

Article

Analysis of Operational Control Data and Development of a Predictive Model of the Content of the Target Component in Melting Products

Natalia Vasilyeva *  and Ivan Pavlyuk

Mineral Raw Material Processing Faculty, Saint Petersburg Mining University, 199106 St. Petersburg, Russia

* Correspondence: vasileva_nv@pers.spmi.ru

Abstract: The relevance of this research is due to the need to stabilize the composition of the melting products of copper–nickel sulfide raw materials. Statistical methods of analyzing the historical data of the real technological object and the correlation analysis of process parameters are described. Factors that exert the greatest influence on the main output parameter (the fraction of copper in a matte) and ensure the physical–chemical transformations are revealed: total charge rate, overall blast volume, oxygen content in the blast (degree of oxygen enrichment in the blowing), temperature of exhaust gases in the off-gas duct, temperature of feed in the smelting zone, copper content in the matte. An approach to the processing of real-time data for the development of a mathematical model for control of the melting process is proposed. The stages of processing of the real-time information are considered. The adequacy of the models was assessed by the value of the mean absolute error (MAE) between the calculated and experimental values.

Keywords: statistical data; data preparation; correlation coefficient; correlation analysis; data approximation



Citation: Vasilyeva, N.; Pavlyuk, I. Analysis of Operational Control Data and Development of a Predictive Model of the Content of the Target Component in Melting Products. *Eng* **2024**, *5*, 1752–1767. <https://doi.org/10.3390/eng5030092>

Academic Editor: Iolanda De Marco

Received: 2 July 2024

Revised: 30 July 2024

Accepted: 3 August 2024

Published: 5 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In the study of the metallurgical process as an object of automatic control and in addressing its optimization, mathematical models that reflect the dependence of output indicators on input indicators and control actions acquire paramount importance [1–3]. There is no single methodology for constructing mathematical models [4,5]. This is due to the great diversity of types of control objects [6,7]: static and dynamic, continuous and discrete, deterministic and stochastic, etc.

Today, several methods exist for preparing production data for modeling [8,9]; however, it is not possible to say definitively which methods are preferable to use in any given case for solving specific tasks [10,11].

The aim of this work is to identify the methods most suitable for preparing operational control data for the development of a control mathematical model of the technological process.

The solution to the problem is considered through the example of processing operational control data of the smelting process of copper–nickel sulfide raw materials in a Vanyukov furnace (Vanyukov process) [12–14].

The research involved databases of operational control information on the Vanyukov process over 5 months of continuous operation of the unit. The analysis of the operational control data of the Vanyukov process showed (Figure 1) that the copper content in the matte fluctuated within the range of 46–68%. That is, the range of fluctuations was more than 20% absolute with an average copper content in the matte of 58% [15].

This indicates the heterogeneity of the data (a large spread relative to the mean value) and the low predictability of the copper content in the matte, i.e., the low stability of the process. The more stable the process, the easier it is to predict, and the smaller the

deviations in copper content in the matte from the specified value, the more accurate the prediction can be [16–18].

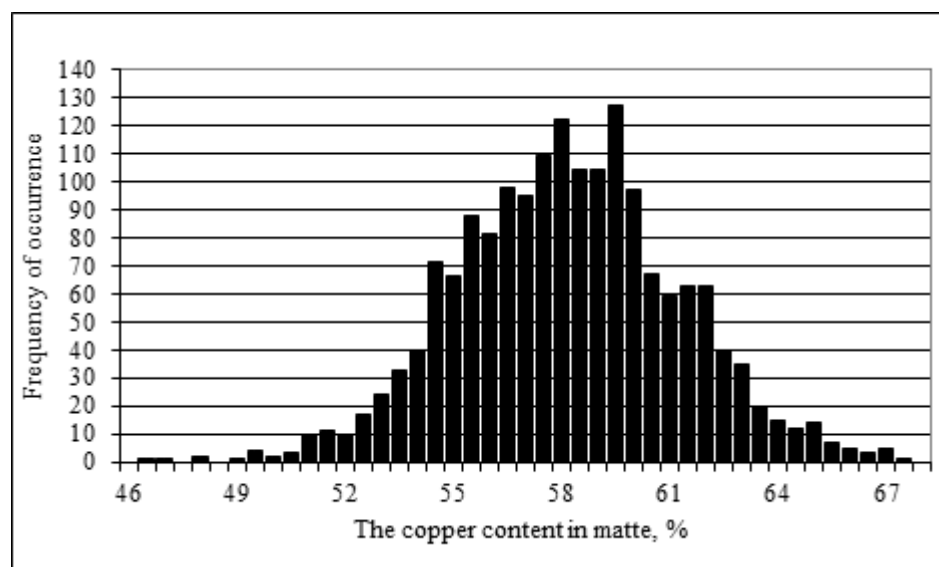


Figure 1. Histogram of the distribution of copper content in matte.

One possible reason for such significant discrepancies (both in terms of the range of fluctuations and the mean value) is the deviation of the actual temperature of the physicochemical reactions from the required one, due to the inconsistency between the supply of enriched air and the charging of the batch (violation of the technological regime of the process) [19].

Thus, the relevance of this work is driven by the existing need to improve the quality of the smelting target products (copper content in the matte) by developing and implementing a control mathematical model of the Vanyukov process to stabilize the copper content in the matte.

2. Data Analysis

The entire code was implemented in the Jupyter Notebook environment, as it offers simple functionality and a user-friendly and optimized interface, making it well-suited to addressing the set task. The initial data were divided into two groups: data characterizing the furnace load (recorded once per minute, totaling 84,354 rows), and data on the composition of smelting products (recorded once every two hours, totaling 1844 rows).

After reading the files using the Pandas library, it was necessary to check them for the presence of missing values (Figure 2).

The furnace load data table did not contain any missing values (Figure 2). The table with data on the composition of smelting products (Figure 3) had a convenient data type, but the “date” column had more values than the others.

The absence of missing values does not guarantee the “purity” of the data; values may be filled incorrectly, for example, with non-numeric characters instead of the required numbers. In such cases, the cell is assigned the type NaN, and there are two options—either replace the value (for example, with the column’s mean) or delete the row/column. To decide, we examined the records containing NaN.

Analysis of Figure 4 revealed that on certain days, information about the values of smelting product components was missing, meaning they could be disregarded, i.e., the corresponding rows could be deleted. This left a total of 1718 rows.

At this stage, we proceeded to merge the tables with independent variables, where values were recorded every minute, with the target variable, which was recorded at a 2 h interval. There are two approaches: the first is to merge tables with matching time

values, i.e., merging based on the “date” column, ignoring intermediate values. The second approach involves averaging over the 2 h interval, considering all rows and values; for example, the value at 2:00 p.m. will have the interval (12:00 p.m.; 2:00 p.m.).

```
Data columns (total 41 columns):
#      Column                                     Non-Null Count  Dtype
---  -
0      date                                           84355 non-null  object
1      blast furnace pressure, point 1                 84355 non-null  object
2      blast furnace pressure, point 2                 84355 non-null  object
3      natural gas pressure                             84355 non-null  object
4      conveyor 31, productivity                       84355 non-null  object
5      conveyor 31, speed                             84355 non-null  object
6      conveyor 32, productivity                       84355 non-null  object
7      conveyor 32, speed                             84355 non-null  object
8      feeder 1, level                                84355 non-null  object
9      feeder 1, speed                                84355 non-null  object
10     feeder 1, productivity                          84355 non-null  object
11     feeder 2, level                                84355 non-null  object
12     feeder 2, speed                                84355 non-null  object
13     feeder 2, productivity                          84355 non-null  object
14     feeder 3, level                                84355 non-null  object
15     feeder 3, speed                                84355 non-null  object
16     feeder 3, productivity                          84355 non-null  object
17     feeder 4, level                                84355 non-null  object
18     feeder 4, speed                                84355 non-null  object
19     feeder 4, productivity                          84355 non-null  object
20     feeder 5, level                                84355 non-null  object
21     feeder 5, speed                                84355 non-null  object
22     feeder 5, productivity                          84355 non-null  object
23     feeder 6, level                                84355 non-null  object
24     feeder 6, speed                                84355 non-null  object
25     feeder 6, productivity                          84355 non-null  object
26     feeder 7, speed                                84355 non-null  object
27     feeder 8, speed                                84355 non-null  object
28     feeder 7, level                                84355 non-null  object
29     feeder 8, level                                84355 non-null  object
30     vacuum in the bunker                           84355 non-null  object
31     blast furnace flow                             84355 non-null  object
32     natural gas flow                               84355 non-null  object
33     oxygen content in the blast furnace             84355 non-null  object
34     blast furnace temperature                     84355 non-null  object
35     temperature of outgoing gases in the bunker    84355 non-null  object
36     temperature of the feed, slag siphon           84355 non-null  object
37     temperature of the feed, matte siphon          84355 non-null  object
38     temperature of the feed, melting zone, point 1 84355 non-null  object
39     temperature of the feed, melting zone, point 2 84355 non-null  object
40     temperature of natural gas                     84355 non-null  object
dtypes: object(41)
memory usage: 26.4+ MB
```

Figure 2. Results of applying the info() method to the first table.

```
Data columns (total 6 columns):
#      Column          Non-Null Count  Dtype
---  -
0      Measurement ID    1730 non-null  float64
1      Date              1856 non-null  datetime64[ns]
2      [Cu]              1730 non-null  float64
3      [Ni]              1730 non-null  float64
4      [Fe]              1730 non-null  float64
5      [S]               1730 non-null  float64
dtypes: datetime64[ns](1), float64(5)
memory usage: 87.1 KB
```

Figure 3. Results of applying the info() method to the fourth table.

We evaluated the accuracy of the obtained data using two different methods. Accuracy characterizes the degree of reliability of information and its approximation to the original that it expresses. Several ways to calculate it include:

- Calculating the sum of the deviations of the average values by columns in the obtained table from the average values of the original;

— Creating a correlation matrix to determine the relationship between the quantities in the initial and final tables.

	Measurement ID	Date	[Cu]	[Ni]	[Fe]	[S]
65	NaN	2008-01-06 11:00:00	NaN	NaN	NaN	NaN
66	NaN	2008-01-06 13:00:00	NaN	NaN	NaN	NaN
67	NaN	2008-01-06 15:00:00	NaN	NaN	NaN	NaN
68	NaN	2008-01-06 17:00:00	NaN	NaN	NaN	NaN
69	NaN	2008-01-06 19:00:00	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...
1793	NaN	2008-05-29 11:00:00	NaN	NaN	NaN	NaN
1794	NaN	2008-05-29 13:00:00	NaN	NaN	NaN	NaN
1795	NaN	2008-05-29 15:00:00	NaN	NaN	NaN	NaN
1796	NaN	2008-05-29 17:00:00	NaN	NaN	NaN	NaN
1797	NaN	2008-05-29 19:00:00	NaN	NaN	NaN	NaN

126 rows × 6 columns

Figure 4. Display of rows with incorrect data types in the fourth table.

As seen in Figure 2, all columns had the “object” type, to which most necessary logical operations are not applicable. However, changing data types in all columns was not necessary at this point, except for “date”, as it was to be used for merging and its type must match the one being merged. Thus, implementing the two methods described above resulted in two data frames (one for each method) for further comparison and evaluation.

As a result of applying the first method to the data, it was found that in the table obtained by a simple combination of common rows, accuracy was lost to a lesser extent, making it preferable.

As is known, machine learning models, especially linear models and the gradient descent method, usually perform better with normalized data [20,21]. Data normalization is the process of scaling feature values to a standard range or distribution [22,23]. Normalization helps balance the influence of different features and can improve model performance. The StandardScaler data normalization method was used in the work, which is one of the common approaches to data normalization. It is based on standardizing data, meaning transforming feature values so that they have a mean of 0 and a standard deviation of 1.

The formula for transforming data using StandardScaler is as follows:

$$x_{scaled} = \frac{x - mean}{sd}$$

where x_{scaled} —transforming data, x is the original feature values, $mean$ is the mean value of the feature, and sd is the standard deviation of the feature.

As a result, missing values were removed from the original dataset, two methods of merging data with different time stamps were considered, and the best one was selected based on accuracy evaluation. Then, the obtained data table was optimized for linear models using the StandardScaler method.

From the initial dataset, the most significant parameters reflecting the process under consideration were selected:

Total charge rate, t/h;

Overall blast volume, m³/h;

Oxygen content in the blast (degree of oxygen enrichment in the blowing), %;

Temperature of exhaust gases in the off-gas duct, °C;

Temperature of feed in the smelting zone, °C;

Copper content in the matte, %.

The remaining parameters were excluded from consideration for the following reasons:

- They were indirect, meaning they do not have a direct impact on the process under consideration;
- They were significantly damaged.

3. Methods

The goal of statistical data analysis, specifically correlation analysis, since all the data in the study were quantitative, was to identify statistically significant correlation coefficients describing the relationship between the target variable and the independent variables [24–26]. As a result, two data frames were formed for further model training—the first contained variables (input values) with sufficient correlation coefficients, and the second consisted only of the column of output values, i.e., the copper content in the matte.

The matrix of correlation coefficients for the data is provided in Table 1.

Table 1. The matrix of correlation coefficients for the data (significant coefficients according to the Student’s criterion are highlighted in bold).

	Total charge rate, t/h	Overall blast volume, m ³ /h	Oxygen content in the blast (degree of oxygen enrichment in the blowing), %	Temperature of exhaust gases in the off-gas duct, °C	Temperature of feed in the smelting zone, °C	Copper content in the matte, %
Total charge rate, t/h	1.000	0.637	0.693	−0.176	−0.156	−0.052
Overall blast volume, m ³ /h	0.637	1.000	0.602	−0.018	0.157	−0.136
Oxygen content in the blast (degree of oxygen enrichment in the blowing), %	0.693	0.602	1.000	0.006	−0.005	−0.155
Temperature of exhaust gases in the off-gas duct, °C	−0.176	−0.018	0.006	1.000	−0.056	−0.068
Temperature of feed in the smelting zone, °C	−0.156	0.157	−0.005	−0.056	1.000	−0.146
Copper content in the matte, %	−0.052	−0.136	−0.155	−0.068	−0.146	1.000

Below are scatter plots illustrating the relationship between the dependent variable and the features (Figure 5):

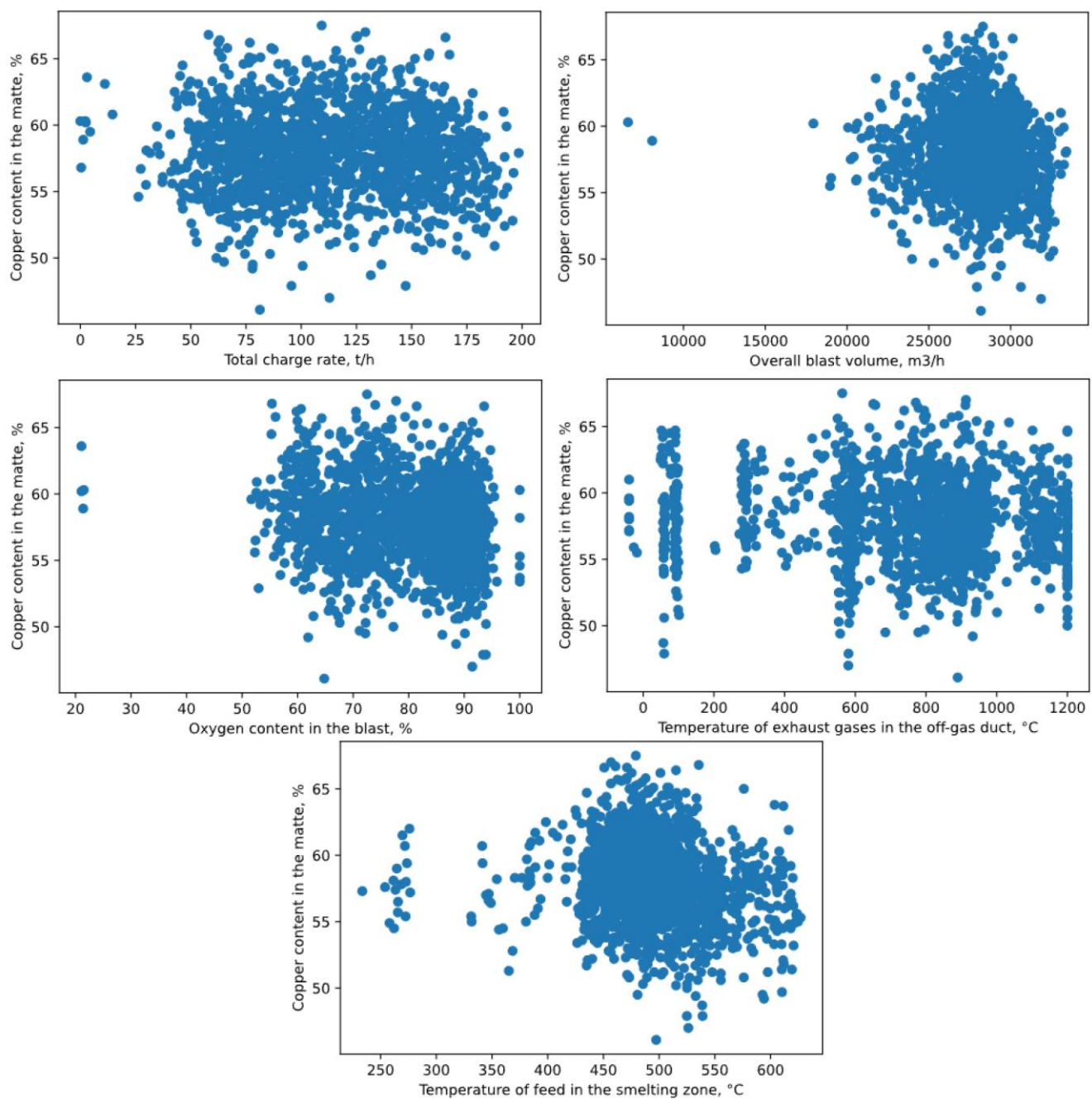


Figure 5. Plots of the target variable against the features.

In Figure 5, it is evident that each independent variable contains outliers, values atypical for that particular feature. They are situated on the left, while the majority of values cluster on the right. After removing atypical cases, new dependency plots were obtained (Figure 6), along with a new correlation matrix (Table 2):

Table 2. The matrix of correlation coefficients for the data (significant coefficients according to the Student's criterion are highlighted in bold).

	Total charge rate, t/h	Overall blast volume, m ³ /h	Oxygen content in the blast (degree of oxygen enrichment in the blowing), %	Temperature of exhaust gases in the off-gas duct, °C	Temperature of feed in the smelting zone, °C	Copper content in the matte, %
Total charge rate, t/h	1.000	0.632	0.691	−0.162	−0.147	−0.044
Overall blast volume, m ³ /h	0.632	1.000	0.577	−0.010	0.190	−0.134
Oxygen content in the blast (degree of oxygen enrichment in the blowing), %	0.691	0.577	1.000	0.0023	0.0004	−0.149
Temperature of exhaust gases in the off-gas duct, °C	−0.162	−0.010	0.023	1.000	−0.033	−0.069
Temperature of feed in the smelting zone, °C	−0.147	0.190	−0.0004	−0.033	1.000	−0.170
Copper content in the matte, %	−0.044	−0.134	−0.149	−0.069	−0.170	1.000

The analysis of the correlation matrix of the data revealed the following:

1. All parameters characterizing the furnace load had a sufficiently high correlation (correlation coefficient around 0.6–0.7). This suggests that the operator tried to maintain the required ratio of “charge load—blowing rate” regulated by the protocol. However, this seems to be insufficient, as it did not affect the copper content in the matte (all correlation coefficients were insignificant).
2. The feed temperature of the furnace in the smelting zone, although correlated with the furnace load parameters (temperature increased with an increase in overall blast volume—positive correlation, and feed temperature decreased with an increase in charge rate—negative correlation), was relatively weakly correlated (correlation coefficient around 0.15).

Therefore, while the operator attempted to control the process, their actions were insufficient to ensure the required quality of smelting products. Hence, there is a clear need for a “tighter” integration of all process parameters to achieve the desired quality of smelting products.

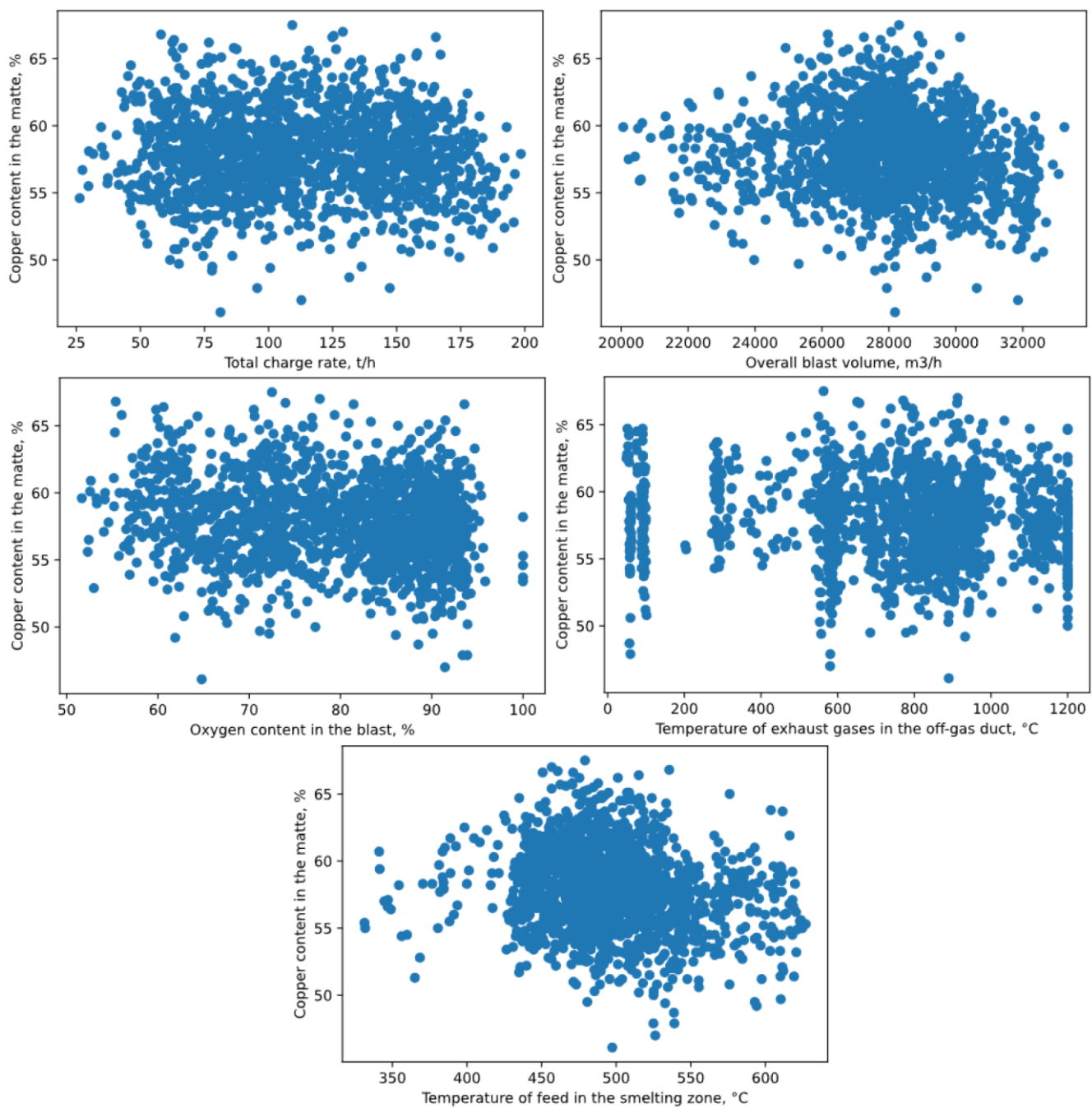


Figure 6. Plots of the target variable against the features without outliers.

4. Results

A small amount of data was available, so the following machine learning models were considered:

- (1) Linear regression—one of the simplest and widely used methods in machine learning, based on the assumption of a linear relationship between input features and the output variable. Linear regression seeks to find the best straight line that most accurately fits the data.
- (2) Stochastic gradient descent (SGDRegressor)—an optimization method for linear regression. Unlike regular linear regression, SGDRegressor updates the model parameters using gradient descent at each iteration.

- (3) Decision tree—a machine learning model that makes decisions based on the sequential application of conditions to input data. It builds a tree where each node represents a specific condition, and each leaf node represents a specific prediction.
- (4) Random forest—an ensemble machine learning model consisting of multiple decision trees. Each tree is trained on a subset of data and a subset of features. In the end, predictions from each tree are combined to obtain the final prediction.

4.1. Linear Regression

These models were chosen due to the small size of the data, because in such a case simple models like linear regression and decision trees may perform better than complex and computationally demanding models such as random forests [27–29].

The first step was to check the homoscedasticity property, train the model on the training data, and examine the following plots (Figure 7):

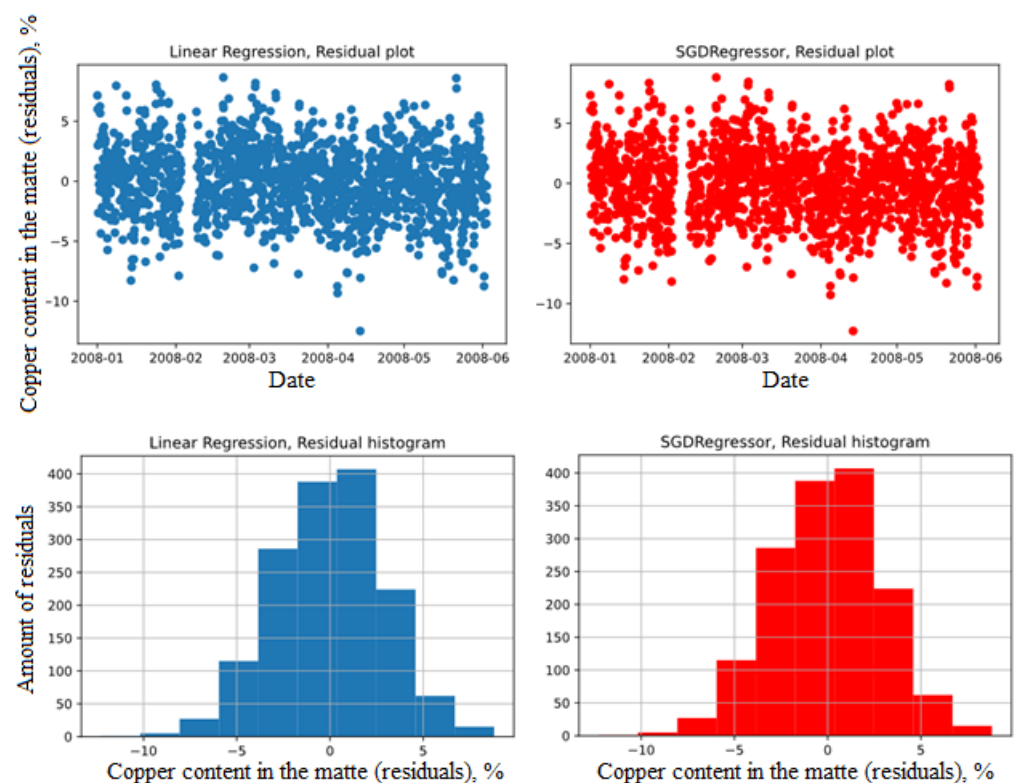


Figure 7. Plots and histograms of residuals of linear models: (left)—linear regression, (right)—stochastic gradient descent.

Here, residuals were the difference between the predicted and actual copper content in the matte, expressed as a percentage. Residuals serve as an indicator of modeling accuracy. The modeling error did not exceed 10% (Figure 7).

The residuals maintained homoscedasticity, and the distribution was normal; therefore, homoscedasticity was satisfied, and linear models were suitable for solving this problem [30,31]. Let us also consider the issue of autocorrelation, a phenomenon where the errors (residuals) of the linear model are correlated over time or observation order. In other words, it is the presence of a systematic relationship between errors in different observations. According to the Durbin–Watson criterion, if its calculated value is equal to 2, autocorrelation is absent. In our case, there was no correlation between errors.

The next step required dividing the samples, and for this purpose, the `train_test_split` function from the `scikit-learn` library was convenient, as it randomly splits the data into training and test subsets. A check was performed (based on the mean and variance of deviations) to ensure that the training and test samples belonged to the same population.

After declaring an empty model, optimal parameters for the training sample can be found using another function from the scikit-learn library—`RandomizedSearchCV`. It takes a dictionary of parameters as one of the arguments, the combination of which is expected to yield the best result. It is used in conjunction with the `best_estimator_method`, which iterates and evaluates all possible parameter combinations on the training set. In the decision tree and random forest models, the parameter dictionary was the same for the sake of comparison.

To assess the accuracy of the obtained predictive model, the following metrics using the error between the predicted value and the actual value were employed:

- (1) Mean absolute error (MAE);
- (2) Root mean squared error (RMSE);
- (3) Mean absolute percentage error (MAPE);
- (4) R-squared score.

MAE (mean absolute error) represents the average of absolute deviations between predicted and actual values. This metric measures the average absolute error and indicates how accurate the model predictions were.

RMSE (root mean squared error) is a widely used metric representing the square root of the average of the squared error between predicted and actual values.

In this work, these two metrics were specifically used to evaluate model accuracy. The methods `mean_absolute_error` and `mean_squared_error` are located in the metrics module of the scikit-learn library.

To understand how well the models will perform on unfamiliar data, i.e., to determine whether they are overfitting or not, it is useful to construct a learning curve. If a model is overfit (overfitting), it will show good metrics (low MAE and RMSE) on training data but perform poorly on test data. An underfitting model shows poor results on both the training and test sets; in other words, it has high bias. The learning curve allows checking how the model's accuracy changes depending on the sample size. The goal of building a learning curve is to identify problems of underfitting and overfitting and determine the optimal size of the training sample and the model's complexity level [32–34].

The learning curve plots for the models are provided in Figure 8.

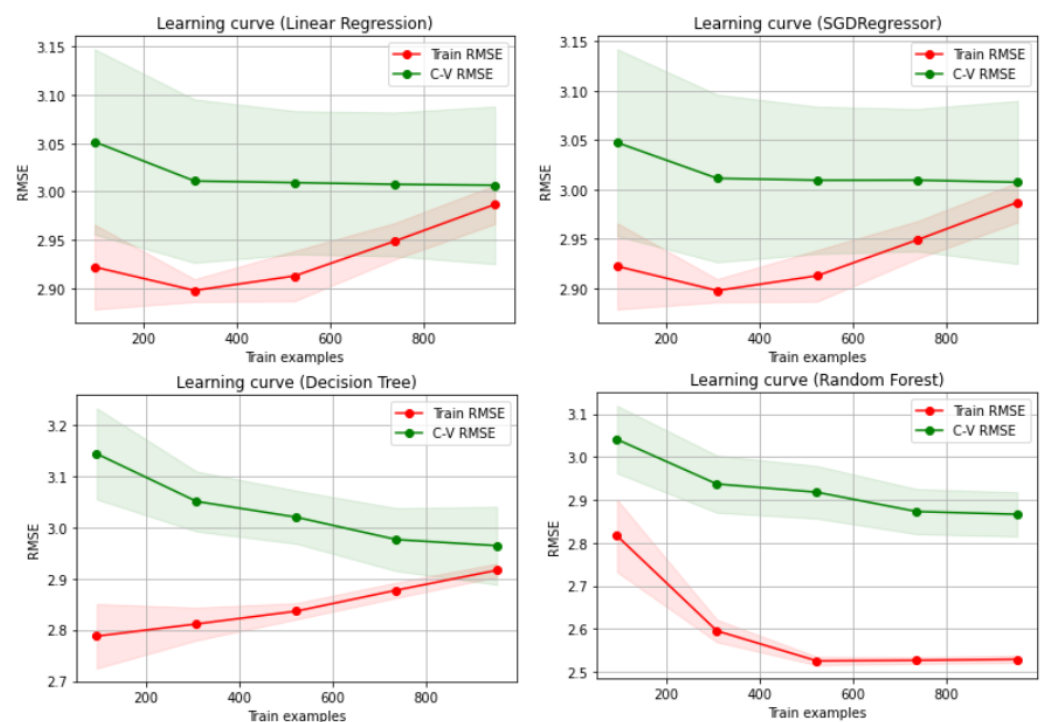


Figure 8. Constructed learning curve graphs.

Based on the analysis of the curves, the following conclusions were drawn:

- (1) In the graphs of linear models and decision trees, there was no overfitting issue, as the training and test curves were quite close to each other;
- (2) Linear regression and stochastic gradient descent models did not improve performance on cross-validation as the training sample increased; therefore, they reached their best RMSE;
- (3) The random forest model had better performance on cross-validation, despite its curve showing an overfitting problem.

In the case of underfitting, it is recommended to increase the number of independent variables in the model and increase its complexity. On the other hand, overfitting implies reducing the model's complexity and the number of indicators, for example, through regularization, as well as increasing training data. Regularization is a method commonly applied to linear models, imposing some restrictions on the model by adding a penalty to the loss function, thereby hindering the learning of overly complex relationships typical of the training sample.

4.2. Addressing the Overfitting Issue

First, consider the case of the random forest model. The size of the training sample had already stopped affecting the result, so considering that the model was nonlinear, reducing the number of features and tuning hyperparameters was explored. To decide which features to keep, it was necessary to look at the importance of each in predicting the target variable. The `feature_importances_` method was convenient for this purpose.

From Figure 9, it can be concluded that all the features used contributed at least 10% to the result. Therefore, considering their removal was a last resort. Consequently, it was necessary to tune the hyperparameters of this model, with the main ones being the depth of the tree and the number of decision trees. Through empirical testing using the `RandomizedSearchCV` method, optimal parameters were found, and the learning curve of the tuned random forest is presented in Figure 10.

The results of the initial testing of the models are summarized in Table 3.

Table 3. The results of the initial testing of the models.

Model	MAE	RMSE
Linear Regression	2.379	3.011
SGDRegressor	2.374	3.007
Decision Tree	2.385	3.009
Random Forest	2.369	3.005

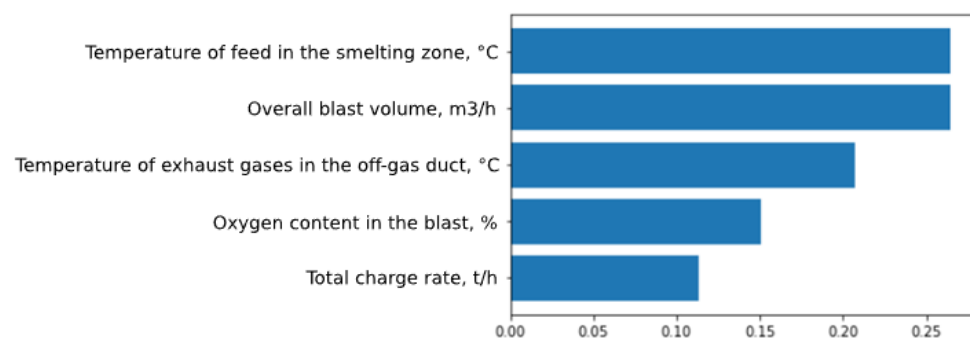


Figure 9. Result of calculating the importance of random forest features.

Now, judging by the learning curves, all the considered models had the characteristic of not improving their performance despite the increase in data. This allowed us to view them all in terms of the high bias problem.

Interpretation of linear model coefficients for each feature.

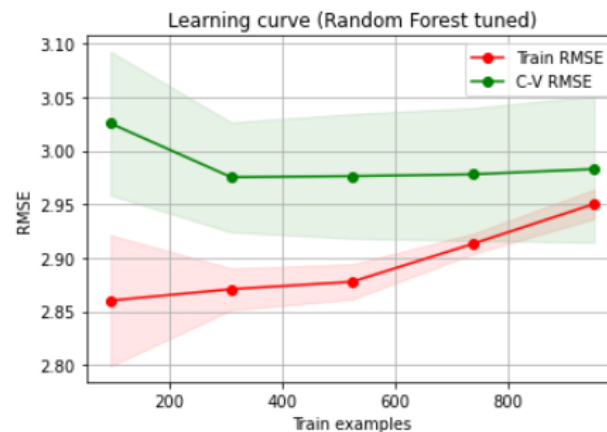


Figure 10. Learning curve of the random forest model.

4.3. High Bias Problem and Polynomial Features

As seen in Figure 11, all indicators had a nonlinear (polynomial) relationship with the dependent variable. However, adding polynomial features can help linear models in modeling complex nonlinear relationships. Instead of assuming a simple linear relationship between the features and the target variable, we could include polynomial features that allow for more complex relationships [35–37].

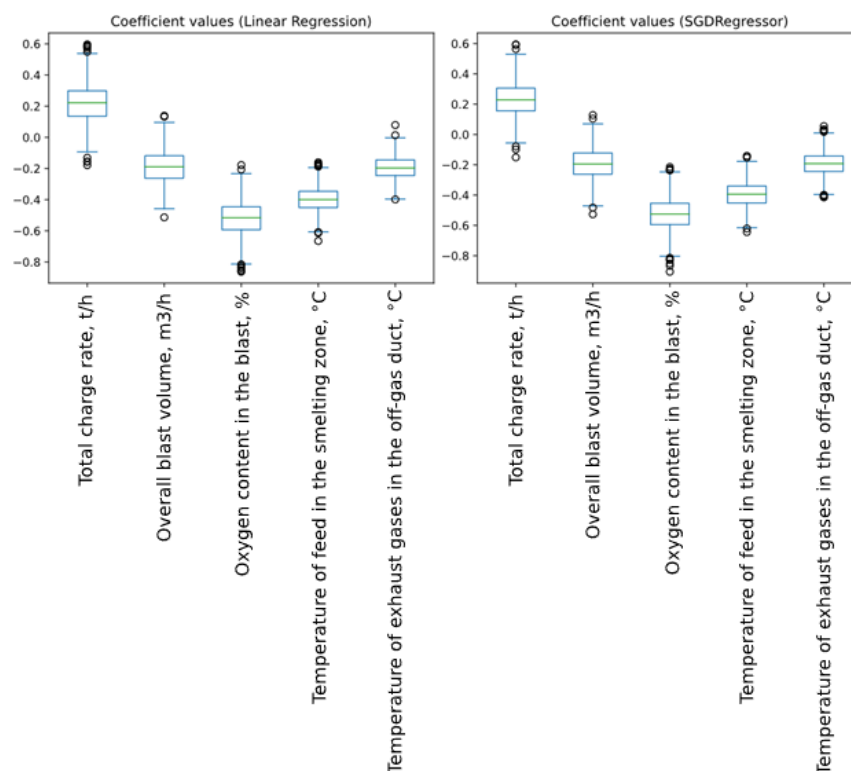


Figure 11. Model interpretation: (left)—linear regression, (right)—gradient descent.

The PolynomialFeatures method is one approach to creating polynomial features based on the original data. It allows adding to the data powered and multiplied combinations of the original features, creating a polynomial relationship and thereby complicating the models. PolynomialFeatures transforms the original features into a new set of features

that includes all possible combinations of the powers of the original features. For example, features x_1, x_2 can be transformed into a new set of features: $1, x_1, x_2, x_1^2, x_1x_2, x_2^2$.

In our case, the PolynomialFeatures function generated a new feature matrix from the input data, consisting of all their polynomial combinations with a degree less than or equal to 2, increasing the number of features from 5 to 20. Then, the data were normalized, and the learning curves are presented in Figure 12.

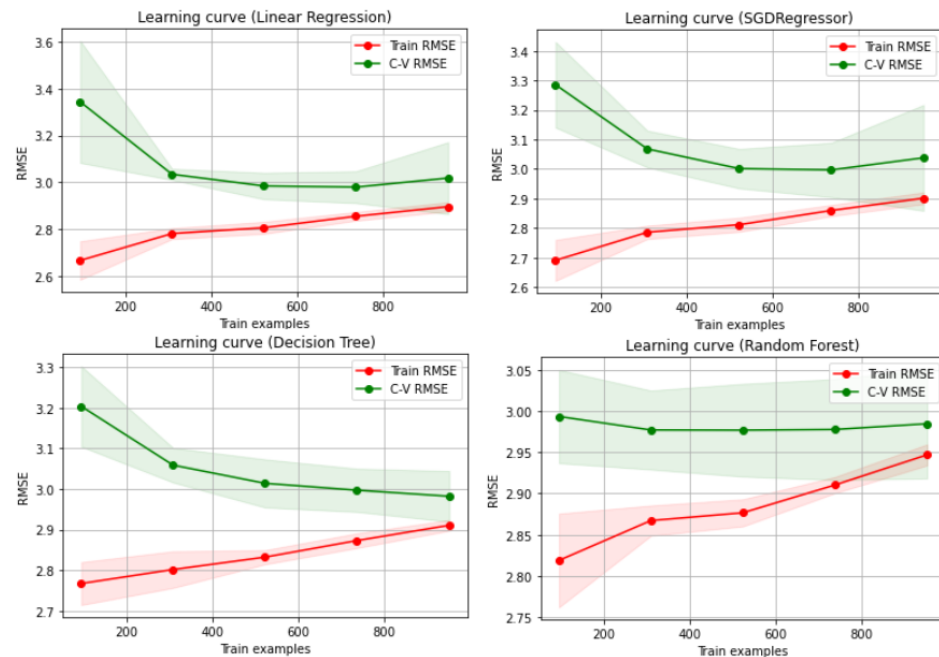


Figure 12. Learning curves after increasing the number of features.

This method has a drawback. Since the number of features increased, models required more data to learn the new complex relationships, so the size of the training dataset was increased by 20% (Figure 13).

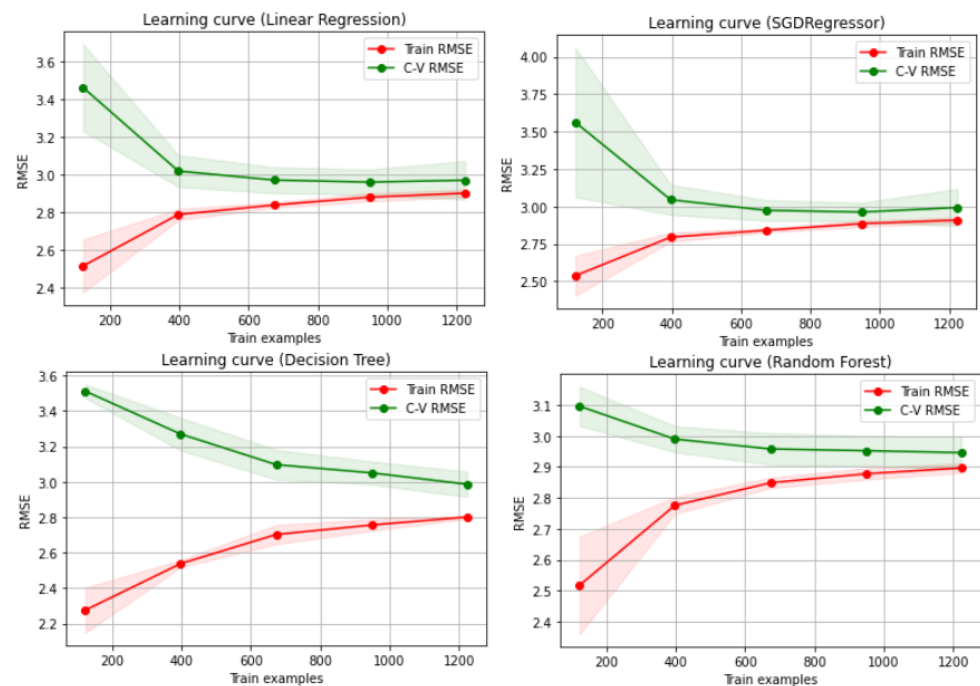


Figure 13. Learning curve plots after increasing the dataset size.

The results of the testing are summarized in Table 4.

Table 4. The results of the testing of the models.

Model	MAE	RMSE
Linear Regression	2.169 (−8.8%)	2.803 (−6.9%)
SGDRegressor	2.153 (−9.3%)	2.774 (−7.7%)
Decision Tree	2.276 (−4.6%)	2.930 (−2.6%)
Random Forest	2.254 (−4.9%)	2.867 (−4.6%)

The plots illustrating the dependencies of calculated values by each model on the original (test) values are provided below. The dashed line highlighted in blue represents the linear approximation of the model predictions, while the cyan line represents $y = x$ (ideal predictions) (Figure 14):

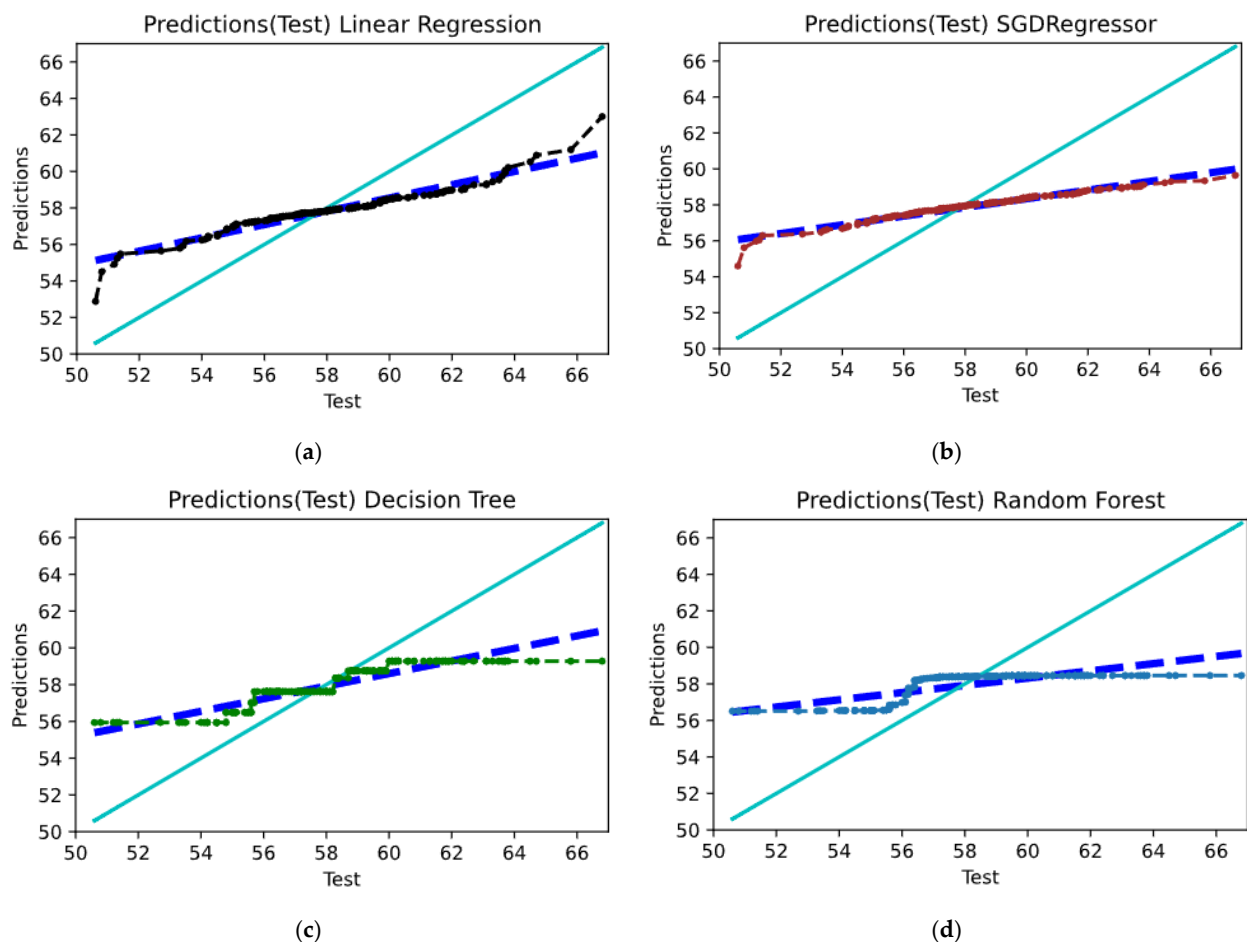


Figure 14. Dependency plot of predicted values on test values for each model: (a) linear regression; (b) SGDRegressor; (c) decision tree; (d) random forest.

5. Conclusions

An analysis of the correlation coefficients between the main control variables of the process (Tables 1 and 2) revealed that the relationship between the charge feed and all blast parameters was significantly different from zero. However, their magnitude was clearly insufficient for effective process control, as the random component in the scatter of points across the correlation fields was too large (Figures 5 and 6).

All considered models for predicting the copper content in the matte showed approximately similar results. The modeling error ranged from 5% to 10%, with the decision tree model demonstrating the best performance (MAE 5%, RMSE 3%).

In the final dependency plots, crucial parameters included the arrangement of data points and the angles of deviation between the approximation and the line of ideal predictions. Through analysis, it was concluded that there was a non-random systematic error present in the observations. This systematic error may have been caused by several factors, including malfunctioning, worn out, or incorrectly tuned process control equipment (APCS), operator errors, insufficient quality control, and others.

It is necessary to conduct a series of additional, more in-depth studies to determine whether the systematic error is caused by significant fluctuations in the furnace charge parameters (human factor) or by the lack of coordination between the amount of technical oxygen supplied and the furnace charge (a deficiency in the automated control system). It is possible that both problems exist. Depending on the findings, a decision can then be made on how to improve the quality of the process.

In conclusion, for the accurate regulation of the technological process and ensuring the quality of the melt product, a more thoughtful approach to the technical aspects of the process, personnel training, and the correct methodology of measurement is necessary.

Author Contributions: Conceptualization, N.V.; methodology, N.V.; software, I.P.; verification, N.V., I.P.; preparation of initial draft, I.P.; writing—review and editing, N.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research is supported by the government assignment of Ministry of Science and Higher Education of the Russian Federation FSRW-2023-0002, executed at the Saint Petersburg mining university.

Data Availability Statement: Data is contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

- George, G.; Lavie, D. Big data and data science methods for management research. *Acad. Manag. J.* **2016**, *59*, 1493–1507. [\[CrossRef\]](#)
- Aazam, M.; Zeadally, S.; Harras, K.A. Deploying fog computing in industrial internet of things and industry 4.0. *IEEE Trans. Ind. Inform.* **2018**, *14*, 4674–4682. [\[CrossRef\]](#)
- Thillaieswari, B. Comparative Study on Tools and Techniques of Big Data Analysis. *Int. J. Adv. Netw. Appl. (IJANA)* **2017**, *08*, 61–66.
- Thombansen, U.; Purrio, M.; Buchholz, G.; Hermanns, T.; Molitor, T.; Willms, K.; Schulz, W.; Reisinger, U. Determination of process variables in melt-based manufacturing processes. *Int. J. Comput. Integr. Manuf.* **2016**, *29*, 1147–1158. [\[CrossRef\]](#)
- Aleksandrova, T.N. Complex and deep processing of mineral raw materials of natural and technogenic origin: State and prospects. *J. Min. Inst.* **2022**, *256*, 503–504.
- Zhukovskiy, Y.L.; Korolev, N.A.; Malkova, Y.M. Monitoring of grinding condition in drum mills based on resulting shaft torque. *J. Min. Inst.* **2022**, *256*, 686–700. [\[CrossRef\]](#)
- Oprea, G.; Andrei, H. Power quality analysis of industrial company based on data acquisition system, numerical algorithms and compensation results. In Proceedings of the 2016 International Symposium on Fundamentals of Electrical Engineering, ISFEE, Bucharest, Romania, 30 June–2 July 2016; p. 7803232. [\[CrossRef\]](#)
- Wang, H.-Y.; Wu, W.-D. Statistical Process Control Based on Two Kinds of Feedback Adjustment for Autocorrelated Process. In Proceedings of the 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, Dalian, China, 12–17 October 2008. [\[CrossRef\]](#)
- Fahle, S.; Prinz, C.; Kuhlenkötter, B. Systematic review on machine learning (ML) methods for manufacturing processes—identifying artificial intelligence (AI) methods for field application. *Procedia CIRP* **2020**, *93*, 413–418. [\[CrossRef\]](#)
- Potapov, A.I.; Kulchitskii, A.A.; Smorodinskii, Y.G.; Smirnov, A.G. Evaluating the Error of a System for Monitoring the Geometry of Anode Posts in Electrolytic Cells with Self-Baking Anode. *Russ. J. Nondestruct. Test.* **2020**, *56*, 268–274. [\[CrossRef\]](#)
- Ilyushin, Y.V.; Kapostey, E.I. Developing a Comprehensive Mathematical Model for Aluminium Production in a Soderberg Electrolyser. *Energies* **2023**, *16*, 6313. [\[CrossRef\]](#)
- Lutskiy, D.S.; Ignatovich, A.S. Study on hydrometallurgical recovery of copper and rhenium in processing of substandard copper concentrates. *J. Min. Inst.* **2021**, *251*, 723–729. [\[CrossRef\]](#)
- Nguyen, H.H.; Bazhin, V.Y. Optimization of the Control System for Electrolytic Copper Refining with Digital Twin During Dendritic Precipitation. *Metallurgist* **2023**, *67*, 41–50. [\[CrossRef\]](#)

14. Kolesnikov, A.S. Kinetic investigations into the distillation of nonferrous metals during complex processing of waste of metallurgical industry. *Russ. J. Non-Ferr. Met.* **2015**, *56*, 1–5. [\[CrossRef\]](#)
15. Vasilyeva, N.V.; Fedorova, E.R. Obrabotka bolshogo massiva dannyh operativnogo kontrolya i podgotovka ego k razrabotke avtomatizirovannoy sistemy upravleniya tekhnologicheskim processom. In *Promyshlennye ASU i Kontrollery*; Scientific & Technical Literature Publishing House: Moscow, Russia, 2019; pp. 3–9. (In Russian) [\[CrossRef\]](#)
16. Semenova, I.N.; Kirpichenkov, I.A. Development of the control system for temperature conditions of melting process in the Vanyukov furnace. *Russ. J. Non-Ferr. Met.* **2009**, *5*, 59–62.
17. Zhang, H.L.; Zhou, C.Q.; Bing, W.U.; Chen, Y.M. Numerical simulation of multiphase flow in a Vanyukov furnace. *J. South. Afr. Inst. Min. Metall.* **2015**, *115*, 457–463. [\[CrossRef\]](#)
18. Lisienko, V.G.; Malikov, G.K.; Morozov, M.V.; Belyaev, V.V.; Kirsanov, V.A. Modeling heat-and-mass exchange processes in the Vanyukov furnace in specific operational conditions. *Russ. J. Non-Ferr. Met.* **2012**, *53*, 272–278. [\[CrossRef\]](#)
19. Bazhin, V.Y.; Masko, O.N.; Martynov, S.A. Automatic burden balance monitoring and control in the production of metallurgical silicon. *Tsvetnye Met.* **2023**, *4*, 53–60. [\[CrossRef\]](#)
20. Tercan, H.; Meisen, T. Machine learning and deep learning based predictive quality in manufacturing: A systematic review. *J. Intell. Manuf.* **2022**, *33*, 1879–1905. [\[CrossRef\]](#)
21. Dalzochio, J.; Kunst, R.; Pignaton, E.; Binotto, A.; Sanyal, S.; Favilla, J.; Barbosa, J. Machine learning and reasoning for predictive maintenance in industry 4.0: Current status and challenges. *Comput. Ind.* **2020**, *123*, 103298. [\[CrossRef\]](#)
22. Platonov, O.I.; Tsemekhman, L.S. Potential copper plant Vanyukov furnace gas desulphurization capacity. *Tsvetnye Met.* **2021**, *2021*, 1–19. [\[CrossRef\]](#)
23. Ozerov, S.S.; Tsybulov, L.B.; Eroshevich, S.Y.; Gritskikh, V.B. Looking at the changing composition of blister copper obtained through continuous converting. *Tsvetnye Met.* **2020**, 64–69. [\[CrossRef\]](#)
24. Fedorova, E.R.; Trifonova, M.E.; Mansurova, O.K. Red mud thickener statistical model in MATLAB system identification toolbox. *J. Phys. Conf. Ser.* **2019**, *1333*, 032019. [\[CrossRef\]](#)
25. Utekhin, G. Use of statistical techniques in quality management systems. In Proceedings of the 8 International Conference Reliability and Statistics in Transportation and Communication–2008, Riga, Latvia, 17–20 October 2018; pp. 329–334.
26. Shklyarskiy, Y.E.; Skamyin, A.N.; Jiménez Carrizosa, M. Energy Efficiency in the Mineral Resources and Raw Materials Complex. *J. Min. Inst.* **2023**, *261*, 323–324.
27. Yun, J.P.; Shin, W.C.; Koo, G.; Kim, M.S.; Lee, C.; Lee, S.J. Automated defect inspection system for metal surfaces based on deep learning and data augmentation. *J. Manuf. Syst.* **2020**, *55*, 317–324. [\[CrossRef\]](#)
28. Goldman, C.V.; Baltaxe, M.; Chakraborty, D.; Arinez, J. Explaining learning models in manufacturing processes. *Procedia Comput. Sci.* **2021**, *180*, 259–268. [\[CrossRef\]](#)
29. Iwana, B.K.; Uchida, S. An empirical survey of data augmentation for time series classification with neural networks. *PLoS ONE* **2021**, *16*, e0254841. [\[CrossRef\]](#)
30. Schmitt, J.; Boenig, J.; Borggraeve, T.; Beiting, G.; Deuse, J. Predictive model-based quality inspection using machine learning and edge cloud Computing. *Adv. Eng. Inform.* **2020**, *45*, 101101. [\[CrossRef\]](#)
31. Wu, X.; Zhu, X.; Wu, G.-Q.; Ding, W. Data Mining with Big Data. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 97–107. [\[CrossRef\]](#)
32. Rao, J.N.; Ramesh, M. A Review on Data Mining & Big Data, Machine Learning Techniques. *Int. J. Recent Technol. Eng. (IJRTE)* **2019**, *7*, 914–916.
33. Machado, G.; Winroth, M.; Carlsson, D.; Almström, P.; Centerholt, V.; Hallin, M. Industry 4.0 readiness in manufacturing companies: Challenges and enablers towards increased digitalization. *CIRP Manuf. Syst. Conf.* **2019**, *81*, 1113–1118. [\[CrossRef\]](#)
34. Kim, A.; Oh, K.; Jung, J.-Y.; Kim, B. Imbalanced classification of manufacturing quality conditions using cost-sensitive decision tree ensembles. *Int. J. Comput. Integr. Manuf.* **2018**, *31*, 701–717. [\[CrossRef\]](#)
35. Serin, G.; Sener, B.; Gudelek, M.U.; Ozbayoglu, A.M.; Unver, H.O. Deep multi-layer perceptron based prediction of energy efficiency and surface quality for milling in the era of sustainability and big data. *Procedia Manuf.* **2020**, *51*, 1166–1177. [\[CrossRef\]](#)
36. Boikov, A.V.; Payor, V.A. The Present Issues of Control Automation for Levitation Metal Melting. *Symmetry* **2022**, *14*, 1968. [\[CrossRef\]](#)
37. Koteleva, N.I.; Khokhlov, S.V.; Frenkel, I.V. Digitalization in Open-Pit Mining: A New Approach in Monitoring and Control of Rock Fragmentation. *Appl. Sci.* **2021**, *11*, 10848. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.