracy of prediction model. Thus, processing them is an essential part of data cleaning. In this work, a box plot is adopted to identify and handle outliers. Fig. 4 is a schematic diagram of a box plot where all values of a feature are sorted from the smallest to largest before identifying outliers. The number at the 25% position after sorting is selected as the lower quartile called Q_1, and the number at the 75% position is the upper quartile called Q_3. The difference between Q_3 and Q_1 is defined as interquartile range (I_{QR}). If a value is greater than $\text{Maximum} = Q_3 + 1.5I_{QR}$ (resp. less than $\text{Minimum} = Q_1 - 1.5I_{QR}$), it is treated as an outlier and modified to $Q_3 + 3I_{QR}$ (resp. $Q_1 - 3I_{QR}$).

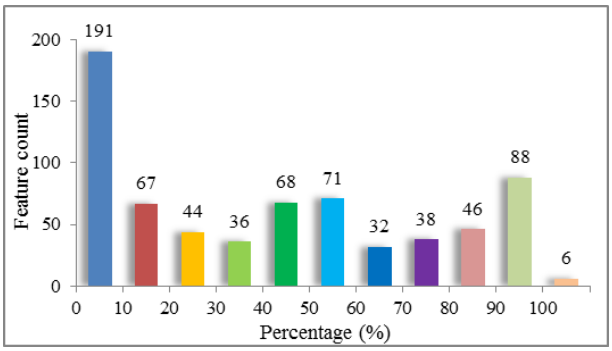

Fig. 3. Histogram of constant values distribution.

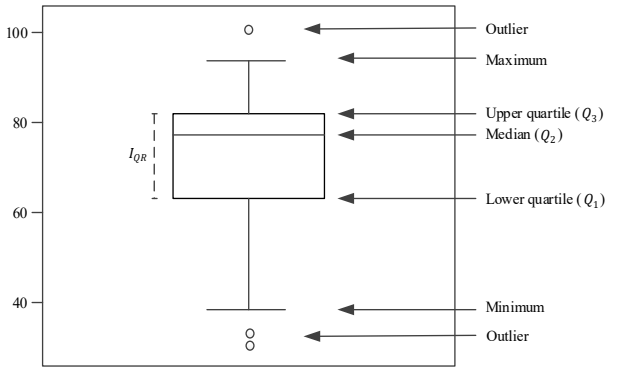

Fig. 4. The diagram of a box plot for identifying and correcting outliers.

IV. Feature Selection

After data cleaning, the identified errors in the dataset can be effectively corrected or removed. However, the dataset contains a large number of redundant highly-correlated features whose use may degrade the accuracy of a prediction model. Thus, feature selection is an essential stage before constructing a prediction model. In this section, feature selection methods based on statistical analysis and machine learning are adopted to select core features.

A. Statistical Analysis

Based on statistical analysis methods, each feature is measured for its correlation with the response, and those having little or no correlation are eliminated. The statistical analysis methods adopted in this work are based on hypothesis testing. First, it is assumed that the tested feature and the response are independent. A probability called *p*-value is then calculated to measure their correlation. If it is less than a threshold, we reject the null hypothesis. In other words, we conclude that a relationship exists between the tested feature and the response. Thus, the tested feature should be kept. If the *p*-value is not less than a threshold, we cannot reject the null hypothesis. In other words, the tested feature and the response are not correlated. Thus, the tested feature is removed. We denote the threshold of a *p*-value for rejecting the null hypothesis by θ_3.

There are two types of features contained in the dataset, i.e., categorical and continuous ones. Depending on different types, we adopt corresponding methods to implement feature selection, as shown in Fig. 2. Chi-square test is used to filter categorical features. Its statistic χ^2 denotes the degree of deviation between actual observations and theoretical estimates of samples, which is calculated as $\chi^2 = \sum \frac{(A-A^0)^2}{A^0}$. A and A^0 represent actual observations and theoretical estimates, respectively. Then, a *p*-value corresponding to χ^2 is able to determine whether a categorical feature needs to be removed or not.

For continuous features, we first use *t*-test to evaluate the correlation between them and the response. In order to further filter irrelevant features, we adopt univariate logistic regression analysis and backward stepwise selection. We also use *p*-value as their metric for selecting features.

Backward stepwise selection is a greedy feature selection method with high computational efficiency. It begins with a full model M_p containing all p features in a dataset, and then iteratively removes the least useful one at a time, until none of the features is in the model (noted as M_0). In each iteration, all the features are traversed, and the least useful one is removed based on the minimum residual sum of squares (RSS) principle,

i.e., $R_{SS} = \sum_{i=1}^{n}(y_i - \tilde{y}_i)^2$, where n, y_i, and $\tilde{y}_i$ represent the number of samples, observed and predicted values, respectively. Finally, we use Akaike Information Criterion (A_{IC}) to select the best model M^* among M_0, M_1,…, and M_p. It is computed as $A_{IC} = \frac{1}{n\tilde{\sigma}^2}(R_{SS} + 2d\tilde{\sigma}^2)$, where d and $\tilde{\sigma}^2$ denote the number of features and an estimate of the variance between the observed and predicted values, respectively. The subset of the features constituting M^* is denoted as F^*. The procedure of the backward stepwise selection method is given in Algorithm 1.

After the above operation, the continuous features correlated with the response are retained. However, there may be collinearity among them, i.e., certain features have a similar effect on the prediction. In order to eliminate the collinearity between two features, a correlation analysis based on Pearson correlation coefficient is performed. The Pearson correlation coefficient between two features is calculated as:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{1}$$

where n denotes the number of samples, x and y denote two features, $\bar{x}$ and $\bar{y}$ are their mean values. If the absolute value of r_{xy} between x and y is greater than a given threshold θ_4, they are considered to be correlated. Then, one of them is randomly removed from the feature set.

Algorithm 1: Backward stepwise selection

Input: p features to be selected and the response for modeling

Output: the features subset F^* constituting model M^*

```
Let all p features constitute a set F_p, and construct a model M_p that contains all these features;
for i = p to 1
    for j = 1 to i
        Remove the jth feature from F_i, and the resulting set is denoted as f_{i/j};
        Construct a model m_{i/j} using f_{i/j} and the response;
        Calculate its R_SS by formula (2);
    Select the one with the minimum R_SS among m_{i/j}, which denotes as M_{i-1}, and its corresponding features set is F_{i-1};
for k = 0 to p
    Calculate the A_IC value of M_k using formula (3);
    The model with minimum A_IC is selected and denoted as M*, and its corresponding features set is denoted as F*;
return F*;
```

B. Machine Learning

After feature selection using a series of statistical analysis-based methods, XGBoost [8] is employed to further filter unimportant features. It is a kind of supervised learning algorithms based on gradient tree boosting algorithm. XGBoost has a great attribute that it can rank the importance of features according to their information gains. We exploit it to realize feature selection. Considering both prediction accuracy and overfitting, XGBoost adopts the following objective function:

$$\begin{aligned} z^{(t)} &= \sum_{i=1}^{n} l\left(y_i, \tilde{y}_i^{(t)}\right) + \sum_{i=1}^{t}\Omega(f_i) \\ &= \sum_{i=1}^{n} l\left(y_i, \tilde{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) + \sum_{i=1}^{t-1}\Omega(f_i) \end{aligned} \tag{2}$$

where n is the number of samples, $l\left(y_i, \tilde{y}_i^{(t)}\right)$ denotes a loss function, $\tilde{y}_i^{(t)}$ is the predicted value of sample i at the t-th iteration, $\Omega(\cdot)$ denotes a regularization term, f_t is the tree to be trained at the t-th iteration. In order to evaluate the loss function effectively, the second-order Taylor expansion is used to approximate the original objective function in (5):

$$\begin{aligned} z^{(t)} = \sum_{i=1}^{n}[l\left(y_i, \tilde{y}_i{}^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2}h_i {f_t}^2(x_i)] \\ + \Omega(f_t) + \sum_{i=1}^{t-1}\Omega(f_i) \end{aligned} \tag{3}$$

where the first and second-order gradient of the loss function are respectively defined as $g_i = \partial_{\tilde{y}_i^{(t-1)}} l\left(y_i, \tilde{y}_i^{(t-1)}\right)$ and $h_i = \partial^2_{\tilde{y}_i^{(t-1)}} l\left(y_i, \tilde{y}_i^{(t-1)}\right)$.

The regularization term $\Omega(f_t)$ is determined by the number of leaf nodes Γ and the L2 norm of the weight of each leaf node ${\omega_j}^2$, i.e., $\Omega(f_t) = \gamma\Gamma + \frac{1}{2}\lambda\sum_{j=1}^{\Gamma}{\omega_j}^2$. Its usage can prevent the decision tree from splitting out too many nodes to result in overfitting.

An XGBoost model can output an importance order of the features according to their information gains. The top 20 features with the largest information gains are selected. However, it should be noted that some of these features are not controllable, i.e., their values cannot be tuned during a production process, such as material index. Guided by a prediction model, our goal is to reduce the generation of surplus material by controlling key features in the production process of cold-rolled coils. Thus, we only retain the controllable ones out of the selected 20 features to build an interpretable prediction model.

V. Intelligent Surplus Material Prediction

A. Extreme Gradient Boosting (XGBoost) Method

As a decision tree-based classification method, XGBoost can naturally be used for the second time to build a classifier after feature selection. At its training stage, it stepwise adds a decision tree to its prediction model to improve the prediction performance. After training, the structure of the trained model and the corresponding weights of trees in it are determined. The final prediction result of an XGBoost model is the sum of each decision tree's weight in it, which is calculated via (6).

Before constructing an XGBoost classifier, some key parameters for a boosting tree model should be determined, such as the learning rate η, the minimum loss reduction

required for a partition on a leaf node of tree ε, the maximum depth of a tree α, the minimum sum of sample weights needed in a child β, the maximum delta step of each leaf output δ, the subsample ratio of samples ζ, and the subsample ratio of features μ when constructing each tree.

B. Logistic Regression (LR) Method

Although XGBoost usually has high prediction accuracy, it is a black-box algorithm that only provides prediction results based on given samples and features. Its unknown functional relationship cannot enable practitioners to adjust strategies to reduce the generation of surplus materials in practical applications. To overcome this drawback, LR [11] introduced in this section aims to build a prediction model with a recognizable regression function. It is a popular choice for classification because of its simplistic formulation and interpretable model structure [12]. Its explicit expression can clearly show the quantitative relationship between input features and response. In LR, the probability of a steel coil to be judged as a surplus material is calculated as follows:

$$q(x) = \omega^{\mathrm{T}} x + b \tag{4}$$

$$g(z) = \frac{1}{1 + e^{-z}} \tag{5}$$

$$h_\omega(x) = g\big(q(x)\big) = \frac{1}{1 + e^{-\omega^{\mathrm{T}} x - b}} \tag{6}$$

$$Y = \begin{cases} 1 & h_\omega(x) \geq \theta_L \\ 0 & h_\omega(x) < \theta_L \end{cases} \tag{7}$$

where $q(x)$ is a linear regression model, ω and b are its weight and intercept coefficients to be trained. A logistic function $g(z)$ in (11) is chosen as an activation function. After the manipulation of (10) and (11), we obtain the probability $h_\omega(x)$ for prediction. The predicted value Y in (13) is a logical one. If $h_\omega(x)$ is greater than a given threshold θ_L, Y is considered as being true (coded as 1). Otherwise, it is considered as being false (coded as 0).

C. Methods for Comparison

SVM is an algorithm that maps data to points in a high-dimensional space [19]. It constructs a classifier by finding an appropriate hyperplane and support vectors. By selecting various kernel functions, it can handle the classification of different linear and nonlinear data. In this work, we select two kinds of kernel functions for modeling, i.e., polynomial and radial basis kernel functions. The general forms of their kernel functions are $K_p(x, y) = (\gamma x^{\mathrm{T}} y + c)^d$ and $K_r(x, y) = exp(\gamma \|x - y\|)^2$, where x and y are two samples, c is a constant, d denotes the power exponent of the polynomial, and γ is a penalty coefficient.

Two competitive methods based on Bayes' theorem, i.e., NB and QDA are adopted to solve SMPP. By calculating the posterior probability that the category of a given sample is 0 or 1, they can classify the sample into the category with the highest posterior probability value. NB assumes that features are independent. It performs classification by learning a joint probability distribution from input to output on a training set. QDA assumes that the samples of each class follow a Gaussian distribution, and each class has its unique covariance matrix. As with LR, when constructing classifiers using these methods, it is necessary to decide their threshold parameters. In this work, we denote them as θ_N and θ_Q, respectively.

KNN algorithm aims to find a neighborhood of k closest samples to a test sample in a training set and label it as the class that most of the samples in this neighborhood belong to. Two key parameters have great impact on its effectiveness, i.e., distance metric among samples and the number of nearest neighbors k. In this work, Euclidean distance is used to measure the k nearest neighbors.

ANN is a widely used type of algorithm because of its high accuracy in various classification problems. However, complicated network structures may cause overfitting and low generalization ability. In this work, we select a multilayer perceptron (MLP) as a representative of ANN, which is a fully connected feedforward network. It is among the most commonly used network structure [20]. A standard error backpropagation function is selected as the learning function of MLP in training. In it, we need to determine two parameters, i.e., the number of units in hidden layer s_M, and learning rate r_M.

VI. Experimental Results and Analysis

In this section, we first present the design details of our experiments. Then, our proposed three-stage approach is implemented on an efficient computational platform. Finally, experimental results are compared and analyzed.

A. Experimental Settings

The raw data are collected from a cold rolling production line of a steel plant. It contains three months of actual production data, with a total of 9670 samples with 832 input features and a response. Among them, 4487 samples are marked as true and 5183 as false.

TABLE I
A SUMMARY OF THE R LIBRARIES USED IN THIS WORK

Step	Method	Library
Statistical analysis-based feature selection	Chi-square test	stats
	T-test	stats
	Univariate logistic regression analysis	stats
	Backward stepwise selection	stats
	Correlation analysis	stats
Machine learning-based feature selection	XGBoost	xgboost
Building prediction models	XGBoost	xgboost
	LR	stats
	SVM	e1071
	QDA	MASS
	KNN	class
	NB	e1071
	MLP	RSNNS

TABLE II
THRESHOLD SETTINGS

Parameter	θ_1	θ_2	θ_3	θ_4
Value	97%	100%	0.05	0.8

After data cleaning, the experimental data are divided into training and test sets with a ratio of 7: 3. All the experiments are implemented in R programming language. The methods used in this work are implemented by using R libraries as shown in Table I. In addition, the threshold settings based on expert experience and some previous work [7] are summarized in Table II.

For other prediction methods introduced in Section V, we tune their optimal hyperparameters on a training set to get competitive competitors. Since each method contains many hyperparameters, we only concentrate on determining the values of the hyperparameters that have important impacts on the prediction results according to experience [6]. We adopt a grid search method to select a suitable value, and its results are summarized in Table III. The rest of the parameters are set to default values of corresponding functions given in R libraries. Note that four values of parameter m_s correspond to the same solution. Note that m_s may help in classification when classes are extremely imbalanced. In this work, our positive and negative samples are approximately balanced.

B. Performance Metrics

In order to evaluate the performance of the selected classifiers comprehensively, we use confusion matrix and five evaluation metrics, i.e., sensitivity, specificity, accuracy (ACC), F-measure, and AUC. Confusion matrix can describe the results of a binary classifier by using four values in Table IV. According to it, the other four metrics can be directly calculated as $\text{sensitivity} = \text{TP}/(\text{TP} + \text{FN})$, $\text{specificity} = \text{TN}/(\text{TN} + \text{FP})$, $\text{ACC} = (\text{TP} + \text{TN})/(\text{TP} + \text{TN} + \text{FP} + \text{FN})$, and $\text{F-measure} = 2 \times \text{TP}/(2 \times \text{TP} + \text{FP} + \text{FN})$. AUC (i.e., area under the curve of ROC) is also a valid numerical metric. It calculates the area under the ROC curve that intersects with the horizontal and vertical axes. All of these five metrics are the higher, the better.

C. Results and Comparative Analysis

According to the implementation of the proposed three-stage approach illustrated in Fig. 2, data cleaning and feature selection are first performed. The numbers of features after each step are shown in Table V. It can be seen that 680 of 832 features except for the response are retained after data cleaning. These 680 features are composed of 272 continuous and 408 categorical ones. Then statistical-analysis-based methods introduced in Section IV.A are performed to select these two types of features separately. First, 105 categorical features are removed by Chi-square test. Then, 2 and 80 redundant continuous features are eliminated by *t*-test and univariate logistic regression analysis, respectively. Next, backward stepwise selection is executed on the remaining continuous features. We find that with the number of features iteratively decreasing, the A_{IC} value keeps decreasing. When the A_{IC} value is no longer decreasing, the backward stepwise selection algorithm terminates and returns the remaining features. The selection result using the A_{IC} criterion is listed in Table VI. It can be seen that the model with 72 continuous features has the minimum A_{IC} value and is selected. After backward stepwise selection, 375 features are kept. Finally, 45 features are removed by correlation analysis. The relationship among continuous features before and after a correlation analysis can be visualized in the heatmaps, as shown in Fig. 5.

TABLE III
HYPERPARAMETERS THAT ARE TUNED BY GRID SEARCH METHOD

Method	Hyper-parameter	Range	Selected value
XGBoost	η	{0.1, 0.2, 0.3, 0.4, 0.5}	0.5
	ε	{0, 1, 2, 3, 4}	0
	α	{4, 5, 6, 7, 8}	7
	β	{1, 2, 3, 4, 5}	3
	δ	{0, 1, 2, 3, 4}	0, 2, 3, 4
	ζ	{0.25, 0.5, 0.75, 1}	1
	μ	{0.25, 0.5, 0.75, 1}	0.75
LR	θ_L	{0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7}	0.5
SVM (polynomial kernel)	cost	{0.001, 0.01, 0.1, 1, 10, 100, 1000}	1
SVM (radial kernel)	cost	{0.001, 0.01, 0.1, 1, 10, 100, 1000}	1
QDA	θ_Q	{0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7}	0.65
KNN	k	{1, 2, 3, 5, 10, 20}	1
NB	θ_N	{0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7}	0.65
MLP	s_M	{3, 4, 5, 6, 7}	5
	r_M	{0.05, 0.10, 0.15, 0.2, 0.25, 0.3}	0.25

TABLE IV
BINARY CLASSIFICATION CONFUSION MATRIX

	Predicted true	Predicted false
Actual true	TP	FN
Actual false	FP	TN

The shades of color are used in heatmaps to indicate the strength of correlation among features. A dark (resp. light) color represents a strong (resp. weak) correlation. It can be seen from Fig. 5(a) that there are many strongly correlated ones among the 72 features before correlation analysis. After that, the remaining 27 features are strongly correlated with themselves (i.e., the elements on the diagonal in Fig. 5(b) have the darkest color). It proves that correlation analysis effectively solves the collinearity problem among continuous features.

After feature selection based on statistical analysis methods, 330 features are retained, including 27 continuous and 303 categorical ones. In the procedure of machine learning-based feature selection, we take the 330 retained features as the inputs to XGBoost. According to the ranking of feature importance in descending order, the top 20 ones are chosen. Moreover, 10 of the top 20 ones are controllable, which are applied to train our classifiers.

XGBoost, LR, and their competitors are compared by using their trained models and their performance on the test set is

shown in Table VII. It shows that XGBoost has the best performance on specificity and three other metrics. LR has the best performance on sensitivity. We consider the five metrics together and identify that XGBoost and LR have better performance than the other methods. Their five evaluation metrics are high enough (all over 95%), which proves that they are ready to be employed in practice. Although XGBoost performs better than LR on specificity, ACC, F-measure, and AUC, LR has better sensitivity than XGBoost.

TABLE V
NUMBER OF FEATURES IN EACH STEP

Step		Number of features (except for the response)
	Original data	833
Data cleaning	Missing value cleaning	830
	Duplicate value cleaning	686
	Constant value cleaning	680
	Outliers correction	680
Statistical analysis	Chi-square test	575
	T-test	573
	Univariate logistic regression analysis	493
	Backward stepwise selection	375
	Pearson correlation analysis	330
Machine learning	XGBoost feature selection	20
	Controllable feature selection	10

TABLE VI
THE RESULT OF BACKWARD STEPWISE SELECTION

Backward stepwise selection	A_{IC} value	Number of continuous features
Start	-29510.2	190
End	-29665.5	72

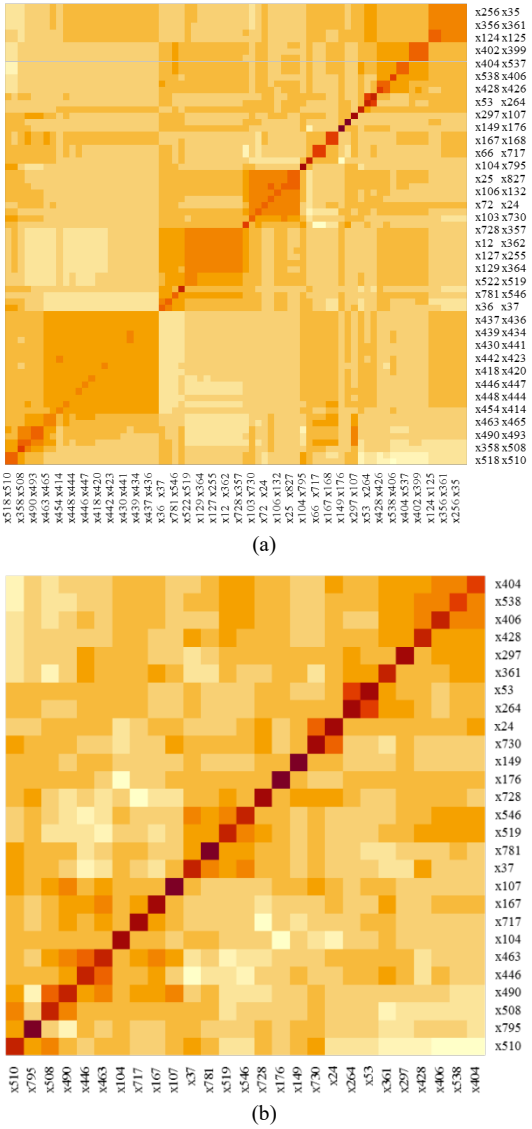

Fig. 5. The heatmaps of different feature counts before and after correlation analysis.

TABLE VII
COMPARISON OF THE PROPOSED PREDICTION METHODS

Prediction method	Confusion matrix		Sensitivity	Specificity	ACC	F-measure	AUC
XGBoost	1288 33	58 1521	0.9509	**0.9788**	**0.9686**	**0.9659**	**0.993**
Logistic Regression	1281 65	49 1505	**0.9632**	0.9586	0.9607	0.9574	0.960
SVM (polynomial kernel)	1263 46	83 1508	0.9383	0.9704	0.9555	0.9514	0.954
SVM (radial kernel)	1268 41	78 1513	0.9421	0.9736	0.9590	0.9552	0.958
KNN	1148 164	198 1390	0.8529	0.8945	0.8752	0.8638	0.874
QDA	1248 55	98 1499	0.9272	0.9646	0.9472	0.9422	0.946
NB	1226 72	120 1482	0.9108	0.9537	0.9338	0.9274	0.932
MLP	1272 44	74 1511	0.9450	0.9717	0.9593	0.9557	0.958

Among these methods, LR, QDA, and NB are probability-based ones for classification. Their comparative results in Table VII show that LR outperforms the other two methods. LR as a strong contender is not only able to provide high prediction performance but can also obtain an explicit expression as follows:

$$q(x) = 20.48 - 59.51x_{24} - 0.03405x_{730} - 0.5055x_{728} + 1.014x_{104} + 0.9331x_{53} - 0.053x_{107} + 0.001x_{361} - 24.26x_{717} + 0.004x_{176} + 0.0006x_{167} \quad (8)$$

The coefficients of the 10 input variables are visible. Their values indicate the importance of their corresponding variables in an LR model. Thus, practitioners can adjust the production parameters according to an LR model. It is extremely helpful to achieve the goal of reducing or even avoiding the generation of surplus material. Therefore, with its performance similar to an XGBoost model's, we consider an LR model to be a better one in practical applications due to its excellent interpretability while the latter cannot be interpreted.

VII. Concluding Remarks

This work investigates SMPP arising from a cold rolling process of steel coils. Taking a set of actual production data as an instance, we propose a three-stage approach to analyze and predict the generation of surplus material. The experimental results show that XGBoost and LR have greater performance than other popular competitors. Their results on five evaluation metrics are all greater than 95%, which proves their validity of prediction and the possibility of practical applications. The intelligent models obtained by the proposed method can be used in intelligent manufacturing processes to predict the generation of surplus materials, effectively reducing or replacing manual recognition. Besides, the LR model gives an explicit expression that can assist practitioners in adjusting process parameters to reduce the generation of surplus material.

Although the LR model gives a functional relationship between the selected features and the response, grid search used in this work only guarantees to find a good setting of hyperparameters. Finding the optimal hyperparameters that result in the fewest surplus materials is an optimization problem, which can be solved by intelligent optimization algorithms as our future work. Since the proposed method is highly extensible, it is capable of solving other prediction problems in most parts of the entire steel intelligent manufacturing processes such as casting, hot rolling, etc. Besides, other industrial applications such as SMPPs for hot-rolled slabs and wire rod products should be sought.

References

[1] Z. Zhao, S. Liu, M. Zhou, X. Guo and L. Qi, "Decomposition Method for New Single-Machine Scheduling Problems From Steel Production Systems," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 3, pp. 1376-1387, July 2020.

[2] Z. Zhao, S. Liu, M. Zhou and A. Abusorrah, "Dual-Objective Mixed Integer Linear Program and Memetic Algorithm for an Industrial Group Scheduling Problem," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 6, pp. 1199-1209, June 2021.

[3] S.Y. Nam, A. Zamaniam, T.J. Shin, *et al*., "A novel on-line model for the prediction of strip profile in cold rolling," *ISIJ International*, vol. 60, no. 2, pp. 308–317, 2020.

[4] A. Sanz-Garcia, J. Fernandez-Ceniceros, F. Antonanzas-Torres, *et al*., "GA-PARSIMONY: A GA-SVR approach with feature selection and parameter optimization to obtain parsimonious solutions for predicting temperature settings in a continuous annealing furnace," *Applied Soft Computing*, vol. 35, pp. 13–28, 2015.

[5] S. Lalam, P.K. Tiwari, S. Sahoo, and A.K. Dalal, "Online prediction and monitoring of mechanical properties of industrial galvanised steel coils using neural networks," *Ironmaking & Steelmaking*, vol. 46, no. 9, pp. 1–8, 2017.

[6] X. Lu, J. Sun, Z.X. Song *et al*., "Prediction and analysis of cold rolling mill vibration based on a data-driven method," *Applied Soft Computing*, vol. 96, 106706, 2020.

[7] Z. Zhao, X. Yong, S. Liu, and M. Zhou, "Data-driven surplus material prediction in steel coil production," *2020 29th Wireless and Optical Communications Conference (WOCC)*, Newark, NJ, USA, 2020.

[8] T. Chen, C. Guestrin, "XGBoost: a scalable tree boosting system," P*roceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, pp. 785–794, 2016.

[9] F. Yan, K. Song, Y. Liu, *et al.*, "Predictions and mechanism analyses of the fatigue strength of steel based on machine learning," *Journal of Materials Science*, vol. 55, no. 31, pp. 15334–15349, 2020.

[10] R. Katirci, H. Aktas, M. Zontul, "The prediction of the ZnNi thickness and Ni % of ZnNi alloy electroplating using a machine learning method," *Transactions of the Institute of Metal Finishing*, vol. 99, no. 3, pp. 162–168, 2021.

[11] G. James, D. Witten, T. Hastie, and R. Tibshirani, An Introduction to Statistical Learning. New York, NY: Springer New York, 2013.

[12] T. Alatarvas, T. Vuolio, E.P. Heikkinen, *et al*., "Prediction of inclusion state in molten steel by morphology and appearance of inclusions in liquid steel samples," *Steel Research International*, vol. 91, no. 2, 1900424, 2019.

[13] D. Kong, Y. Chen, N. Li, *et al*., "Tool wear estimation in end-milling of titanium alloy using NPE and a novel WOA-SVM model," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 7, pp. 5219–5232, 2019.

[14] M. Kotzabasaki, I. Sotiropoulos, C. Charitidis, *et al*., "Machine learning methods for multi-walled carbon nanotubes (MWCNT) genotoxicity prediction," *Nanoscale Advances*, vol. 2, no. 11, pp. 3167–3176, 2021.

[15] T. Thaler, P. Potocnik, I. Bric, *et al*., "Chatter detection in band sawing based on discriminant analysis of sound features," *Applied Acoustics*, vol. 77, pp. 114–121, 2014.

[16] S. Lalam, P.K. Tiwari, S. Sahoo, *et al*., "Online prediction and monitoring of mechanical properties of industrial galvanised steel coils using neural networks," *Ironmaking & Steelmaking*, vol. 46, no. 1, pp. 89–96, 2017.

[17] Y. Ji, S. Liu, M. Zhou, *et al*., "A machine learning and genetic algorithm-based method for predicting width deviation of hot-rolled strip in steel production systems," *Information Sciences*, vol. 589, pp. 360–375, 2022.

[18] I. Chatterjee, M. Zhou, A. Abusorrah, *et al*., "Statistics-based outlier detection and correction method for amazon customer reviews," *Entropy*, vol. 23, 2021, doi:10.3390/e23121645.

[19] P. Zhang, S. Shu and M. Zhou, "An online fault detection model and strategies based on SVM-grid in clouds," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 2, pp. 445–456, 2018.

[20] S. Gao, M. Zhou, Y. Wang, *et al*., "Dendritic neuron model with effective learning algorithms for classification, approximation and prediction," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 2, pp. 601−614, 2019.

Yingjun Ji received his B.S., M.S., and Ph. D. degrees in 2012, 2015, and 2022, respectively, from the College of Information Science and Engineering, Northeastern University, Shenyang, China. He is a Lecturer in Liaoning University. His research concentrates on intelligent manufacturing, industrial data analysis, intelligent decision-making, and intelligent optimization algorithm.

Ziyan Zhao received his B.S., M.S., and Ph. D. degrees in 2015, 2017, and 2021, respectively, from the College of Information Science and Engineering, Northeastern University, Shenyang, China. He is a Postdoctoral Researcher and Lecturer in Northeastern University. From October 2018 to October 2020, he worked as a visiting Ph. D. student in the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ, USA., supported by a scholarship from the China Scholarship Council. His research focuses on intelligent manufacturing, intelligent optimization algorithm, industrial big data, and production planning and scheduling. Till now, he has published over 20 international journal and conference papers in the above areas.

Shixin Liu received his B.S. degree in Mechanical Engineering from Southwest Jiaotong University, Sichuan, China in 1990, M.S. and Ph. D. degrees in Systems Engineering from Northeastern University, Shenyang, China in 1993, and 2000, respectively. He is currently a Professor of the College of Information Science and Engineering, Northeastern University, Shenyang, China. His research interests are in intelligent manufacturing, industrial big data, intelligent decision-making, and production planning and scheduling. He has over 100 publications including one book.

Xiaoyue Yong received her B.S. and M.S. degrees in 2016 and 2019 from the College of Information Science and Engineering, Northeastern University, Shenyang, China. She is currently an engineer at Shanghai Baosight Software Co., Ltd, China. Her research focuses on industrial data analysis, intelligent manufacturing, intelligent decision-making, and production planning and scheduling.