AI-Powered decision support system for early defect prediction in cast-iron products

Stefano Dettori, Antonella Zaccara, Valentina Colla – Scuola Superiore Sant'Anna, Italy
Laura Forlani, Gianpietro Bontempi – Fonderia di Torbole s.r.l., Italy

Cast-iron disc brake production requires tight control of metallurgical and process parameters, as defects such as reduced mechanical strength, altered natural frequency, cementite formation, and micro-shrinkage porosity are common in ferrous foundries. These defects originate during liquid metal processing but are only detected downstream and cause significant waste and rework costs. This work presents a Decision Support System integrating machine learning for predictive quality control in cast-iron manufacturing, covering cupola melting, secondary refining, and inoculation during mould filling. The DSS uses real-time data, in terms of melt chemistry, thermal analysis, pouring temperature, inoculant flow, and filling time-to feed models trained on historical data, allowing forecasting defect occurrence 3-4 hours in advance. This horizon enables engineers to adjust melt chemistry or inoculation before non-conforming parts are cast. Validated on industrial production data, the system shows reliable accuracy across defect categories, demonstrating a replicable, Artificial Intelligence-driven approach to foundry digital transformation.

KEYWORDS: CEMENTITE – MICRO SHRINKAGE – ARTIFICIAL INTELLIGENCE– DECISION SUPPORT SYSTEM – DEFECT FORECASTING – CAST IRON MANUFACTURING – PRODUCT QUALITY

INTRODUCTION

The production of high-quality cast components is a key objective in modern foundries, where growing demands for product reliability, process efficiency and sustainability require rigorous control of production operations. Casting defects, such as micro-shrinkages, porosity, and mechanical property changes, can significantly compromise mechanical performance and lifetime of cast products, whilst increasing production costs due to raw material and energy consumption, rework operations, waste production and delivery delays. Even a relatively low defect rate can result in significant economic losses due to the high material, energy and operative costs associated with all the stages related to cast-iron manufacturing. Developing innovative tools capable of predicting defects, while optimizing the production process, is of great interest to manufacturing industries, particularly as they are simultaneously facing the challenges of the green transition and digitalization and need to ensure both product quality and economic sustainability.

The foundry industry presents substantial opportunities for the adoption of Artificial Intelligence (AI) and Machine Learning (ML) technologies across production management, quality assurance, and defect monitoring. In particular, the prediction of defects in cast components is a key AI application, enabling manufacturers to move from 'post-production' quality control, with defects identified in the final stages of production during quality checks, towards timely corrective actions, which allows to adjust the process at an early stage, with economic and environmental benefits. ML models can leverage large volumes of process and operational data for capturing the complex relationships between input parameters and the occurrence of defects [1-3]. In this context, in the recent years main research activities are devoted to applying AI solutions to forecast defect in iron and steel-cast products. An example is the work of Chen and Kaufmann [4], which demonstrated how foundry production data allows to develop data-driven models for predicting casting surface defects. To establish a robust benchmark, six ML regression algorithms were tested and compared: Random Forest (RF) regression, Ridge Regression, Extremely Randomized Trees Regression, XGBoost, CatBoost, Gradient Boosting Regression. SHAP analysis, used to provide information on feature importance, confirmed the ML potential of ML for this type of application, providing results already known in literature and foundry environments. BramahHazela et al. [5] tested several supervised algorithms to predicts the micro shrinkage and ultimate tensile strength using ML classifier and regression method. Pastor-López et al. [6] proposed a

technique for detecting surface defects based on a segmentation method that identify potentially defective areas on the cast part and, subsequently, using the Best Crossing Line Profile technique to classify areas as “correct” or as different types of defects. Similarly, the effectiveness of data-driven models was demonstrated also for forecasting defects and estimating mechanical properties of steel products. Millner et al. [7] compared the application of Feed Forward Neural Network (FFNN) with other modelling techniques (Semi-empirical model, Symbolic regression) for predicting tensile strength of steel coils. FFNN-based models provided the best results, despite the limitation due to few processing parameters used as input variables. Boto et al. [8] focused on improving the steelmaking process and reducing the number of surface and sub-surface defects in micro-alloyed steel products. In particular, the work aimed at identifying indicators related to process performance through the development of ensemble models and NN approaches. Vijaykumar and Reddy [2] proposed a literature review on ML applications for defect forecasting and process optimization in foundry, where they highlighted the high potential of AI solutions for significantly improving defect detection, reducing production costs and enhancing product quality. Advanced algorithms, such as FFNN and ensemble learning methods, can efficiently analyze large datasets, detect defects at an early stage and optimize process parameters. This review also highlights the challenges for the widespread application of ML models in foundries, such as lack of high-quality data, difficult model interpretability and resistance of plant staff in using AI-based technologies. Therefore, a multidisciplinary approach focusing on real-world applications is required to develop robust models and enhance workforce skills.

The work presented in this paper moves a step in this direction with a Decision Support System (DSS) integrating ML models for predictive quality control in a cast-iron foundry. The objective of the models is to forecast and prevent defect generation by exploiting process data collected prior to casting, including the chemical and thermal analyses of molten cast iron together with information on the inoculant flow, pouring temperature and casting duration. The proposed approach is validated on two cast iron grades and targets the prediction of two critical metallurgical defects, micro-shrinkage and cementite formation, and the natural frequency of the brake disc products, an important quality parameter. The presented work was developed in the context of the EU-funded project ALCHIMIA, which aimed at enhancing sustainability and circularity of steelmaking and metallurgical processes by optimizing raw material selection and production operations. The project approach was validated through 2 industrial use cases: a major European steel producer (participating with 3 industrial plants) and an Italian foundry manufacturing automotive components. Its methodology combined first-principles and artificial intelligence models within a human-centric platform based on Federated and Continual Learning [9], developed using a multidisciplinary approach integrating metallurgical expertise, IT skills and knowledge in the social sciences, which are essential to train and engage plant staff in the adoption of the AI-based solution. The resulting DSS, deployed at the participating industrial sites, provide real-time decision support for material selection and process optimization, supporting plant managers and operators in improving resource efficiency while balancing environmental performance, product quality, and industrial competitiveness. Here the foundry use case is presented.

2. PLANT AND PROCESS DESCRIPTION

Fonderia di Torbole (FdT) is an automotive component manufacturer specialized in the production of brake discs, brake drums, castings, and machined parts. The charge mix of FdT’s cupola furnace primarily consists of pig iron, returns, steel scrap, and coke, which are melted in a cupola furnace. The molten metal is then transferred to two induction holding furnaces: most of the material is sent directly to the pouring system for the moulding process, while a smaller fraction intended for higher-quality products undergoes chemical composition adjustment in two tandem coreless induction furnaces before pouring. After casting, the products typically require approximately three hours of cooling before final quality inspections can be performed. When rapid process feedback is needed, an accelerated cooling procedure can be applied to a single mould, reducing the inspection time to about one hour.

The objective of this study is to predict the final quality of cast products by exploiting process measurements before moulding, enabling early process adjustments and preventing the production of non-conforming parts. Early defect prediction reduces the delay associated with conventional quality control, leading to lower energy and raw material consumption, and improved process efficiency avoiding rework procedures. To achieve this, ML models were developed using process data routinely collected during production. The input dataset

includes thermal and chemical analyses of the molten iron performed at selected process locations, together with casting temperature and inoculant addition data. The sampling points are chosen according to the cast iron grade and their correlation with the final product quality. The output variables consist of quality control measurements and the occurrence of two critical casting defects, namely cementite formation and porosity. The proposed modelling approach was evaluated on two cast iron grades: high-carbon grey cast iron (GG15HC) and standard grey cast iron (GG25). For GG15HC, thermal and chemical analyses were performed at sampling points 2 and 3, whereas for GG25 the corresponding analyses were carried out at points 1 and 3.

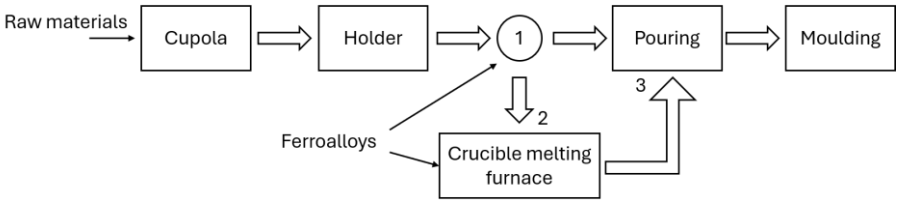


Fig. 1 – Process scheme

More in details, the chemical analysis measures the main chemical contents (C, Si, Mn, P, S, Cr, Cu, Mo, Sn, Ti, Al, Pb, Ni, V, Nb) including the Carbon equivalent and the degree of eutectic saturation. The thermal analysis includes temperatures and timing of the liquidus, the start of the eutectic, min and maximum eutectic, and solidus. Additionally, the main parameters of the thermal derivative curve are also measured in terms of recalescence, PAE, VPS.

3. METHODS

This work presents a set of models that solves two different predictive tasks: (i) the regression of the natural frequency of brake disc products as a continuous function of the available process measurements, and (ii) the binary classification of defect occurrence, complicated by a severe class imbalance, as defective observations represent a small fraction of the total dataset [10]. For both tasks, 3 common families of supervised learning algorithms were evaluated and compared: regularized linear models, RFs, and XGBoost. These methods were selected to represent complementary modelling paradigms with different underlying assumptions and different tradeoffs. Linear models provide an interpretable baseline and are well suited to capturing additive, monotonic relationships between metallurgical variables and the target property or defect occurrence. RFs and XGBoost, by contrast are non-parametric ensemble methods capable of capturing non-linear and interacting nature of the phenomena. In addition, for the regression task, FFNNs were good candidates to model possible nonlinear phenomena between the variables of interest. For defect classification task, given the strong imbalanced distribution of nominal and defective products, neural-based classification algorithms tend to be biased toward the majority class and are unable to produce stable and trustable results.

More in detail, ordinary least-square linear regression estimates model coefficients by minimizing the Mean Square Error (MSE) of the regression task, leading to unstable results and overfitting when input variables may be empirically correlated (e.g. Carbon equivalent and C contents). For this reason, regularized variants were used for the regression task, introducing a penalty term on the magnitude of the regression coefficients. Ridge regression, Lasso regression, and Elastic Net models [11] were selected for the scope. For classification task, the direct analogue is logistic regression, which models the log-odds of an observation belonging to the defective class as a linear combination of the input variables, regularized with the same mentioned penalty structure. Class imbalance can be addressed by weighing the contribution of each observation to the loss function inversely proportional to its class frequency, increasing the relative penalty for misclassifying defective (minority-class) observations during training. RF is an ensemble learning method that builds a large number of decision trees and aggregates their individual prediction to produce a final decision [12]. For the regression task, the final prediction is calculated by averaging the output of all trees, while for classification the final decision is calculated via majority voting. Class re-weighting can be applied during training to counteract the imbalance between defective and nominal products. XGBoost is a scalable and regularized implementation of the gradient boosting framework [13], which unlike RFs, it builds trees sequentially, where each tree is trained to correct the residual errors of the ensemble constructed so far, by performing a gradient descent of the loss

function. Also in this case, the class imbalance can be addressed by increasing the relative importance assigned to the minority class (defects) during training. XGBoost models also support early stopping algorithm, that allows to monitor the model performance on the validation dataset, allowing to stop the training session once the performance starts decreasing, avoiding the overfitting on the training dataset.
FFNNs leverage backpropagation for learning hidden nonlinear patterns in the input data. In this work, an encoder-shaped architecture was designed, in which the number of units in each successive sequentially connected layer is constrained to diminish, to model the hierarchy of information contents in the hidden layer, while compressing the data and identifying generalized transformations. The dropout regularization is used to increase generalization and reduce overfitting. The network is trained to minimize the MSE on the training dataset, leveraging the early stopping algorithm to avoid overfitting. As mentioned before, FFNNs were unsuitable for defect classification in this work, considering the severe class imbalance characterizing the datasets. FFNNs typically contain substantially more trainable parameters than the other proposed methods and generally require a correspondingly larger and more balanced set of examples per class to learn a robust and generalized decision boundary.

4. MODELLING EXPERIMENTS AND RESULTS

Data processing and experimental setup

The data collection focused on the most critical products in terms of defect occurrence frequency. Four main products have been identified, three for GG25 cast iron (product numbers: 1692, 1693, 21000238) and one for GG15HC (product number: 1740). Products 1692 is mainly affected by cementite issues, while 1693 and 21000238 are affected by micro-shrinks defects. Product 1740 is mainly affected by micro-shrinks defects. All the products are tested in terms of natural frequency, but 21000238 dataset is characterized by a low number of samples, so a model for this target is not considered. The raw data collected is characterized by the issues typical of industrial data, which affect the overall data quality: outliers [14] and missing values. Here outliers are always fixed by substitution with mean values, while missing values, numerous in the collected data, are treated by eliminating the related row to avoid poisoning the data distribution. Table 1 reports the datasets available for all the products, after the data pre-processing and cleaning.

Tab. 1 – characteristics of the dataset collected.

DATASET COLLECTION					
Cast iron	**Product**	**Target**	**Number of samples**	**Number of defects / distribution**	**Defects [%]**
GG15HC	1740	μshrinks	1239	62	5.0
		Frequency	1245	Range: [690, 749] [Hz], mean: 717 [Hz], std: 7.28 [Hz]	-
GG25	1692	Cementite	2009	22	1.1
		Frequency	2018	Range: [2148, 2244] [Hz], mean: 2198 [Hz], std: 13.9 [Hz]	-
	1693	μshrinks	729	24	3.3
		Frequency	734	Range: [2381, 2477] [Hz], mean: 2427 [Hz], std: 16.7 [Hz]	-
	21000238	μshrinks	236	23	9.8

Once data cleaning is completed, an additional step of data preprocessing was carried out to extract additional information from raw data. In particular, the following features were identified: (i) the inoculant ratio, calculated as the ration between the inoculant flow and the filling duration, (ii) the over temperature, calculated as the difference between the casting and the liquidus temperatures, and (iii) the solidification interval, calculated as the difference between liquidus and solidus temperatures. Additionally, a set of ad-hoc heuristics was used to extract information from input data: the excess of manganese f_1, a heuristic carbide stabilization factor f_2, and f_2 balanced with the carbon (f_3):

$$f_1 = [\%Mn] - 1.7[\%S] - 0.3 \quad [1]$$

$$f_2 = [\%Cr] + 2.5[\%V] + 0.5[\%Mo] + 0{,}3f_1 - [\%Si] - 0.5[\%Al] - 0.35[\%Ni] - 0.3[\%Cu] \quad [2]$$

$$f_3 = [\%C] + f_2 \quad [3]$$

The final dataset is then divided into three portions needed for the training, the validation and the test. More in detail, the training is used for tuning model parameters, the validation set is used for searching the best model architecture and for early stopping algorithms (XGBoost and NNs). For regression tasks, training, validation and test sets have respectively 60%, 20%, and 20% of the overall number of samples. For classification tasks, since the datasets are strongly unbalanced, the percentage of samples are divided respectively into 60, 15, and 25 percent, to have a relevant number of defective samples on the test dataset and verify the generalization capacity of the final model. For the classification dataset, the split is performed through stratification, to approximately preserve the relative class frequencies.

All the models were developed and trained in python environment, using Scikit-learn, XGBoost and Keras libraries. The predictive performance of each candidate model depends substantially on a set of hyperparameters that govern the model capacity. These hyperparameters were tuned through an automated search procedure, evaluated against a held-out validation set, kept separate from the final test used for the final model assessment. The hyperparameters are specifically optimized for each model family. For regularized linear models: (i) regularization type (Ridge, Lasso, Elastic net), (ii) regularization penalty $\in [10^{-3} \quad 10]$, (iii) L1 ratio (the balance between L1 and L2 penalty components, only for elastic net) $\in [0.1 \quad 0.9]$. The class weight (only for classification task) is automatically calculated by the library, in function of the real imbalance of the dataset. For RFs: (i) the number of trees in the ensemble $\in [50 \quad 400]$, (ii) the maximum tree depth $\in \{3,\ 5,\ 7,\ 10,\ 'None'\}$, (iii) the minimum samples per leaf $\in [1 \quad 10]$, (iv) the maximum features per split $\in \{'sqrt',\ 'log2',\ 0.5,\ 0.8\}$. The class weight (only for classification task) is automatically calculated by the library, in function of the real imbalance of the dataset. For XGBoost, the hyperparameters tuned were: (i) the number of boosting rounds (number of sequential trees added to the ensemble) $n_t \in [50 \quad 400]$, (ii) the maximum tree depth $d_t \in [2 \quad 6]$, (iii) the learning rate $lr_t \in [0.01 \quad 0.3]$, (iv) the subsample ratio $subsample \in [0.5 \quad 1]$, (v) the column subsample ratio $colsample \in [0.5 \quad 1]$, and (vi) the class weight (for classification tasks) $w_{xgb} \in [0.3 \quad 3]$. For FFNNs, the hyperparameters optimized were: (i) the number of hidden layers $L \in [1 \quad 3]$, (ii) the number of neurons per layer $N \in [8 \quad 128]$, (iii) the dropout (use $\in \{True,\ False\}$, and rate $dr \in [0.1 \quad 0.5]$), (iv) the learning rate $lr_{nn} \in [10^{-3} \quad 10^{-2}]$, and (v) the batch size $bs \in \{16,\ 32,\ 64\}$.

Variable selection, model selection, and hyperparameter tuning were performed jointly, rather than as separate sequential steps, motivated by several characteristics of the tasks and the datasets. Both the regression and classification tasks are inherently complex and must be addressed with a limited number of available samples. In addition, all the target variables are heavily imbalanced, though in different ways. The regression target (brake disc natural frequency) follows an approximately Gaussian distribution, with most observations concentrated near the mean and comparatively few in the tails, the latter of which are important for understanding the final quality of the disc, whether it will be within the minimum and maximum frequency limits. Defect classification targets are characterized by a strong class imbalance, with defective samples representing only a small fraction of the dataset. Furthermore, the number of candidate predictor variables is large relative to the sample size, requiring a dedicated selection step to retain only the most descriptive subset. Critically, different model families are expected to differ in their capacity to discriminate defective from nominal samples, and, more specifically, in their ability to construct effective decision boundaries from different subsets of predictors; consequently, no unique feature subset can be assumed to be uniformly optimal across all candidate model types. For this reason, variable selection, model selection, and hyperparameter optimization were treated as a single, jointly optimized problem rather than as independent sequential stages. In this context, genetic algorithms are a good search paradigm for solving complex optimization problems. In this work, the implementation of NSGA-II algorithm available in the Optuna library (python environment) has been used for the scope. For regression task, the objective was minimizing the Root Mean Square Error (RMSE) on the validation dataset. For defect classification task, the objective was to obtain a model able to minimize the number of False Positive (FP) and maximize the True Positive, a multi-objective optimization. In the resulting

Pareto front, the best solution was selected to maximize the difference between TP and FP, to find a model able to minimize production and environmental costs. For the GA search, the population is set to 50 individuals, the number of generations is set to 2000, for crossover and mutation the default library parameters are used.

Modelling results

The optimal model search, including variable selection, model type and hyperparameters, has been performed through the protocol presented in the previous subsection. For each task, the best models were selected according to the results on the validation dataset. The final results were evaluated on the test dataset, with the scope of assessing the generalization capacity of the best models and possibly identifying weaknesses of the approach followed. For regression task, the final assessment of the model is carried out by calculating the Normalized RMSE, evaluated by normalizing the RMSE on the test set by the standard deviation of the target. For classification tasks, the final evaluation is carried out by analysing FP, TP, the precision and F1 scores, in order to understand the capability of the model of recognizing defects, while also minimizing false positives, under strong unbalanced conditions. Table 2 and 3 respectively report the test results for the defect forecasting and regression tasks. For regression tasks, NN are often chosen as the best model. Results show that one layer with numerous neurons is sufficient for solving the regression task, helped by dropout used for improving the regularization and the generalization capacity of the model. Only for the product 1692 XGBoost regression resulted the best modelling solution. For defect classification tasks, XGBoost resulted the best modelling solution except for product 21000238, where RF perfectly recognized all the defects on the test dataset, with zero FP. In general, the results show that, for both classification and regression tasks, linear models were never selected as the best performing model, highlighting the complexity of the phenomena.

Tab. 2 – results for defect forecasting and classification.

CLASSIFICATION RESULTS							
Cast iron	**Product**	**Defect**	**Best model**	**Hyperparameters**	**FP, TP**	**Prec.**	**F1 score**
GG15HC	1740	μshrinks	XGBoost	n_t: 300, d_t: 5, lr_t: 0.141, $subsample$: 0.687, $colsample$: 0.882, w_{xgb}: 2.68	1, 8	0.88	0.64
GG25	1692	Cementite	XGBoost	n_t: 75, d_t: 4, lr_t: 0.0104, $subsample$: 0.734, $colsample$: 0.528, w_{xgb}: 3	0, 2	1.0	0.5
GG25	1693	μshrinks	XGBoost	n_t: 100, d_t: 4, lr_t: 0.0128, $subsample$: 0.761, $colsample$: 0.510, w_{xgb}: 3	0, 4	1.0	0.8
GG25	21000238	μshrinks	RF	n_t: 200, d_t: 7, min_samples_leaf: 1, max_features: 'log2'	0, 6	1.0	1.0

Tab. 3 - modelling results for natural frequency forecasting.

REGRESSION MODELLING RESULTS				
Cast iron	**Product**	**Best model**	**Hyperparameters**	**NRMSE**
GG15HC	1740	NN	L: 1, N: 128, use dropout: True, dr: 0.123, lr_{nn}: 0.0021, bs: 32	0.7341
GG25	1692	XGBoost	n_t: 200, d_t: 6, lr_t: 0.0361, $subsample$: 0.621, $colsample$: 0.912	0.7572
GG25	1693	NN	L: 1, N: 108, use dropout: True, dr: 0.112, lr_{nn}: 0.0021, bs: 64	0.7614

5. THE DECISION SUPPORT SYSTEM

In this section, we present the DSS developed to assist process operators in estimating the likelihood of production defects based on current process parameters. The architecture of the DSS is depicted in Figure 2. In particular, the DSS is composed of four main components: (i) a Graphical User Interface (GUI), (ii) a web service deploying the ML models, (iii) a repository containing the historical trained models, and (iv) a message

broker. The GUI, partially shown in Figure 3, reports the main information on the current production for the two production lines (1 and 2), the current chemical and thermal analysis and the last 1-hour analyses, and the current moulding settings. Also, the GUI allows operators to infer the ML models to forecast the quality check according to the last available measurements. The GUI was developed in python environment through the Dash Plotly library. The GUI receives the last process measurements through a message broker developed through Kafka, which manages the processed data to be available for the software. Additionally, the quality check forecasted through the ML models are stored and organized in a SQLite database. The models presented in this work, the core of the DSS, are deployed and inferred through a web service developed in python environment through FastAPI, Scikit learn and Keras libraries. The models' repository is deployed through Min.IO, a web-service repository architecture that allows storing and organizing objects, such as the trained models. The DSS was deployed on the plant and is currently used by plant operators, to evaluate both ergonomics and performance of the software. Feedback from operators and plant managers was used to improve the performance of the models and to add features to the software.

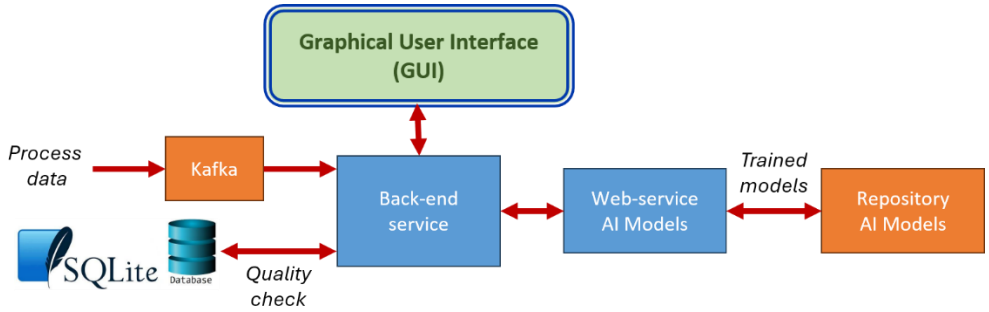
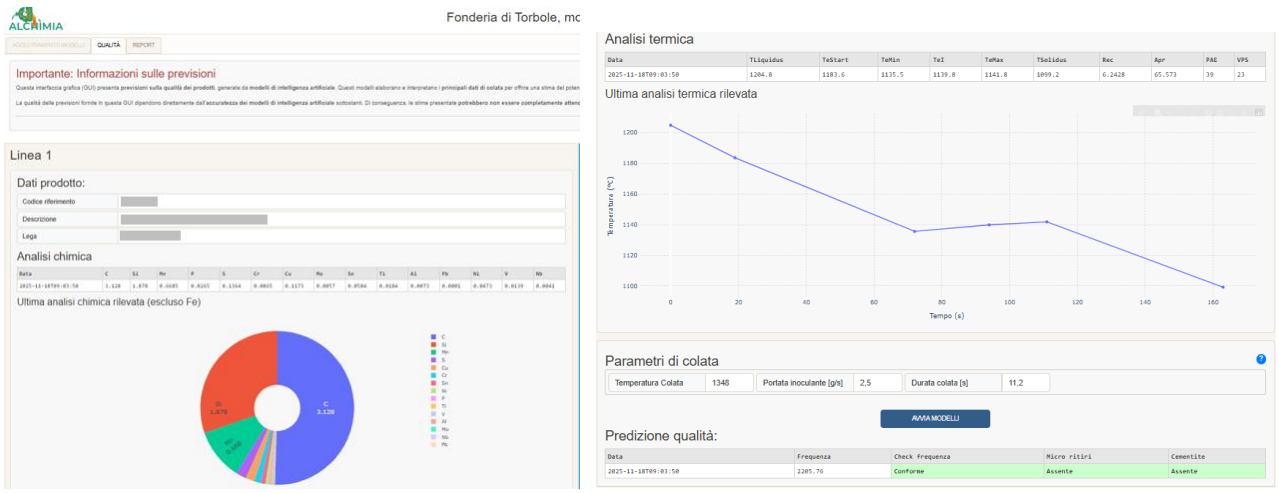


Fig.2 – DSS software architecture.



(a) (b)

Fig.3 – The DSS: (a) main information of the product and chemical analysis, (b) thermal analysis and quality forecasting.

6. CONCLUSIONS

This work presents a DSS that integrates a set of ML models trained to forecast product quality in cast iron foundries. These models allow to anticipate quality control by approximately 2-3 hours, allowing anticipating corrective actions on the forecasted defective production lines. A set of models was trained by exploiting collected process and product data to forecast the natural frequency of brake disc products, and the occurrence of micro shrinkage and cementite defects for two types of cast iron qualities. Regularized linear models, RF, XGBoost and FFNN were compared and validated for identifying the best performing model. The trained models were deployed within a software architecture, providing a GUI to process operators for supporting their decision making and it is currently used on a foundry.

This research work shows metallurgical defects and quality forecasting in foundry products is a complex task

with no single universal solution. The modelling effectiveness depends on the quality and quantity of the data, and in many cases on the imbalance in the distribution. However, this paper demonstrates that there are a set of effective techniques that allow for the creation of working solutions that can gradually be made more robust through data collection and model improvement. Future work will focus on improving the accuracy of the current modelling portfolio and targeting additional products types and materials.

AKNOWLEDGEMENTS

The work presented in this paper has been developed within the project entitled "*Data and decentralized Artificial intelligence for a competitive and green European metallurgy industry*" (Alchimia, G.A. 101070046) that received funding from the Horizon Europe research and innovation program of the European Union, which is gratefully acknowledged. The sole responsibility for the issues treated in the present paper lies with the authors; the Commission is not responsible for any use that may be made of the information contained therein.

REFERENCES

[1] Alamuru S., Reddy G.S., Raju M.V.J. Artificial intelligence and machine learning for defect detection in castings. Journal of Physics: Conference Series, 6th Int. Conf. Advancements in Materials and Manufacturing. 2024; 2837. https://doi.org/10.1088/1742-6596/2837/1/012079

[2] Vijaykumar H.K., Reddy N.R. Review on applications of machine learning for defect prediction and optimization in foundry processes. Manufacturing Technology Today. 2025; 24(1-2).

[3] Pribulová A, Bartošová M, Baricová D. Quality Control in Foundry–Analysis of Casting Defects. 2013.

[4] Chen S, Kaufmann T. Development of Data-Driven Machine Learning Models for the Prediction of Casting Surface Defects. Metals, 12(1), 2022. https://doi.org/10.3390/met12010001

[5] BramahHazela, Hymavathi J, Kumar TR, Kavitha S, Deepa D, Lalar S, et al. Machine Learning: Supervised Algorithms to Determine the Defect in High-Precision Foundry Operation. Journal of Nanomaterials. 2022. https://doi.org/10.1155/2022/1732441.

[6] Pastor-López I, de-la-Peña-Sordo J, Santos I, Bringas P.G. Surface defect categorization of imperfections in high precision automotive iron foundries using best crossing line profile. .Proc. 10th IEEE Conf. Ind. Electr. Appl., ICIEA 2015. 339-344 2015. https://doi.org/10.1109/ICIEA.2015.7334136

[7] Millner G, Kronberger G, Mücke M, Romaner L, Scheiber D. Comparison of semi-empirical models, symbolic regression, and machine learning approaches for prediction of tensile strength in steels. Int. Journal of Engineering Science. 2025; 212. https://doi.org/10.1016/j.ijengsci.2025.104247

[8] Boto F, Murua M, Gutierrez T, Casado S, Carrillo A, Arteaga A. Data Driven Performance Prediction in Steel Making. Metals. 2022; 12(2), 172. https://doi.org/10.3390/met12020172

[9] Jaehong Y, Wonyong J, Giwoong L, Eunho Y, Sung Ju H. Federated Continual Learning with Weighted Inter-client Transfer. Proc. Machine Learning Research . 2021, 139, 12073-12086.

[10] Vannucci M., Colla V. Classification of unbalanced datasets and detection of rare events in industry: Issues and solutions, Communications in Computer and Information Science, 629, 337-351, 2016. https://doi.org/10.1007/978-3-319-44188-7_26

[11] Gillariose J., Joseph, J., Chesneau, C. Lasso and Ridge regression: a comprehensive review of applications and developments in machine learning. International Journal of Data Science and Analytics. 2026; 21(1): 7. https://doi.org/10.1007/s41060-025-00957-y

[12] Jaiswal J.K., Samikannu R. Application of random forest algorithm on feature subset selection and classification and regression. Proc. 2nd World Congress on Computing and Communication Technologies, WCCCT 2017, February 2-4, 2017; 65-68. https://doi.org/10.1109/WCCCT.2016.25

[13] Zhang PaJYaSY. Research and application of XGBoost in imbalanced data. International Journal of Distributed Sensor Networks. 2022; 18(6): 15501329221106935.

[14] Cateni S., Colla V., Nastasi G. A multivariate fuzzy system applied for outliers detection, Journal of Intelligent and Fuzzy Systems, 24(4), 2013 889-903. https://doi.org/10.3233/IFS-2012-0607