The Mechanical Properties of Ultra-High Performance Concrete

Farzad Rezazadeh P.[1,*] , Amin Abrishambaf[2], Axel Dürrbaum[3] , Gregor Zimmermannb[4] ,Andreas Kroll [5]

[1] Department of Measurement and Control, University of Kassel, Mönchebergstr. 7, Kassel, 34125, Hessen, Germany

[2] German Technologies & Engineering Concepts - G. tecz Engineering GmbH, Kassel, 34123, Hessen, Germany

[3] Department of Measurement and Control, University of Kassel, Mönchebergstr. 7, Kassel, 34125, Hessen, Germany

[4] German Technologies & Engineering Concepts - G. tecz Engineering GmbH, Kassel, 34123, Hessen, Germany

[5] Department of Measurement and Control, University of Kassel, Mönchebergstr. 7, Kassel, 34125, Hessen, Germany

***Corresponding author, E-mail: farzad .rezazadeh@mrt .uni-kassel .de**

Abstract

Ultra-High Performance Concrete (UHPC) surpasses conventional concrete in performance. However, producing UHPC with consistent mechanical properties, even with an identical recipe, remains challenging. The quality of UHPC can be significantly influenced by material quality, environmental factors, and human intervention during large-scale production. This study, for the first time, takes a holistic view of the UHPC manufacturing process to investigate the impact of material quality, environmental conditions, measurement errors, and mixing and curing conditions on the final mechanical properties. This comprehensive approach to the UHPC manufacturing process presents two challenges. First, there is no publicly available dataset for this research. Therefore, 150 experiments were conducted, measuring both Compressive and Flexural Strength after 28 days of curing, resulting in two experimental datasets for this study. Second, this wide view increases data dimensionality and, coupled with the high cost of UHPC experiments, yields sparse data. Traditional evolutionary algorithms, while effective in feature selection, struggle in high-dimensional, small-sample data. To address this, an Informed Non- dominated Sorting Genetic Algorithm II (I-NSGA-II) is developed in this study, incorporating domain-specific knowledge to enhance prediction accuracy and solution stability. Comparative evaluations using different machine learning algorithms on the two experimental datasets and a dataset generated by a test function demonstrated the significant superiority of I-NSGA-II over the classic NSGA-II. Finally, the significance of each studied parameter on the mechanical behavior of UHPC is discussed.

Keywords: Ultra-High Performance Concrete; Concrete Manufacturing Process; Ensemble-based Outlier Detection; Multiobjective Feature Selection; Data-Driven Modeling

1. Introduction

Concrete is the second-most consumed resource globally, after water, and cement production, as the cornerstone of any type of concrete, accounts for around 7 % of CO_2 emissions worldwide [1].

Ultra-high performance concrete (UHPC) is an advanced cement-based composite known for its exceptional mechanical strength and durability. It typically contains a high volume of short steel fibers (around 2 % by volume) distributed within a dense matrix with a low water-to-binder ratio, often incorporating silica fume. This composition allows UHPC to exhibit uniaxial tensile hardening behavior, leading to stable microcracks and excellent transport properties even under demanding conditions. These characteristics make UHPC ideal for innovative applications such as bridge construction, structural strengthening, and waterproofing. However, the high cement and silica fume content contribute to increased production costs and a significant CO_2 footprint [2].

Additionally, the production process for UHPC is highly sensitive [3–6], with minor deviations from the recipe or changes in environmental conditions drastically impacting consistency and mechanical behavior, leading to increased waste. To address these issues, the construction industry requires an advanced support system capable of predicting UHPC properties in real time. Such a system would enhance quality control during production, reduce waste, improve product quality, and achieve cost savings.

To date, investigations into the UHPC manufacturing process have typically focused on individual factors rather than a comprehensive analysis of all relevant variables. Consequently, there are no published datasets that encompass the full spectrum of variables affecting UHPC production and quality. In this study, we conduct a systematic investigation of variations arising from all relevant factors around a reference UHPC production condition (Figure1). We examine the impact of raw material properties (such as impurities and particle size distribution), dosing system errors, mixing duration and speed, and environmental conditions (affecting both raw materials and specimen curing) on UHPC quality. Modeling the UHPC production process holistically presents significant challenges due to the complex physical and chemical subprocesses involved. Generating data for a single experimental point requires 28 days, making the process extremely time-consuming and costly. To address this, a two-phase experimental design was implemented in this study to generate 150 data points, optimally covering the input space. However, the high dimensionality and small sample size of the data further complicate modeling efforts. Standard methods, such as sequential feature selection and recursive feature elimination, often struggle to identify nuanced patterns in sparse data, frequently missing critical relationships and becoming trapped in local minima [7,8]. While evolutionary multiobjective feature selection methods can outperform grid search-based approaches, they are less effective when applied to high-dimensional, small-sized datasets.

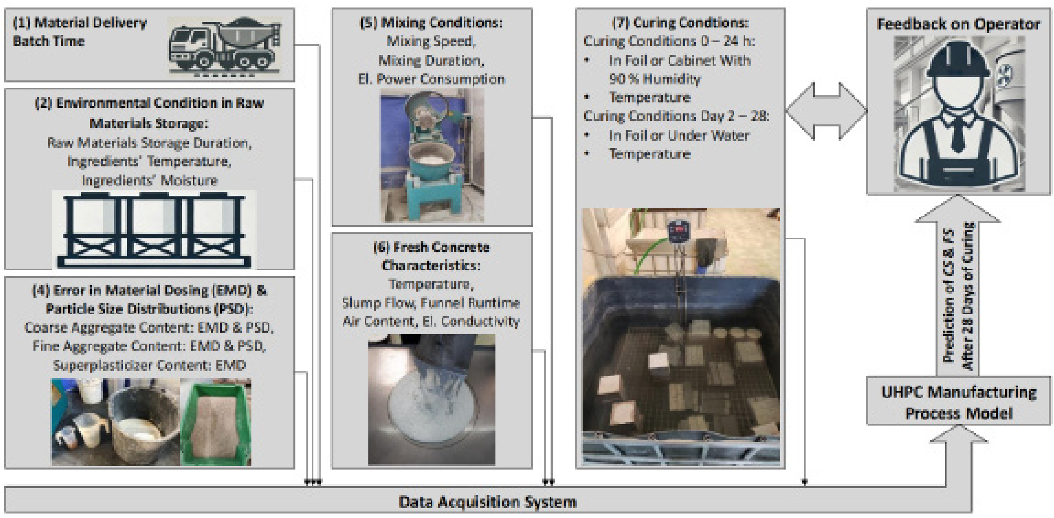


Figure 1: Modeling of Reproducible UHPC with Consistent Properties[1]

1 Ultra-High Performance Concrete Manufacturing Process: Production and testing processes, influencing factors, and key considerations regarding the impact of material quality, measurement errors, and environmental conditions on final UHPC mechanical quality [6]. (El.: Electrical, CS: Compressive Strength, FS: Flexural Strength)

To address these challenges, this paper focuses on dimensionality reduction in multiple steps. By injecting domain knowledge into the classic Non-dominated Sorting Genetic Algorithm II (NSGA-II), prediction results and solution stability are enhanced. The proposed Informed NSGA-II (I-NSGA-II) in this study addresses high dimensionality issues with small sample sizes for evolutionary multiobjective feature selection.

To the authors' knowledge, this is the first study to provide a holistic view of the UHPC manufacturing process by investigating the causal effects of various variables on final product quality consistency from an application perspective. Additionally, this study uniquely addresses evolutionary multiobjective feature selection by incorporating prior knowledge into the Non-dominated Sorting Genetic Algorithm II, thereby enhancing prediction performance and solution stability for high-dimensional datasets with small sample sizes. The contributions of this paper are summarized as follows:

- Present a holistic approach to the UHPC manufacturing process.
- Investigate the impact of material quality, environmental conditions, measurement errors, and mixing and curing conditions on reproducibility of UHPC with consistent mechanical properties, as well as the relation of fresh UHPC characteristics with these mechanical properties.
- Introduce a Human-in-the-Loop Ensemble-based Outlier Detection Method, informed by expert insights, to enhance data quality.

2. Related Work

2.1. Machine Learning Applications in Predicting Concrete Mechanical Properties

The task of predicting concrete compressive strength (CS), particularly its 28-day CS, is typically addressed by two major approaches: traditional empirical models and modern machine learning techniques [11]. One of the earliest empirical methods, known as Abram's Law [12], is:

$$CS = \frac{a_0}{a_1^{w/c}}, \tag{1}$$

where a0 and a1 are empirical constants, and w and c represent the quantities of water and cement, respectively. An advancement of this method is multiple linear regression [13]:

$$CS = a_0 + a_1 \frac{W}{C} + a_2 CA + a_3 FA + C, \tag{2}$$

which incorporates the water-to-cement ratio w/c, amounts of coarse (CA) and fine aggregate (FA), and cement quantity (C). However, these models do not consider all the complex steps in the concrete production process. This makes it hard to accurately predict the properties of advanced concrete types like UHPC [14].

The significance of mix proportion parameters is emphasized by Ozbay et al. [15], yet environmental variables and curing conditions are overlooked. Farzampour [16] highlights certain mix proportion parameters while acknowledging environmental factors during curing; however, a comprehensive analysis of these aspects is lacking, and an in- depth examination of curing conditions is not conducted. The necessity of longer mixing times to achieve optimal homogeneity in UHPC, compared to CC and HPC, is pointed out by Safranek [17], which also cautions against high mixing speeds due to potential thermal effects.

The initial challenge in adopting machine learning techniques for concrete research is the scarcity of comprehensive and reliable datasets. The datasets on Compressive Strength [18] and Slump Flow Test [19], compiled by Yeh from various research sources, are widely used. Despite their extensive utilization [13, 20–28], these datasets exhibit significant shortcomings: limited coverage of input factors across the concrete production process and potential inconsistencies in material quality and production conditions, including mixing and curing processes. These issues compromise the utility of the datasets for both academic and practical applications [6,29].

Nguyen et al. [30] employed the XGBoost model to forecast the CS of UHPC using a dataset of 931 UHPC mix formulations, compiled from laboratory experiments and existing literature, encompassing 17 input variables. The study aimed to enhance precision in UHPC compressive strength estimates and facilitate the development of new UHPC mixtures, reducing both time and costs associated with their creation. Despite these ambitious objectives, Nguyen et al.'s methodology presents notable limitations. The amalgamation of data from varied sources without stringent standardization introduces biases, potentially skewing results. Moreover, the study's exclusive reliance on the XGBoost algorithm and intensive focus on hyperparameter tuning at the expense of feature selection restricts the research scope. Such a narrowed focus overlooks potential insights from a broader spectrum of algorithms and a comprehensive evaluation of input features' relevance. Additionally, the absence of possible uncertainties in material quality, material dosing, mixing, and curing conditions in the analysis undermines the model's practical applicability, given their critical role in determining UHPC's compressive strength.

Designing a pipeline for the concrete production process with sparse data is a topic that has been rarely researched [31–33]. The recently proposed pipeline for modeling concrete production and optimizing concrete mixtures [33] overlooks key aspects of managing high-dimensional datasets, especially the complexities of small datasets with high dimensionality and a comprehensive understanding of the concrete production process, including material quality and curing time. Its approach to data generation – limited to compiling data from various sources – risks introducing redundancy. Moreover, the reliance on a narrow set of tree-based machine learning algorithms may not adequately capture the complexity of the data. The study also lacks crucial preprocessing steps, such as outlier detection and appropriate handling of missing values, opting instead for simple mean imputation. Furthermore, its simplistic training- testing strategy, which relies on a single data split, fails to ensure model reliability across various data partitions.

Despite advancements in predicting concrete strength, significant gaps remain, underscoring the challenges in capturing the full spectrum of variables that influence end-product quality [34]. These gaps encompass limited data coverage, including material quality, measurement errors, and mixing and curing conditions, which lack a holistic view of the production process. These issues can cause discrepancies inUHPC's final quality even with identical recipes [35]. Other challenges include systematic data generation, ineffective feature selection methods struggling with the high- dimensional nature of the data, inadequate training-test strategies [36–45], and a narrow range of explored algorithms [14,46–53].

Addressing these challenges, this work takes a holistic view of the UHPC manufacturing process to investigate the effect of material quality, uncertainties in material dosing and particle size distribution, and mixing and environmental conditions on final mechanical UHPC properties. This is achieved by introducing an Automatic Modeling Pipeline that includes advanced data collection, data cleaning, and sophisticated feature selection techniques designed to fully capture the complexities of the UHPC manufacturing process. Employing a diverse array of 10 machine learning algorithms and adopting a Leave-One-Out Cross Validation (LOOCV) [54] training-test strategy with multiple initializations, the proposed methodology ensures the reliability of UHPC mechanical properties predictions.

2.2. Evolutionary Multiobjective Feature Selection

In the domain of machine learning, especially with high-dimensional and small datasets, feature selection (FS) is a pivotal process aimed at enhancing model interpretability, reducing overfitting, and improving prediction performance by eliminating irrelevant or redundant features from the dataset [55]. Traditionally, FS methods have been categorized into three approaches: filter, wrapper, and embedded [7]. However, these methods often suffer from a high possibility of selecting redundant features, experiencing nesting effects, and/or falling into local optima, especially in complex datasets with high dimensionality [7,8].

Feature selection faces three main challenges [8]: the exponential growth of the search space with the number of features; possibly complex interactions among features, where redundancy and complementarity significantly impact prediction performance; and the inherent multiobjective nature of feature selection, which requires a trade-off between maximizing prediction performance and minimizing feature count, which are often conflicting objectives. Significant advancements have been made with the incorporation of evolutionary multiobjective optimization (EMO) techniques into feature selection. However, due to the inherent randomness of the evolutionary process, the outcomes are often unstable, especially when dealing with complex datasets and conditions of data sparsity.

Despite the widespread adoption of multiobjective feature selection (MOFS) due to its capability in global optimization, which eliminates the need for prior assumptions in these algorithms, and their adeptness at handling high- dimensional cases, they typically require relatively large datasets. This necessity, coupled with the inherent randomness of evolutionary processes, contributes to the instability of results obtained from EMO, especially in complex datasets with high dimensionality. As a result, they have been applied primarily to high-dimensional datasets but with large data sizes, especially in classification tasks [56–60]. These challenges are more pronounced in scenarios involving high-dimensional datasets with limited samples, highlighting the need for innovative approaches.

To address the above challenges in high-dimensional datasets with small sizes, the integration of prior knowledge in the process of feature selection is necessary [61]. Evolutionary multiobjective feature selection methodologies distinguish themselves through various core design elements, including solution representation, evaluation functions, initialization strategies, offspring generation methods, environmental selection techniques, and decision-making processes [8]. One potential for the integration of prior knowledge is in the initialization of solutions and also in the mutation function of the feature selection process.

Kropp et al. [62] present a Sparse Population Sampling technique to enhance the efficiency of optimization algorithms in sparse settings by seeding the population with sparse initial solutions, reflecting a strategic use of prior knowledge for algorithm initialization. Similarly, a study by Xu et al. [63] introduces an evolutionary approach that employs duplication analysis to streamline the feature selection process, leveraging patterns of feature redundancy to implicitly incorporate prior insights. Song et al. [64] combine feature clustering based on correlations with particle swarm optimization, using prior knowledge of feature relationships to address feature selection challenges. Ren et al.[65] develop an algorithm for sparse optimization, hinting at the consideration of the distribution of non-zero elements to inform its strategy, thus possibly integrating domain-specific knowledge indirectly. Additionally, Wang et al. [66] apply multiobjective differential evolution to balance feature count minimization with classification performance, potentially adjusting evaluation criteria based on domain-specific feature importance, suggesting an indirect method of utilizing prior knowledge.

Despite these advancements, none of these methods explicitly utilize predefined features as prior knowledge, nor do they directly address problems characterized by high dimensionality combined with small sample sizes.

There is a significant gap in addressing the challenges of EMO in MOFS tasks, particularly in achieving stable results with high-dimensional datasets and small sample sizes. Integrating domain expertise and prior knowledge can influence initial conditions and evaluation trajectories, potentially enhancing the stability and performance of EMO in these contexts. This gap is addressed in this work by the direct utilization of domain knowledge in partially initializing algorithm populations and by finely adjusting mutation probabilities to balance exploration and exploitation. This approach embeds predefined, domain-specific knowledge at the outset of the optimization process, ensuring that evolutionary trajectories are informed by critical insights from the start.

3. Informed Automatic Modeling Pipeline for UHPC Production Process

This section outlines the establishment and application of a modeling pipeline specifically tailored for the UHPC production process, as depicted in Figure 2. The process begins with the Design of Experiments and Augmentation of Design techniques to compile comprehensive datasets. This step is followed by data preprocessing to prepare the data for modeling.

The next phase introduces the Ensemble-based Feature Importance Determination [6], utilizing the prepared data to identify significant features as prior knowledge for the next step. The culmination of the pipeline is the block of Informed Evolutionary Multiobjective Feature and Algorithm Selection (IEM-FAS), which implements a systematic approach to optimize feature and algorithm selection, specifically designed for high-dimensional, small datasets typical in UHPC production. This phase enables the progressive and systematic integration of proposed features based on their significance, thereby facilitating the model's development in the final stage.

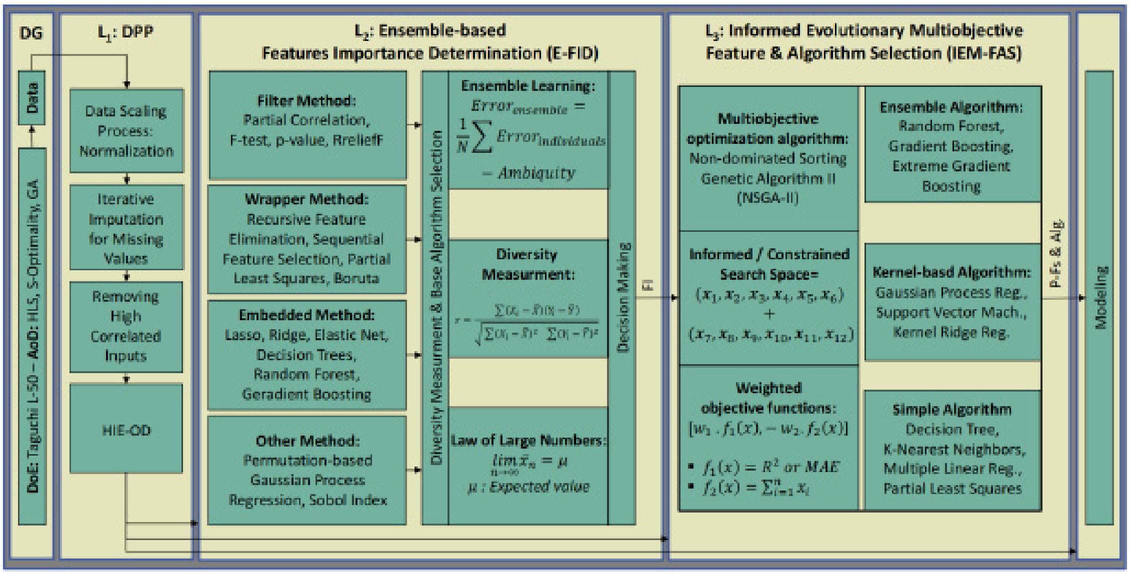


Figure 2: Modeling of Reproducible UHPC with Consistent Properties[2]

3.1. Data Generation

A dataset of x ∈ R22 with N = 150 observations and two outputs (Compressive and Flexural Strength on Day 28 after the mixing process) was generated for this work. For a detailed description of this dataset, including the complete data generation and analysis processes, readers are encouraged to refer to [67].

3.1.1. Key Factors and Characteristics in UHPC Production Process

For this study, a fixed reference recipe with a constant amount of materials is considered. However, the quality of UHPC can be influenced by various critical factors throughout its production process. These factors include material quality, particle size distribution, environmental conditions during raw material storage, measurement errors in the dosing system, and mixing and curing conditions (refer to Table 1). To evaluate the quality of UHPC, alongside mechanical tests at different ages, temperature, electrical conductivity, air content, slump flow test [68], and funnel runtime are also measured in the fresh state.

In the initial stage, the effect of material quality, environmental conditions during raw material storage, and impurities in silica fume are examined by considering variables such as Material Delivery Batch Time (DB), Cement Reactivity (CR), Ingredient Moisture (IM), Ingredient Temperatures (IT), and Graphite content (GRP). Materials are delivered in different batches, and even those from the same company (classified as DB1 and DB2) may exhibit subtle differences. These distinctions, though minor, can introduce variability in material quality at the microstructural level, emphasizing the need to account for such nuances in the production process. Cement Reactivity, influenced by its chemical composition and external factors such as storage duration and environmental conditions, is classified based on storage time. Varying humidity levels during storage can alter the cement's reactivity, impacting the final UHPC quality. Concrete raw materials in industrial settings are typically stored outside without protection from the sun, rain, and humidity. To simulate these conditions,

Figure 2: Proposed Automatic Modeling Pipeline for Ultra-High Performance Concrete Manufacturing Process: A Comprehensive Approach from Data Generation to Modeling. (DG: Data Generation, GA: Genetic Algorithm, LHS: Latin Hypercube Sampling, AoD: Augmentation of Design, DoE: Design of Experiments, L: Layer, DPP: Data Preprocessing, HIE-OD: Human-in-the-Loop Informed Ensemble-based Outlier Detection, FI: Feature Importance, MAE: Mean Absolute Error, P-Fs & Alg.: Proposed Features & Algorithm)

the temperature of raw materials before mixing is artificially set to different values. In addition to the fixed water value in the reference recipe, variations simulating humidity are considered. One of the crucial characteristics of UHPC is its low water content formulation; thus, impurities that absorb water are significant. High carbon content in silica fume can reduce workability by absorbing water, impacting cement hydration and final UHPC quality.

The formulation of the recipe, including appropriate ratios of aggregates, Superplasticizers (SPP), and silica fume, is critical for UHPC's strength and durability. Precision in measuring these ingredients and potential measurement errors, along with variations in particle size distribution, introduce complexities. Although these errors may be small (see Table 1), they can lead to significant variations in UHPC quality, especially when amplified by mixing parameters such as Speed (MS) and Duration (MD). Properties of fresh concrete, including Temperature (FCT), Electrical Conductivity (EC), Air Content (AC), Slump Flow (SF), and Funnel Runtime (FR), are crucial for assessing homogeneity, workability, and structural integrity of UHPC. These factors are considered to predict final product quality and provide feedback to operators to avoid off-spec products at the fresh stage of UHPC.

In real UHPC applications, the product is used worldwide under varying environmental conditions. To simulate this variability, different curing conditions, detailed in Figure 3, involving two main stages, are designed in this study. Initially, the UHPC transitions from paste to a hardened state with minimal strength. During the first 24 hours, it is either stored in a humidity-controlled cabinet at 90 % relative humidity and 20 ℃ (Figure 3a) or covered with plastic film and stored at 20 to 40 ℃, depending on environmental conditions (Figure 3b). After the first 24 hours, the specimens are demolded and cured until day 28. They are either maintained in plastic film at 20 ℃ (Figure 3c) or submerged in water with temperatures varying from 20 to 40 ℃(Figure 3d). Integrating these conditions into the developed pipeline provides an in-depth understanding of UHPC production, aiming for optimization and consistency in quality.

(a) Specimens Subjected to Different Humidity and Temperature Conditions

(b) Specimens Subjected to Different Temperature Conditions

(c) Specimens Encased in Plastic Film

(d) Specimens Submerged in Water

Figure 3: Illustration of Various Curing Conditions Employed in this Work on UHPC Production Process

By analyzing information at the fresh stage and adjusting curing conditions, operators can avoid waste and achieve the desired product quality.

3.1.2. Experimental Design and Methodology

This study initiated a structured experimental campaign, comprising 150 designed experiments performed at G.tecz Engineering GmbH in Germany, to delve into the UHPC manufacturing process. Addressing the challenges of data generation in this field, particularly the aspects of cost and time, we implemented a dual-phase approach: the Screening Phase for initial data analysis [6] and the Optimal Phase for detailed process modeling. The first phase established objectives, identified relevant variables, and carried out preliminary analyses, utilizing the Taguchi Orthogonal Array (TOA) [69]. TOA is favored for its efficiency in managing high-dimensional spaces with a limited number of experiments, ensuring a balanced distribution of input variables. In the subsequent phase, the experimental design was augmented through a process incorporating Latin Hypercube Sampling (LHS) [70], which strategically places additional points within the input space to optimize the experimental layout. A Genetic Algorithm [71] maximized the mean distance between these points (S-Optimality [72]), facilitating a comprehensive examination of the input space. The optimal integration of TOA and LHS data points established a robust foundation for further analysis.

The Screening Phase analysis led to the elimination of three variables: Cement Reactivity (CR), Mixing Speed (MS), and Mixing Duration (MD) [6]. These were determined to have negligible impacts on the final quality of the concrete and were therefore excluded from further consideration. For a more detailed understanding of the Screening Phase methodology and its results, interested readers are encouraged to consult [6].

3.1.3. General Setting

Throughout the experiments, the same mixing tool was used, with controlled environmental conditions for material storage and production to mitigate seasonal variations. The temperature of the mixer chamber was consistently maintained near the ambient laboratory temperature of 20 ℃ and was monitored before each experiment to ensure minimal impact on process variability.

3.2. Data Preprocessing

The collected data undergoes two stages of preprocessing: preliminary processing and outlier detection.

3.2.1. Preliminary Data Processing

Data standardization and normalization are utilized to enhance model performance. Standardization scales data to have a mean of zero and a standard deviation of one, while normalization scales data to fall within the range of [0, 1]. This normalization improved consistency and comparability across features, leading to better model performance in this study. As next step, to reduce dimensionality and prevent multicollinearity, Pearson's correlation coefficient [73] is used to identify and remove highly correlated inputs. This process helps in simplifying the model and improving its generalizability.

Iterative Imputation is employed to handle missing data, which constitutes 4 % of the dataset. Traditional imputation methods like mean or median imputation often fail to capture complex feature correlations and can result in biased estimates [74, 75]. Iterative Imputation [76–78], on the other hand, treats each variable with missing values as a function of other variables in a sequential process, offering a refined estimation that respects complex interdependencies among variables. This method, implemented using Scikit-Learn's IterativeImputer [79], employs Bayesian Ridge Regression [80–82] – the default model for the IterativeImputer function – to iteratively model and predict missing values, ensuring a more accurate and robust handling of the incomplete UHPC data.

3.2.2. Human-in-the-Loop Informed Ensemble-based Outlier Detection

Traditional techniques often struggle to identify outliers in complex datasets [83]. Several outlier detection methodologies, each beneficial in specific contexts, have been developed. These include statistical methods that detect deviations through probabilistic models, distance-based approaches that assess spatial separation, clustering-based strategies that identify outliers as ill-fitting data points within clusters, and density-based methods that pinpoint outliers due to their relative isolation in the data space [84,85].

However, these methods face challenges in specific scenarios, such as with small or sparsely distributed datasets, which are common in Design of Experiments contexts. This issue is particularly pronounced in high-dimensional datasets with small sample sizes, where the goal is to evenly cover the input space. Their reliance on proximity or density can lead to inaccuracies in outlier detection when data points lack sufficient neighborhood, cluster, or density characteristics (see Figure 4), potentially resulting in critical errors in measurement outputs [83, 85–88]. Furthermore, traditional methods such as box plots, histogram plots, and linear regression, which primarily focus on linear relationships, prove inadequate in non-linear contexts [83].

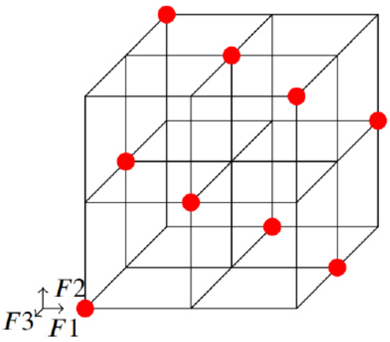


Figure 4: Visual Representation of an L9(33) Orthogonal Array [69]: Given the sparse data design by Taguchi Orthogonal Array, data points cannot be selected based on their distance, density, or neighborhood.

This paper introduces a novel method that combines expert knowledge and Human-in-the-Loop (HITL) insights with the robustness and generalization capabilities of ensemble methods to enhance outlier detection reliability in our case study. The proposed method, named Human-in-the-Loop Informed Ensemble-based Outlier Detection (HIE-OD) (Figure 5), is founded on five key principles:

- Diverse Base Learners
- Application of the Law of Large Numbers
- Domain Expertise for Informed Filtering
- Majority Voting for Ensemble Decision to Recommend Possible Outliers
- HITL Intervention for Final Decision-Making

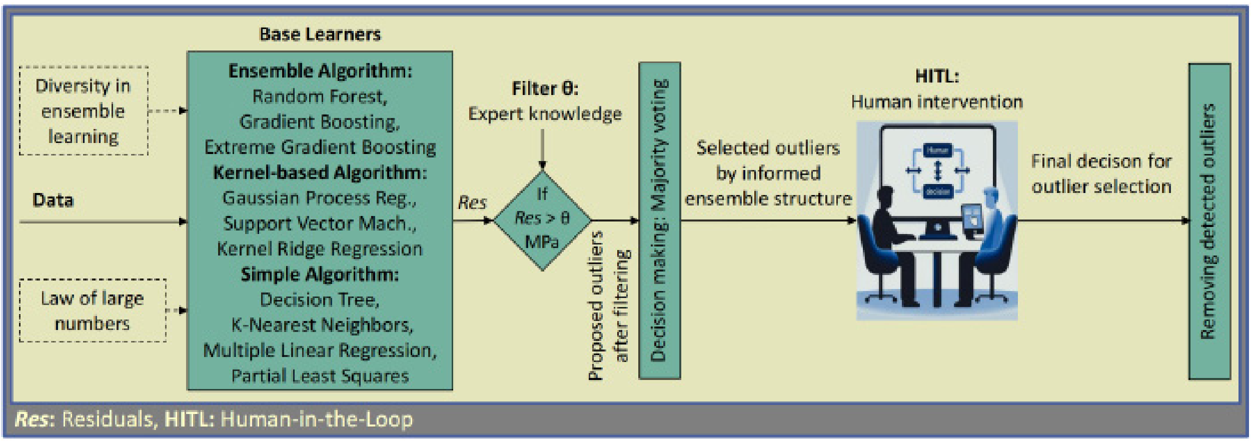


Figure 5: Human-in-the-Loop Informed Ensemble-based Outlier Detection (HIE-OD): Advancing outlier detection with informed ensemble-based method, leveraging expert knowledge and Human-in-the-Loop for precise anomaly detection. Possible outliers, recommended by the informed ensemble component of HIE-OD, are then subject to final decisions made through human intervention in a HITL process.

An ensemble learning approach, inspired by the wisdom of the crowd concept, enhances decision-making in machine learning by leveraging collective intelligence [89]. This method aggregates insights from various base learners (BLs), each contributing unique perspectives, thereby improving generalization and reducing overfitting [89]. The ensemble aims to achieve or exceed the average performance of its base learners, with diversity in the ensemble reducing error, as demonstrated by:

$$Error_{ensemble} = \frac{1}{M}\sum Error_{individuals} - Ambiguity, \tag{3}$$

where Errorindividuals is the ensemble base learners'(BLs) errors, M is the count of BLs, and Ambiguity measures their diversity [90]. This illustrates how diversity minimizes the overall error [91]. Depending on the application, the Law of Large Numbers supports this principle by indicating that a larger number of observations leads to a sample average that more accurately reflects the population mean, which is vital for effective outlier detection:

$$\lim_{N\to\infty} \bar{O}_N = \mu. \tag{4}$$

Here, *ON* represents the average of *N* observations, and *μ* is the mean value of the population, highlighting the improved reliability with an increasing number of observations. This convergence, as observed in our case study, assumes that the observations are independent, identically distributed, and exhibit finite variance. To leverage this principle, the proposed outlier detector incorporates 10 diverse BLs, ranging from linear to non-linear, parametric to non-parametric, and simple to complex algorithms, as illustrated in Figure 5.

For a dataset comprising N data points $\{d_1; d_2; \ldots ; d_N\}$ and a set of models, the residual R_{ij} for the j-th data point

against predictions from all models is calculated by the proposed method as:

$$R_{ij} = |y_j - \hat{y}_{ij}(d_j)|, \tag{5}$$

where yj is the actual value for the j-th data point, $\hat{y}_{ij}(d_j)$ denotes the predicted value for the *j*-th data point by the *i*-th model. A data point is flagged as a possible outlier if:

$$\max_i(R_{ij}) > \theta, \tag{6}$$

where is the threshold defined by expert knowledge.

The ultimate decision on outliers is determined using a wisdom of the crowd approach, which requires more than 50 % agreement among the algorithms:

$$C(d) = \frac{1}{M}\sum_{i=1}^{M} \mathbb{1}_{\{f_i(d)=\hat{y}_i \text{ and } R_i(d)>\theta\}}, \tag{7}$$

where *C*(*d*) denotes the consensus for data point d, $f_i(d)$ represents the decision function of the *i*-th BL, *M* is the total number of BLs, and is the consensus threshold set at 15 MPa. The indicator function **1** returns 1 if $R_i(d)$ > . A data point *d* is considered a possible outlier if *C*(*d*) exceeds 0.5, indicating majority consensus. If *C*(*d*) exceeds 0.5, the data point *d* is recommended to the HITL for further analysis. The 50 % threshold is the minimum required for the effectiveness of wisdom of the crowd (majority voting), as a binary decision process requires more than half of the votes to favor an option for it to be selected [92]. Setting the threshold below this compromises the purpose of majority voting, while setting it above risks missing true positives. In this study, setting the threshold at 60 % ensures that at least 6 out of 10 base learners agree, which balances minimizing false positives with maintaining the effectiveness of the wisdom of the crowd. By leveraging a blend of multiple BLs, informed detection criteria, and a consensus- and HITL-enhanced decision framework, the HIE-OD presents a comprehensive, sophisticated, and flexible approach to outlier detection.

3.3. Ensemble-based Feature Importance Determination

Within the context of this work, the Ensemble-based Feature Importance Determination (E-FID) framework has been deployed to analyze factors influencing UHPC quality. Unlike approaches that rely on a single model, the ensemble method is recognized for integrating insights from a diverse array of predictive algorithms [89]. This strategy is primarily adopted to overcome limitations inherent in individual models, such as bias and variance issues [91]. Furthermore, it is particularly effective in scenarios characterized by high dimensionality and limited data, where the ensemble approach enhances generalization and reduces the risk of overfitting [89]. By synergistically leveraging the strengths of multiple algorithms, the ensemble framework significantly enhances the reliability of feature importance rankings. This approach effectively mitigates the the law of ensemble learning described in Equation3and leverages the Law of Large Numbers as detailed in Equation4, facilitating a nuanced understanding of the critical factors influencing UHPC quality.

For a comprehensive exposition of the E-FID framework's development and operational specifics, readers are referred to [6].

3.4. Informed Evolutionary Multiobjective Feature and Algorithm Selection Framework

In the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [93], used for feature selection, an individual (chromosome) **x** is represented as a binary vector:

$$\mathbf{x} = [x1; x2; \dots; xn] \tag{8}$$

where n is the number of features, and each $x_i \in \{0; 1\}$ indicates the absence (0) or presence (1) of the i-th feature. The fitness of an individual is evaluated through multiple objective functions. Common objectives for feature selection in complex tasks include maximizing model prediction performance and minimizing the number of selected features.

The dominance relation between two individuals **x** and **y** in the case of a minimization problem is defined as:

$$\mathbf{x} \text{ dominates } \mathbf{y} \text{ if } \forall i; f_i(\mathbf{x}) \leq f_i(\mathbf{y}) \text{ and } \exists j; fj(\mathbf{x}) < fj(\mathbf{y}) \tag{9}$$

where f_i represents an objective function.

Crowding distance is a measure of the density of the solution space around a given individual, which helps maintaining diversity in the population. Crossover and mutation are genetic operators used to create new individuals [93]. These operators are typically implemented as follows:

- **Crossover** combines segments of two parent chromosomes to produce offsprings, introducing genetic diversity into the population. A common method used is single-point crossover.
- **Mutation** alters genes within a chromosome with a probability P_{mut} to introduce variability, often implemented as bit-flip mutation.

The algorithm iteratively performs selection, crossover, and mutation to generate new populations. After each iteration, both the parent and offspring populations are sorted based on non-dominance and crowding distance, preparing them for the next generation cycle [93].

The set of optimal solutions $\mathbf{X}^*$ is defined as [94]:

$$\mathbf{X}^* = \arg\min_{\mathbf{x}\in\mathbf{X}} / \max\{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_l(\mathbf{x})\} \tag{10}$$

where l is the number of objectives, **X** is the set of all possible solutions, and f_i are the objective functions to be minimized (or maximized, depending on the problem definition).

In the proposed Informed Non-dominated Sorting Genetic Algorithm II (I-NSGA-II), an individual (ind) is represented as a binary vector (see algorithm 1):

$$\mathbf{X} = [\mathbf{xn}; \mathbf{hk}] = [x1; x2; \dots; xn; h_1; h_2; \dots; h_k]; \tag{11}$$

where $x_i \in \{0; 1\}$ indicates the absence (0) or presence (1) of the i-th feature, and h_j are hyperparameters for the machine learning algorithm. The total number of features is denoted by n, and the number of hyperparameters by k.

A crucial aspect of our algorithm is the incorporation of predefined features based on prior knowledge. Let $S \subseteq \{1; 2; \dots; p\}$ represent the set of indices corresponding to these predefined features (see Figure 6). During the initialization and mutation phases of the algorithm, these predefined features are enforced by setting $x_i = 1$ for each $i \in S$. Mathematically, this enforcement is represented as:

$$x_i = \begin{cases} 1 & \text{if } i \in S \\ x_i & \text{otherwise} \end{cases} \tag{12}$$

This approach ensures the inclusion of predefined features in each individual of the population across generations, thereby enhancing the robustness of the feature selection process. However, to introduce a controlled degree of variability and to avoid stagnation in local optima, the mutation operation is applied with a probability P_{mut}, typically a small value. For predefined features, the mutation probability is effectively reduced or nullified, denoted by P_{mut}^{predef} to minimize the alteration of these features. The mutation step for the i-th feature can thus be expressed as:

$$x_i' = \begin{cases} x_i & \text{if } i \in S \text{ and rand}() > P_{\text{mut}}^{\text{predef}} \\ 1 - x_i & \text{if } i \in S \text{ and rand}() < P_{\text{mut}}^{\text{predef}} \\ 1 - x_i & \text{if } i \notin S \text{ and rand}() < P_{\text{mut}} \\ x_i & \text{otherwise} \end{cases} \tag{13}$$

where rand(), a function, generates a real-valued random number between 0 and 1.

Algorithm 1 I-NSGA-II:

Require: D = $\{X;y\}$, where X is the feature matrix, y is the target variable, and N is the number of data points. **Require:** Predefined features F_{predef} with selection probability P* jc3 * hps14 \o\al\s\up 5premu$_t^{def}$, and Mutation selection probability P_{mut}.

Require: Population (pop) size Q, Number of generations G, Crossover probability Pc, Mutation probability P_m.

```
function INITIALIZEPOP(Q; F)
    for i ← 1 to Q do
        ind_i ← initialize with F_predef and random features
        ind_i ← append random hyperparameters
    return pop
function CALCULATEFITNESS(ind; D_train)
    X_subset ← X[F_i]; F_i ← feature subset from ind
    H_i ← hyperparameters from ind
    Train/Validate model on X_subset using H_i (CV=10)
    fit_1 ← model's R2 score; fit_2 ← −count(F_i)
    return weighted fitness (2 . fit_1 ; 1 . fit_2 )
function MUTATE(ind; Pmut; P\* jc3 \* hps14 \o\al\s\up 4premu_t^def ; Fpredef)
    for each feature f in ind do
        if f ∈ F_predef and rand() > P\* jc3 \* hps14 \o\al\s\up 4premu_t^def then
            Keep f unchanged
        else if f ∉ F_predef and rand() < P_mut then
            Toggle inclusion of f in ind
        else
            Keep f unchanged
    for each hyperparameter h in ind do
        if rand() < P_mut then
            Adjust hyperparameter h
    return ind
function GA(D_train; Q; G; Pc; Pm; Pmut; P\* jc3 \* hps14 \o\al\s\up 4premu_t^def)
```

$pop \leftarrow$ INITIALIZEPOP(Q;F)
for $g \leftarrow 1$ to G **do**
Assess and assign fitness to each ind $\in$ *pop*using CALCULATEFITNESS(ind;D_{train})
offspring $\leftarrow$ apply crossover with probability P_c on selected parents; Select parents from *pop*
Apply mutation with probability P_m on *offspring*
for each ind $\in$ *offspring* **do**
ind $\leftarrow$ MUTATE(ind;*P*mut;*P** jc3 * hps14 \o\al\s\up 5premu$_t^{def}$;Fpredef)
Ensure compliance with F_{predef}
Recalculate fitness using CALCULATEFITNESS(ind;D_{train})
pop $\leftarrow$ next generation from *pop*$\cup$ *offspring*
Fopt;H_{opt} $\leftarrow$ best ind in *pop*based on multiobjective criteria
return Fopt;H_{op}
for each run $r \leftarrow 1$ to R**do** ▷LOOCV with different random initialization of the algorithm in each iteration.
D_{train};D_{test} $\leftarrow$ Split dataset D using LOOCV
F^r_{opt};H^r_{opt} $\leftarrow$ GA(D_{train};Q;G;Pc;Pm;*P*mut;*P** jc3 * hps14 \o\al\s\up 4premu$_t^{def}$)
Evaluate F^r_{opt};H^r_{opt} on D_{test}
Record the true and predicted values

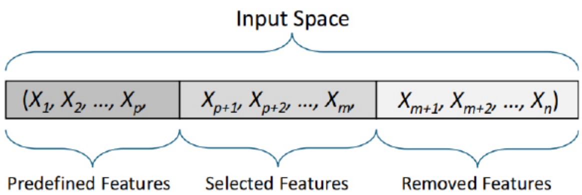


Figure 6: Overview of I-NSGA-II Search Space for Feature Selection: I-NSGA-II is designed to explore a constrained search space, prioritizing predefined features (X_1 to X_p) with a high likelihood of selection during each individual generation. It aims to identify the most significant features (X_{p+1} to X_m) whose interactions with predefined features greatly impact the objectives. This structure limits the search space by integrating prior knowledge into the search space of NSGA-II, enhancing its performance and ensuring stable solutions by addressing the challenges of multiobjective feature selection, especially for high-dimensional data with small sample sizes.

In the mutation operation, if neither condition for mutating predefined ($i \in S$) nor non-predefined features is satisfied – specifically, when rand() $\geq P_{mut}^{predef}$ for predefined features and rand() $\geq P_{mut}$ for others – the feature indicator x_i remains unaltered (N). This *otherwise* case ensures the integrity of the individual by stabilizing the feature composition against unnecessary random perturbations, thereby preserving the current selection state for both sets of features and maintaining the evolutionary approach's balance between exploring the solution space and respecting predefined feature sets. This strategic design guides the genetic algorithm towards effective solutions that adhere to problem-specific constraints.

In the optimization framework of the I-NSGA-II, emphasis is placed on both maximizing the model's predictive accuracy and minimizing the size of the feature set. Weights are assigned to reflect the relative importance of these objectives, with predictive accuracy deemed twice as significant as the simplicity of the model. Consequently, the objective formulation is expressed as:

where $f_1(\mathbf{x}) = R2(\mathbf{x})$ represents the model's predictive accuracy.

The simplicity criterion, [illegible] the model's complexity by the sum of selected features, with each x_i indicating the inclusion (1) or exclusion (0) of the i-th feature. This weighting scheme methodically emphasizes the enhanced priority of accuracy over minimizing feature count, balancing the trade-off between high predictive performance and model simplicity.

The general process of the I-NSGA-II is recorded in Algorithm 1. The core of our methodology relies on a dataset D = $\{X; y\}_{j=1}^{N}$, where X represents the set of feature values, y is the target variable value, and N is the number of data points. The algorithm commences with a predefined set of features F_{predef} and operates under specific operational parameters.

The initialization phase, InitializePop, creates a population of Q individuals, each initialized with the prede- fined feature set F_{predef} and a random selection of additional features, augmented by randomly chosen hyperparameters. The fitness of each individual is assessed in the CalculateFitness function. This function extracts a feature subset F_i and hyperparameters H_i from each individual, trains a model on X_{subset} using Hi, and calculates fitness based on model accuracy and the count of features used. The dual objectives are to achieve high accuracy and minimize the feature set, while also respecting the constraints imposed by predefined features.

Mutation, handled by the Mutate function, toggles the inclusion of features in an individual based on Pmut, provided the features are not part of F_{predef}. The genetic algorithm (GA), detailed in the GA function, iteratively performs the following steps: evaluates the population, selects parents, and applies crossover and mutation according to P_c and P_m. Offsprings undergo mutation, compliance adjustment to ensure adherence to predefined feature constraints, and fitness recalculation to update their fitness values based on the new feature set and hyperparameters. After G generations, the algorithm selects the best individual based on the optimized feature subset F_{opt} and hyperparameters H_{opt}.

In the nested validation and evaluation loop, a single test data point is extracted at the start of the IEM-FAS framework using the LOOCV mechanism for the final test phase. Then, for training and validation of each algorithm and finding the optimal features and hyperparameters in the I-NSGA-II algorithm, 10-fold CV is used on the training data from LOOCV. After finalizing the model and identifying the optimal features and hyperparameters, the model is evaluated using the unseen data points from LOOCV. These steps are repeated across LOOCV with different initializations for each fold, and the average prediction performance and frequency of selection of each feature are recorded.

3.4.1. Parameter Optimizationfor I-NSGA-II

To configure and run the I-NSGA-II algorithm optimally, we conducted a series of trial-and-error experiments to determine the most effective parameter values. The parameters include a mutation selection probability P_{mut} = 0:05, a population size Q = 100, a number of generations G = 100, a crossover probability P_c = 0:9, and a mutation probability P_m = 0:2.

The objective function, also known as the fitness function, consists of two objectives: prediction accuracy and the number of features. To reflect the importance of prediction accuracy over the number of features, the fitness function assigns a weight of 2 to prediction accuracy and a weight of 1 to the number of features. The prediction accuracy is

calculated using the $R2$ metric, chosen after evaluating several metrics including *MAE*, *RMSE*, *R** jc3 * hps14 \o\al\s\up 32*adj*, and $R2$.

The search space for feature selection in I-NSGA-II consists of 16 possible bits. Of these, 6 bits are allocated for predefined features and 6 bits for other features, allowing 4 features to be potentially removed from the pool of 16 features in each run. These configurations are also derived from trial-and-error experiments aimed at achieving the best performance.

3.4.2. Machine Learning Algorithms

Within the IEM-FAS framework (see Figure 2), the feature selection methodology employs a comprehensive set of 10 different machine learning algorithms, ranging from parametric and non-parametric methods, including simple linear, advanced ensemble, and Bayesian approaches. The specific algorithms used are Multiple Linear Regression (MLR) [95], Partial Least Squares (PLS) [95, 96], Kernel Ridge Regression (KRR) [95, 97], K-Nearest Neighbors (KNN) [95, 98], Support Vector Regression (SVR) [95, 99], Decision Tree (DT) [95, 100], Random Forest (RF) [95, 101], Gradient Boosting (GB) [95, 102], Extreme Gradient Boosting (XGB) [103], and Gaussian Process Regression (GPR) [95, 104]. The selected machine learning algorithms are commonly used in industry prediction applications [105], particularly for predicting the mechanical properties of concrete.

3.4.3. Evaluation Metrics of Models Performance

In the process of evaluating the performance of trained models within the testing phase, a comprehensive approach is adopted, utilizing different evaluation metrics tailored specifically for regression tasks. The R^2 metric provides a measure of how well variations regarding the average of output in the observed outcomes are explained by the model:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - y)^2} \tag{15}$$

The R^2_{adj} adjusts the R^2 statistic based on the number of predictors in the model, preventing overestimation of the model's explanatory power when more predictors are added:

(16)

where D is the number of predictors. The R^2_{adj} thus accounts for the model's complexity and is always lower than or equal to R^2.

The *MAE* (Mean Absolute Error) measures the average magnitude of the errors in a set of predictions, without considering their direction:

(17)

The*MAPE* (Mean Absolute Percentage Error) expresses the error as a percentage of the observed values, providing a simple interpretation of the error magnitude:

(18)

The*RMSE* (Root Mean Squared Error) is a quadratic scoring rule that measures the average magnitude of the error.

It is calculated as the square root of the average squared differences between the predictions and actual observations:

(19)

In all metrics, yi represents the observed values, $\hat{y}i$ the predicted values, y the mean of the observed values, and N the number of observations.

3.5. UHPC Manufacturing Process Modeling

Building on the ranked features and the best algorithm (with its optimal hyperparameter configuration) proposed by the IEM-FAS framework (see Figure 2), this section outlines a streamlined modeling phase, as detailed in Algorithm 2. The methodology employs the proposed algorithm with LOOCV, beginning with data loading and normalization. The approach incrementally introduces features based on their importance ranking determined by the E-FID and the IEM-FAS frameworks, preparing them alongside the target variable for analysis.

Algorithm 2 UHPC Production Process Modeling

function PREPAREDATA(*data*, *features*, *target*)
    $X \leftarrow data[features]$
    $y \leftarrow data[target]$
    Normalize X to have values between 0 and 1
    return X, y
function LOOCV(X, y)
    Initialize *predictions* as empty list
    for each split in LOOCV of X, y **do**
        Train model on the training set
        predictions ← Predict on the test set and save it
    return average of the MAE, R^2 ; $\boldsymbol{R}^2_{adj}$
function MAIN
    data ← A dataset with $X \in \mathbf{R}N \times D$ and $Y \in \mathbf{R}N \times 1$
    Define *features*, *target*
    for $i \leftarrow 1$ **to** length(*features*) **do**
        $currentFeatures \leftarrow features[1 : i]$
        $X, y \leftarrow$ PREPAREDATA(*data*, *currentFeatures*, *target*)
        $avgMAE$, $avgR^2$; $avg\ \boldsymbol{R}^2_{adj}$ ← LOOCV(X, y)

The model is trained and evaluated iteratively, using each data point as a test case while the remaining data points serve as the training set. During this process, features are systematically added according to their importance to assess their cumulative impact on the model's performance. The performance metrics, R^2, $\boldsymbol{R}^2_{adj}$, and MAE, are calculated based on predictions made on the test data. This approach allows for an effective evaluation of the model's predictive accuracy and provides insight into the impact of each feature as recommended by the IEM-FAS framework.

4. Results and Discussions

4.1. Assessing Data Preprocessing

4.1.1. Correlation Patterns in Studied Factors and UHPC Mechanical Properties

In the preprocessing phase of this study, based on the results of the Screening Phase [6], which led to the removal of three features (Cement Reactivity, Mixing Speed, and Mixing Duration), 19 factors from an initial pool of 22 (Table 1) were selected for further analysis based on their relevance to the final quality of UHPC.

Table 1: This dataset [67] includes a primary Ultra-High Performance Concrete (UHPC) recipe, highlighting variations in material quality, potential measurement errors in the primary recipe, mixing conditions, fresh concrete characteristics, and curing conditions. The cement and silica fume content are fixed across all experiments in this study.

Group	Factor name	Var.	Unit	Mean	Median	Std	Min	Max
Material quality	Material delivery batch time	*DB*	Class	-	-	-	1	2
	Cement reactivity	*CR*	Class	-	-	-	1	2
	Ingredient moisture	*IM*	%	3.13	3.15	0.16	2.92	3.36
	Ingredient/Water temperature	*IT*	°C	24.20	25	9.05	10	40
	Graphite	*GRP*	Kg	0.08	0.09	0.07	0	0.22
Particle size distributions & Measurement errors	Sand I	*SAI*	Kg	5.98	6	0.59	5.10	6.90
	Sand II	*SAII*	Kg	10.53	10.50	1.04	8.92	12.07
	Filler I	*FLI*	Kg	6.00	6	0.59	5.10	6.90
	Filler II	*FLII*	Kg	0.75	0.75	0.07	0.63	0.86
	Superplasticizer	*SPP*	Kg	0.32	0.32	0.02	0.29	0.35
Mixing conditions	Mixing speed	*MS*	RPM	275.75	200	101.35	200	500
	Mixing duration	*MD*	s	309.09	300	63.63	210	480
	Average power consumption	*APW*	kW	1.04	1.06	0.19	0.36	1.40
Fresh concrete properties	Fresh concrete temperature	*FCT*	°C	26.77	27	3.43	17.6	33.30
	Electrical conductivity	*EC*	V	4.615	4.611	0.029	4.541	4.745
	Air content	*AC*	%	1.62	1.50	0.77	0.40	7
	Slump flow	*SF*	mm	335.91	340	26.35	215	395
	Funnel runtime	*FR*	s	7.53	7	2.82	4	24.10
Curing conditions	Curing temperature day 1	*CT1*	°C	24.60	20	9.72	10	40
	Curing class day 1	*CC1*	Class	-	-	-	1	2
	Curing temperature day 2–28	*CT28*	°C	22.15	20	9.38	10	40
	Curing class day 2–28	*CC28*	Class	-	-	-	1	2
Output	Compressive strength at day 28	*CS28*	MPa	109.83	110.57	11.83	85.06	135.71
	Flexural strength at day 28	*FS28*	MPa	16.98	17.29	3.60	8.15	24.57

To refine the dataset and reduce dimensionality, correlations among the factors were examined using the Pearson correlation method. The correlation analysis, illustrated in the heatmap presented in Figure 7, identified strong correlations among certain pairs of variables: *SAI* and *SAII* (Sand Type I and II), *FLI* and *FLII* (Filler Type I and II), and *IT* and *FCT* (Ingredient Temperature and Fresh Concrete Temperature). Due to the high correlations, *SAII*, *FLII*, and *FCT* were removed to avoid factor redundancy. This decision led to a reduction of the factor pool to 16.

Subsequent correlation analysis between the refined set of input factors and key outputs, specifically Compressive Strength at day 28 (*CS28*) and Flexural Strength at day 28 (*FS28*), revealed significant relationships. This analysis highlighted the impact of curing temperature during the second stage (*CT28*) on both mechanical properties of the final UHPC product. Notably, the conditions under which specimens were cured – whether submerged underwater or encased in air within a plastic film from day 2 to day 28 (*CC28*) – were determined to critically affect *FS28*.

This highlights the importance of maintaining a continuously wet surface on UHPC during the curing process. Due to its high binder content (cement and silica fume) and low water content, UHPC does not contain enough water to fully hydrate all the binders. As a result, it is essential to compensate for this water deficit by absorbing moisture from the environment.

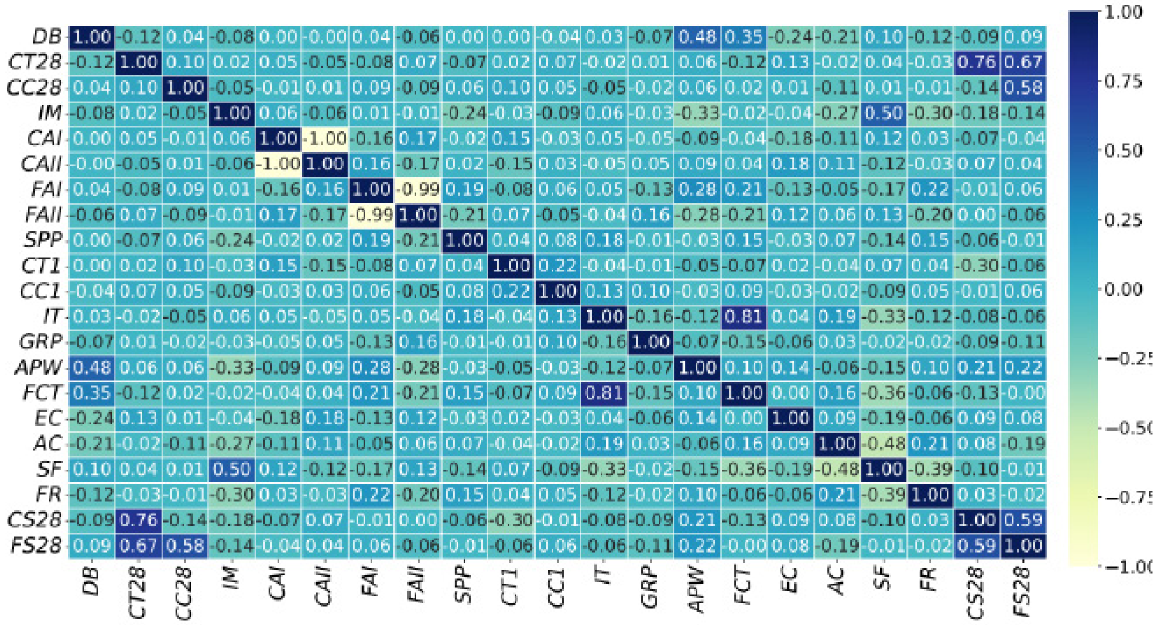


Figure 7: Pearson Correlation Coefficients for Variables in UHPC Production: This heatmap outlines the correlations between material composition, processing parameters, and environmental factors, as well as their correlations with the compressive (CS28) and flexural (FS28) strengths of UHPC at day 28 of curing. For a detailed explanation of the variables used in this heatmap, see Table 1.

4.1.2. Outlier Detection by HIE-OD

Figure 8 illustrates the distribution of the outputs (*CS28* and *FS28*) using box, histogram, and scatter plots. The box and histogram plots suggest the presence of two potential outliers. As discussed in Section 3.2.2, and further evidenced by the scatter plots in Figure 8, these data points do not provide a clear basis for assessment using clustering and distance-based perspectives commonly applied in outlier detection.

During the first stage of outlier detection, the two potential outliers identified from the box and histogram plots were examined and confirmed as true outliers by domain experts. Additionally, one data point, which exhibited some missing values in fresh concrete characteristics and outputs, was also identified as problematic. Consequently, these three data points were removed from the initial dataset of 150 data points, reducing the dataset to 147 data points.

The results from the HIE-OD method are detailed in Table 2. This table provides a comprehensive summary of votes from an ensemble of 10 BLs for the detected experiments, indicating that the experiments listed were identified as possible outliers by at least one BL. Experiments not listed in the table were not detected as possible outliers by any of the 10 BL.

The criterion for outlier detection by each BL was set to exceed an informed threshold of $\theta = 15$ MPa in residuals. The criterion for outlier detection by the informed ensemble-based part was established using majority voting, with a benchmark of six or more votes required for potential outlier identification. This threshold is depicted in Table 2with red rectangles for easy reference. The experiments marked with red rectangles – numbers 4, 16, 29, 44, 54, 96, 98, and 144 – were identified by the informed ensemble-based part as potential outliers. This threshold of six votes was strategically chosen to balance the need for sensitivity in detecting outliers against the risk of false positives.

Remarkably, each data point identified as an outlier by the informed ensemble part of the HIE-OD method, using the majority voting criterion, underwent subsequent examination by domain experts. This review process validated the ensemble method's recommendations, with all highlighted experiments being confirmed as true outliers.

Following expert validation, all eight data points recommended as outliers were removed from the dataset. This action reduced the dataset to a total of $N = 139$ data points.

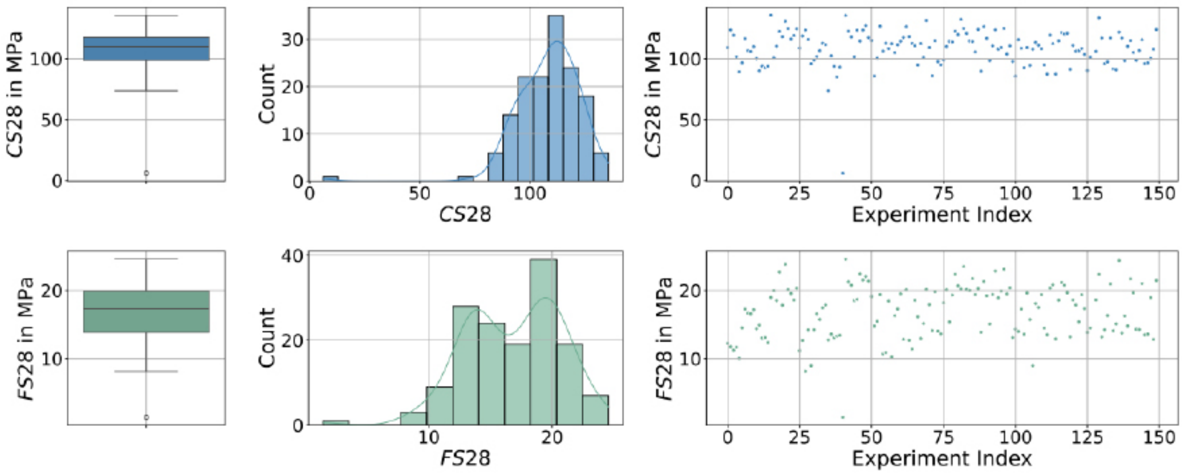


Figure 8: Distribution of Compressive Strength (CS28) and Flexural Strength (FS28) at Day 28 Using Box, Histogram, and Scatter Plots: The box and scatter plots reveal one possible outlier in both outputs, while the histograms suggest two possible outliers, especially for CS28.

Table 2: Summary of detection outcomes from 10 base learners on data points identified as containing potential outliers, with red rectangles highlighting experiments receiving six or more votes indicating a consensus on outlier status. All flagged data points were validated by domain experts. (BL: Base Learner, Exp.: Experiment)

BL \ Exp. Index	4	15	16	27	29	30	42	44	54	60	74	84	87	96	98	102	127	136	144
Multiple Linear Regression			X		X			X	X		X			X	X				X
Partial Least Squares			X		X			X	X					X					X
Kernel Ridge Regression	X	X	X	X	X			X	X					X	X				X
K-Nearest Neighbors	X		X		X						X				X				X
Support Vector Machine	X		X	X	X			X	X					X					X
Decision Tree	X			X			X					X	X			X		X	X
Random Forest	X		X		X	X		X	X					X	X			X	X
Gradient Boostong	X	X	X		X	X		X		X	X				X		X		
Extreme Gradient Boosting			X		X	X					X				X		X		X
Gaussian Process Regression			X	X	X			X	X					X	X				X
Majority Voting	$\boxed{6}$	2	$\boxed{9}$	4	$\boxed{9}$	3	1	$\boxed{7}$	$\boxed{6}$	1	4	1	1	$\boxed{6}$	$\boxed{7}$	1	2	2	$\boxed{9}$

4.2. Gaining Insights into Feature Importance for UHPC Mechanical Properties Using E-FID

The feature importance analysis for *CS28* (Figure9) highlights the paramount importance of Curing Temperature from day 2 to 28 (*CT28*), underscoring the critical role that environmental conditions play during the second phase of the curing process. Interestingly, the Initial Curing Temperature (*CT1*) on the first day of curing also emerged as a significant factor, albeit with less influence than *CT28*. This suggests that the curing conditions on the first day establish a significant foundational strength, which is further enhanced by the curing conditions from day 2 to day 28. This confirms the well-known fact that higher temperatures accelerate cement hydration. However, it is equally important not to overlook the need for sufficient moisture in such environments.

Ingredient Moisture (*IM*) shows a crucial impact on the final compressive strength of the UHPC, highlighting the importance of moisture content within the mix. Similarly, Average Power Consumption (*APW*), while not a controllable factor, serves as an informative indicator, reflecting the rheology of concrete paste (energy input) during the mixing process and helping to predict the final compressive strength. The addition of Graphite (*GRP*) (to simulate the impurity in silica fume) and of the Curing Conditions from day 2 to day 28 (*CC28*) also play notable roles in the analysis. The presence of carbon (Graphite) as an impurity in silica fume significantly absorbs water in the mixture, leading to a reduction in flowability. This is a critical factor, especially in UHPC, which has a very low water content.

For *FS28*, as detailed in Figure 9, the dominance of curing temperature from day 2 to day 28 (*CT28*) remains unchallenged, reinforcing the overarching influence of the curing processes. However, in a notable departure from the findings related to compressive strength, the curing condition from day 2 to day 28 (*CC28*) stands out as the second most critical factor for flexural strength as well. This distinction highlights the different impact of environmental conditions on the material's resistance to bending stresses.

Ingredient Moisture (*IM*) and initial Curing Temperature (*CT1*) retain their significance, reflecting a consistent theme across both strength characteristics regarding the importance of moisture and initial curing conditions. Notably, Air Content (*AC*), measured after the mixing step in fresh concrete, emerges as a more informative factor for predicting *FS28* compared to *CS28*. This indicates its role in affecting the material's flexural properties, likely through its influence on the pore structure and distribution within the concrete matrix. The comparative analysis of *CS28* and *FS28* results from the E-FID framework reveals a nuanced landscape of feature importance, with several key takeaways:

- Curing Conditions' Primacy: The curing temperature at various stages unequivocally influences both compressive and flexural strength, emphasizing the need for controlled environmental conditions throughout the curing process.

- Differential Impact of Factors: Certain factors, such as *CC28* and *AC*, exhibit a varied influence on *CS28* versus *FS28*. *AC*, particularly, serves as an informative indicator rather than a direct influencing factor, highlighting its role in affecting the material's flexural properties through its influence on the pore structure and distribution.

However, this aspect needs further study, as the presence of carbon is expected to influence compressive strength similarly to flexural strength.

- Importance of moisture and energy consumption of the mixer: The consistent significance of Ingredient Moisture (*IM*) across both analyses underscores the fundamental role of ingredient quality in determining UHPC's mechanical properties. This further emphasizes the necessity of maintaining adequate moisture levels to ensure proper binder hydration and optimal performance. Simultaneously, *APW*, as an informative indicator of mixing efficiency, aids in predicting UHPC's strength outcomes rather than directly influencing them.

4.3. Enhanced Predictive Modeling of UHPC Mechanical Properties Using I-NSGA-II

4.3.1. Impact of I-NSGA-II on Model Performance and Algorithm Selection

This study presents a comprehensive evaluation of various machine learning algorithms for predicting the *CS28* and *FS28* of the UHPC. In the IEM-FAS framework, each algorithm is trained and tested using the LOOCV approach, with different random initializations in each fold. The entire process employs both I-NSGA-II and NSGA-II for both outputs. After training, the performance of each model is tested using the test dataset based on several metrics: $R2$, *MAE*, *MAPE*, and *RMSE*. The results from the average of all folds in the LOOCV loop during the test step are summarized in Table 3afor *CS28* and in Table 3bfor *FS28*.

In Table 3a, models employing the I-NSGA-II demonstrated substantial improvements in $R2$ values. For instance, the MLR model experienced an increase from 72.47 % under traditional Feature Selection (FS) using NSGA-II to 76.37 % with Informed Feature Selection (I-FS) using I-NSGA-II. This improvement indicates a more robust model fit, which can be attributed to the effective integration of domain-specific knowledge within the feature selection process. Moreover, reductions in *MAE*, *MAPE*, and *RMSE* across models further validate the efficacy of I-NSGA-II. KRR and GB also exhibited significant improvements with the application ofI-NSGA-II. Notably, KRR demonstrated an increase in $R2$ from 62.38 % to 76.26 %, marking one of the highest improvements observed. This enhancement is accompanied by a notable decrease in *RMSE* from 7.33 to 5.66, underscoring the effectiveness of I-NSGA-II in reducing prediction errors.

In the context of flexural strength (*FS28*), the SVR model demonstrated the most substantial gains with I-NSGA-

II, as detailed in Table 3b. The $R2$ value surged from 72.62 % under NSGA-II (FS) to 81.75 % under I-NSGA-II (I-FS), highlighting a significant enhancement in the model's ability to capture the variability in flexural strength data. Moreover, the *MAE* reduced dramatically from 1.56 MPa to 0.93 MPa, indicating a higher accuracy in the model's predictive performance. Similarly, the *MAPE* and *RMSE* mirrored this trend, improving from 8.54 % to 7.08 % and from 2.04 to 1.37, respectively. GPR also showed improved performance metrics with the application of I-NSGA-II (FS). The $R2$ value increased from 72.67 % to 81.70 %, and there were significant reductions in both *MAE*, *MAPE*, and *RMSE*, reinforcing the effectiveness ofI-NSGA-II in enhancing the predictive accuracy of complex regression models.

In summary, the application of I-NSGA-II across various machine learning models consistently outperforms the traditional NSGA-II method in all assessed metrics for both compressive and flexural strengths after 28 days (Table 3). This comprehensive analysis conclusively demonstrates the superior predictive capabilities ofthe I-NSGA-II approach, establishing its efficacy in enhancing model performance. Notably, for compressive strength (*CS28*), the MLR model exhibits the most notable enhancement, emerging as the optimal model. Similarly, for flexural strength (*FS28*), the SVR model stands out with the most substantial improvements in all key performance indicators, marking it as the best-performing model under the I-NSGA-II framework. Conversely, for both outputs, the DT model demonstrates the lowest performance, illustrating the weakness of this algorithm in capturing the patterns effectively.

Table 3: Comparative analysis of modeling performance for compressive strength after 28 days (CS28) and flexural strength after 28 days (FS28) using informed feature selection with I-NSGA-II (I-FS) and normal feature selection with NSGA-II (FS). The tables display performance metrics such as R2 , Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), and Root Mean Square Error (RMSE) for various machine learning models.

		R^2 in %		*MAE* in MPa		*MAPE* in %		*RMSE* in MPa	
Models	Abb.	I-FS	FS	I-FS	FS	I-FS	FS	I-FS	FS
Multiple Linear Regression	**MLR**	**76.21**	**72.47**	**4.66**	**5.26**	**4.20**	**4.79**	**5.65**	**6.27**
Partial Least Squares	PLS	75.59	71.35	4.71	5.37	4.26	4.87	5.71	6.40
Kernel Ridge Regression	**KRR**	**76.23**	**62.38**	**4.54**	**6.00**	**4.10**	**5.46**	**5.66**	**7.33**
K-Nearest Neighbors	KNN	70.61	63.25	5.28	5.91	4.89	5.39	6.29	7.25
Support Vector Regression	SVR	73.18	68.93	5.05	5.54	4.56	5.05	6.00	6.66
Decision Tree	DT	60.21	55.34	5.88	6.50	5.34	5.90	7.32	7.99
Random Forest	RF	72.21	64.98	5.01	5.94	4.55	5.43	6.12	7.08
Gradient Boosting	**GB**	**71.53**	**60.77**	**5.01**	**6.00**	**4.54**	**5.46**	**6.20**	**7.49**
Extreme Gradient Boosting	XGB	73.76	65.77	4.76	5.70	4.30	5.18	5.95	7.00
Gaussian Process Regression	GPR	75.36	67.47	4.76	5.73	4.31	5.22	5.71	6.82

(a) Prediction Performance of Various Machine Learning Models For Compressive Strength After 28 Days (*CS28*)

		R^2 in %		*MAE* in MPa		*MAPE* in %		*RMSE* in MPa	
Models	Abb.	I-FS	FS	I-FS	FS	I-FS	FS	I-FS	FS
Multiple Linear Regression	MLR	78.06	74.02	1.15	1.53	8.09	8.64	1.51	1.99
Partial Least Squares	PLS	78.29	72.49	1.14	1.67	8.03	9.26	1.51	2.04
Kernel Ridge Regression	KRR	78.33	73.10	1.07	1.63	8.00	8.90	1.50	2.02
K-Nearest Neighbors	KNN	76.66	73.00	1.22	1.56	8.93	8.71	1.60	1.99
Support Vector Regression	**SVR**	**81.75**	**72.62**	**0.93**	**1.56**	**7.08**	**8.54**	**1.37**	**2.04**
Decision Tree	DT	75.60	71.08	1.18	1.66	8.30	9.19	1.60	2.09
Random Forest	RF	78.80	71.77	1.09	1.67	8.09	9.31	1.49	2.07
Gradient Boosting	GB	77.10	72.26	1.05	1.60	7.82	8.91	1.55	2.05
Extreme Gradient Boosting	XGB	79.46	70.36	1.05	1.59	7.82	8.91	1.51	2.12
Gaussian Process Regression	**GPR**	**81.70**	**72.67**	**0.96**	**1.57**	**7.23**	**8.58**	**1.37**	**2.04**

(b) Prediction Performance of Various Machine Learning Models For Flexural Strength After 28 Days (*FS28*)

4.3.2. Impact of I-NSGA-II on Model Interpretability, Solution Stability, and Feature Selection

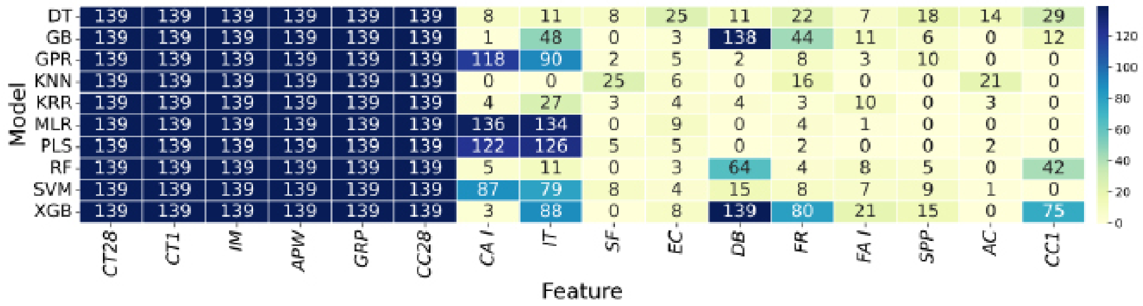
The results illustrated in Figures10and11demonstrate the impact of incorporating predefined feature importances on model interpretability within feature selection algorithms. Data from two experimental setups – one utilizing predefined feature importances through I-NSGA-II and the other without, using standard NSGA-II – were collected and analyzed. Each setup involved 139 LOOCV runs across various models, with feature selection frequencies recorded for each model, as discussed in Section 4.3.1.

Figure 10a demonstrates that the first six features, defined a priori as critical (I-NSGA-II), are invariably selected with the highest frequency (139 times) across all model evaluations. In contrast, under the NSGA-II scenario, depicted in Figure 10b, which lacks predefined feature guidance, the same six features also emerge as the most frequently selected. This consistent trend underscores the accuracy of the initial feature importance assessment by the E-FID method. Such parallelism in results validates the initial assumption about the critical nature of these features, thus supporting the effectiveness of informed feature preselection in I-NSGA-II.

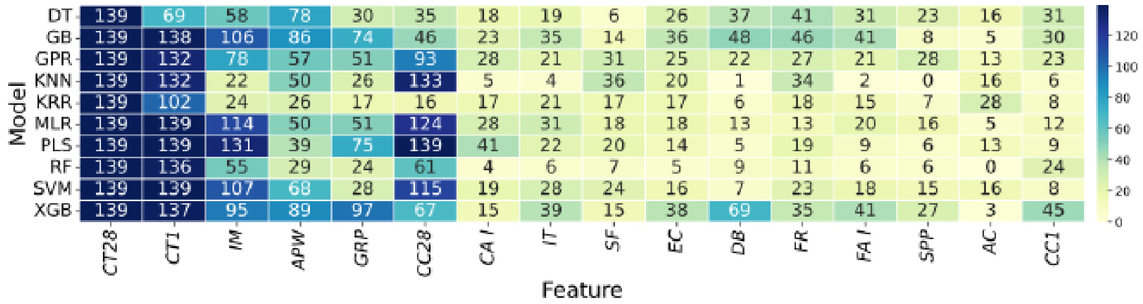
A notable divergence between the I-NSGA-II and the standard NSGA-II is observed in the stability of feature selection. I-NSGA-II consistently identifies the predefined features in every model iteration, reflecting enhanced stability and reliability in feature selection. This consistency is absent in the NSGA-II approach, where feature selection exhibits higher variability. This indicates potential instability and unpredictability in model performance without the injection of prior knowledge, which is a critical point of view for evaluating every algorithm.

Moreover, additional features such as Sand Type I (*SAI*), Ingredient Temperature (*IT*), and Material Delivery Batch Time (*DB*) exhibit significantly higher selection frequencies in the I-NSGA-II models. This observation suggests that the algorithm not only reinforces the importance of predefined features but also effectively identifies and elevates other relevant features based on the dataset's intrinsic characteristics.

Conversely, features such as Superplasticizer (*SPP*) and Initial Curing Conditions (*CC1*), which exhibit low or zero selection frequencies in some models under I-NSGA-II, highlight the algorithm's capacity to deprioritize less impactful features.



(a) Heatmap illustrating the frequency of feature selection in models employing I-NSGA-II for Compressive Strength at 28 days (*CS28*). Features predefined as critical consistently appear with high selection frequency, demonstrating the stability and focus of the I-NSGA-II algorithm.



(b) Heatmap displaying the frequency of feature selection in models using NSGA-II for *CS28*. This shows more variability in feature selection, highlighting the non-deterministic nature of NSGA-II without predefined feature guidance.

Figure 10: Comparative Analysis of Feature Selection Frequencies in I-NSGA-II and NSGA-II for CS28 : It highlights the increased stability and efficacy of feature selection when domain-specific knowledge is incorporated via I-NSGA-II. The comparison of the results demonstrates the reliability of E-FID for integrating priors into I-NSGA-II. For details about the variables, refer to Table 1. (I-NSGA-II: Informed Non-Dominated Sorting Genetic Algorithm II, CS28 : Compressive Strength at Day 28)

This selective enhancement by I-NSGA-II not only augments the model's interpretability but also clearly delineates which features are consistently valuable. It takes into account the interactions with predefined features to boost prediction performance while simultaneously preserving model simplicity and ensuring stability in the solutions.

In case of*FS28*, the analysis presented in Figure11reveals distinct differences in feature selection patterns between the two testing scenarios. In the I-NSGA-II scenario (Figure 11a), features such as Curing Temperature from day 2 to day 28 (*CT28*), Curing Conditions from day 2 to day 28 (*CC28*), Ingredient Moisture (*IM*), Initial Curing Temperature (*CT1*), Air Content (*AC*), and Material Delivery Batch Time (*DB*) show maximum selection frequency (139 times) across all models. This uniformity indicates that these features are consistently deemed crucial when predefined importances are considered, suggesting a strong alignment with the predefined importances and highlighting the influence of domain knowledge in guiding the selection process.

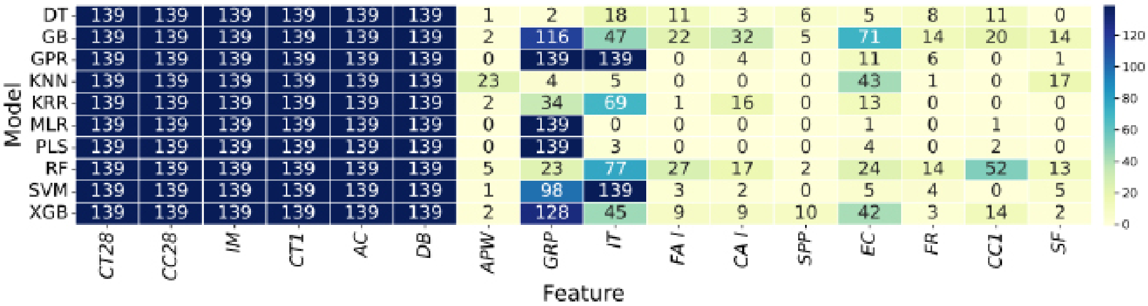
Conversely, the NSGA-II scenario (Figure 11b) demonstrates more variability in feature selection. Features such as *IM*, *CT1*, *AC*, and *DB*, while still frequently selected, show reduced counts compared to the I-NSGA-II scenario.

Additionally, the interaction assessed by I-NSGA-II reveals that Average Power Consumption (*APW*), although significant in the NSGA-II scenario, is less emphasized in I-NSGA-II due to the more critical interactions with predefined features. Conversely, the importance of Graphite (*GRP*) and especially Electrical Conductivity (*EC*) appears less crucial by NSGA-II but shows relatively good interactions with predefined features by I-NSGA-II.

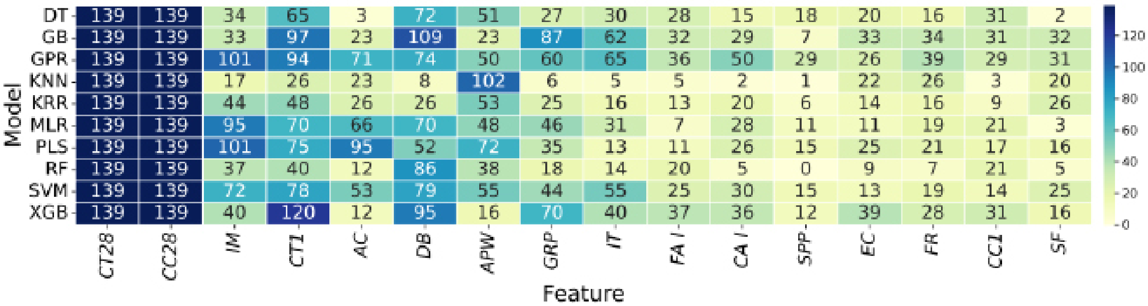
Furthermore, interesting results emerge from the comparison of both scenarios – using I-NSGA-II and NSGA-II

– as illustrated in Figures 10 and 11 for both *CS28* and *FS28*. Notably, I-NSGA-II sets the frequency of selection for some factors to zero in many cases. By comparing both scenarios, it can be concluded that I-NSGA-II tends to definitively decide whether a feature is selected or not, which leads to more stable solutions, higher accuracy in prediction performance, and better interpretability for use cases.

As discussed in Section 4.3.1, the selected algorithm for *CS28* is MLR and for *FS28* is SVR when employing I-NSGA-II, due to their superior prediction accuracy. From Figure 10, the MLR model selects features *CT28*, *CT1*, *IM*, *APW*, *GRP*, *CC28*, *SAI*, *IT*, *EC*, *FR*, and *FLI* for their critical importance in the subsequent investigation phase of the modeling process. Similarly, for SVR, as illustrated in Figure 11, the features *CT28*, *CT1*, *IM*, *APW*, *GRP*, *CC28*, *SAI*, *IT*, *DB*, *SPP*, *SF*, *FR*, *FLI*, and *EC* are selected for the next phase of investigation due to their pivotal roles.



(a) Heatmap illustrating the frequency of feature selection by I-NSGA-II across all models for FS28, showing high selection consistency for predefined features.



(b) Heatmap displaying the frequency of feature selection by NSGA-II across all models for *FS28*. Variability in feature selection underlines the challenges of NSGA-II without domain-specific guidance, resulting in less consistent feature prioritization.

Figure 11: Comparative Analysis of Feature Selection Frequencies in the Models Using I-NSGA-II versus NSGA-II for FS28 : These heatmaps highlight the impact of incorporating predefined feature importances on the stability and reliability of the feature selection process. The variables are explained in Table 1. (I-NSGA-II: Informed Non-Dominated Sorting Genetic Algorithm II, FS28 : Flexural Strength at Day 28)

4.4. UHPC Manufacturing Process Modeling

The IEM-FAS framework identified the MLR algorithm as the optimal choice for predicting *CS28*. As illustrated in Table 3a and Figure 10, a set of 11 critical features were selected to enhance the accuracy of the MLR model. In contrast, for predicting *FS28*, the framework recommended the SVR algorithm, supported by a distinct pool of 14 significant features (Table 3b, Figure 11).

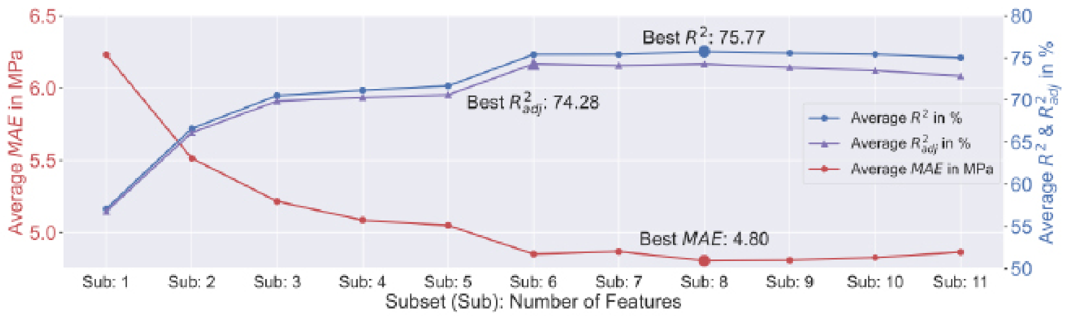
The modeling process (Algorithm 2) for *CS28* commenced with the inclusion of the most influential factor, *CT28*.

Utilizing *CT28* as a solitary predictor yielded an $R2$ value of 57.10 %, demonstrating the substantial role of curing temperature in explaining the variance of *CS28*. Subsequently, incorporating the first 24 hours of curing temperature (*CT1*) led to a significant enhancement in model performance, increasing $R2$ to 66.61 %. This improvement suggests that *CT1* provides additional variance information not captured by *CT28*. The further inclusion of the impact factor (*IM*) elevated the $R2$ to 70.53 %, indicating its critical contribution to the predictive model.

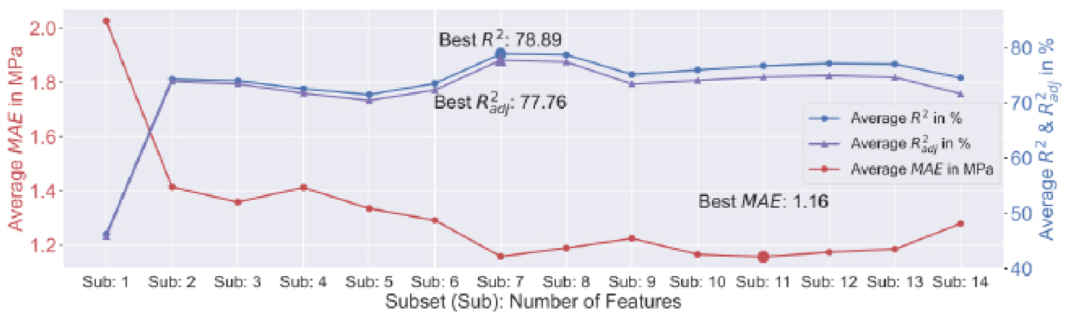
The addition of (*APW*) resulted in a slight performance boost, with $R2$ increasing to 71.16 %. Incorporating (*GRP*) and (*CC28*) further refined the model, yielding a notable $R2$ improvement to 75.40 %. Although the inclusion of additional features such as *SAI* and *IT* only marginally increased $R2$ to 75.77 %, these features were retained based on domain expertise, recognizing their potential significance in practical scenarios. Ultimately, the model achieved its highest adjusted $R2$ of 74.28 % with a core subset of six predictors: *CT28*, *CT1*, *IM*, *APW*, *GRP*, and *CC28*. This subset represents a balance between model accuracy and computational efficiency, highlighting the key variables necessary for optimal *CS28* prediction.

In the modeling of *FS28*, the process began with the inclusion of *CT28*, which yielded an initial average $R2$ of 46.23 %, indicating the significant impact of temperature on flexural strength at day 28. The subsequent inclusion of *CC28* substantially improved the model's performance, raising the average $R2$ to 74.32 %. This notable improvement underscores the importance of the interaction between curing conditions after 24 hours until day 28 in predicting *FS28*. However, the introduction of additional variables such as *IM*, *CT1*, and *AC* led to fluctuations in model accuracy. Specifically, the incorporation of *IM* slightly decreased the average $R2$ to 74.02 %, while *CT1* and *AC* further reduced it to 72.56 % and 71.51 %, respectively. These variations suggest that while some features introduce valuable new information, others may contribute to model complexity without a corresponding increase in predictive power. The highest average $R2$ of 78.89 % was achieved with a specific combination of features, including *DB* and *IT*, highlighting their importance in enhancing model accuracy.

These findings (Figure 12) emphasize the critical role of curing conditions in optimizing the mechanical properties of UHPC. Moreover, factors such as delivery batch timing and raw material storage conditions, which affect material moisture and temperature, significantly influence the quality of UHPC. Additionally, measurement errors in key materials, such as sand and impurities in silica fume (simulated as Graphite content), have a substantial impact on UHPC performance.



(a) Prediction Performance of MLR for *CS28*: Average R^2, R^2_{adj}, and *MAE* in the test phase as the number of features increases. The Sub:6 consists of *CT28*, *CT1*, *IM*, *APW*, *GRP*, and *CC28*. The Sub:8 consists of all features from Sub:6 plus *SAI* and *IT*.



(b) Prediction Performance of SVR for *FS28*: Average R^2, R^2_{adj}, and *MAE* in the test phase as the number of features increases. The Sub:7 consists of *CT28*, *CC28*, *IM*, *CT1*, *AC*, *DB*, and *IT*.

Figure 12: Comparative Performance Evaluation of MLR (Multiple Linear Regression) and SVR (Support Vector Regression) in the Test Phase: This figure highlights the impact of adding features on the prediction metrics for Compressive Strength at day 28 (CS28) and Flexural Strength at day 28 (FS28). The variables are explained in Table 1.

The proposed modeling strategy offers a robust approach to improving UHPC quality, particularly in the event of production faults. By enabling real-time prediction of mechanical properties, this strategy allows for prompt adjustments to the UHPC mixture to ensure target performance values are met. If predictions indicate that the mixture will not achieve the desired properties, corrective actions can be taken in two primary ways:

• Modifying the fresh mix based on the identified importance of each parameter, followed by additional mixing and re-evaluation of the predicted outcomes.

• Adjusting the curing regime, including conditions and temperature, to further enhance the mechanical properties of the UHPC.

5. Conclusions and Future Work

This study presents an Informed Automatic Modeling Pipeline, spanning from the Design of Experiments to the modeling phase, aimed at predicting the mechanical properties of UHPC in real-time. By adopting a holistic approach to UHPC manufacturing, the pipeline addresses the challenge of replicating UHPC products with consistent mechanical properties using the same recipe, despite inherent uncertainties in the production process. This comprehensive perspective on UHPC manufacturing and its impact on mechanical properties is, to the authors' knowledge, uniquely addressed in this work. The research contributes to a larger project aimed at developing a self-healing production system for the construction industry, capable of continuously monitoring UHPC quality and recommending real-time corrective actions.

Dueto the lack of datasets with a holistic view of the UHPC manufacturing process, 150 experiments were designed and conducted at the laboratory of G.tecz Engineering GmbH. The limited number of experiments, coupled with the complex nature of the manufacturing process, resulted in data sparsity. To mitigate this challenge, the study emphasizes dimensionality reduction and feature selection for modeling UHPC's mechanical properties.

A key contribution of this research is addressing the significant challenges associated with MOFS in high-dimensional contexts. These challenges include an exponentially expanding search space, ambiguity in identifying optimal interactions in complex and sparse datasets, and conflicts among objectives. The development ofthe I-NSGA- II, which incorporates insights from the E-FID framework into the traditional NSGA-II algorithm, effectively addresses these issues. The I-NSGA-II not only overcomes the instability typically associated with MOFS in high-dimensional, limited-sample scenarios but also enhances the interpretability and stability of feature selection.

The findings demonstrate that the I-NSGA-II outperforms the standard NSGA-II in two critical aspects. First, it achieves superior prediction performance. Second, it improves the interpretability of the models and the consistency of the feature selection process. Specifically, I-NSGA-II stabilizes feature selection frequency, either consistently selecting or excluding features across all iterations, in contrast to the considerable variability observed with NSGA-II.

The analysis revealed that curing temperature and curing humidity are the most critical factors influencing UHPC quality. Additional key parameters include sand (in terms of content and particle size distribution), graphite content (as an impurity in silica fume), and the moisture and temperature conditions during raw material storage. These findings highlight the necessity of careful control over these variables to improve UHPC quality.

The proposed modeling strategy can significantly improve UHPC quality control by addressing potential production faults and enabling real-time prediction of mechanical properties. This allows mixer operators to assess whether the UHPC mixture will meet the target specifications.

While the results are promising, this study is limited to datasets concerning the compressive and flexural strength of UHPC. Future research should explore a broader range of datasets to validate and refine the proposed methodologies in real-world UHPC manufacturing settings. Addressing these aspects will enable the developed framework to contribute more comprehensively to the field of concrete production, ensuring higher quality and performance across various concrete types.

Data availability

Research dataset (generated by the Welch test function [9, 10]) will be made available upon request. Our experimental datasets, supplied by our industry project partner, contains sensitive and confidential information and, therefore, cannot be publicly disclosed.

Acknowledgment

This project [HA project no. 1011/21-13] is financed with funds of LOEWE – Landes-Offensive zur Entwicklung Wissenschaftlich-ökonomischer Exzellenz, Förderlinie 3: KMU-Verbundvorhaben (State Offensive for the Develop- ment of Scientific and Economic Excellence).

LOEWE Exzellente Forschung für Hessens Zukunft

HESSEN Hessisches Ministerium für Wissenschaft und Kunst

Appendix: Evaluation ofI-NSGA-II on Data Generated by the Welch Test Function

The Welch test function, characterized by its complex interactions and nonlinear effects, serves as a benchmark for evaluating optimization algorithms [9,10]. The function y is defined as follows:

$$y = \frac{5x_{12}}{1+x_1} + 5(x_4 - x_{20})^2 + x_5 + 40x_{19}^3 - 5x_{19} + 0.05x_2 + 0.08x_3 - 0.03x_6 + 0.03x_7 \\ - 0.09x_9 - 0.01x_{10} - 0.07x_{11} + 0.25x_{13}^2 - 0.04x_{14} + 0.06x_{15} - 0.01x_{17} - 0.03x_{18}, \tag{20}$$

where the input domain is $x_i \in$ [-0:5; 0:5] for $i = 1; \ldots ; 20$. From the Welch function (Equation 20), it is evident that inputs $x1, x4, x5, x12, x13, x_{19}$, and x_{20} are the most significant features.

To assess the performance ofthe I-NSGA-II algorithm on data generated by the Welch test function ($X \in \mathbf{R}250{\times}20$), the most significant features were identified using the E-FID framework ($x_4, x_{12}, x_{13}, x_{19}, x_{20}$) and incorporated as prior knowledge into the IEM-FAS framework. The optimization process was conducted using advanced machine learning algorithms, and the results comparing I-NSGA-II with the classical NSGA-II are presented in Table 4 and Figure 14. The results show that I-NSGA-II outperforms classical NSGA-II in two key aspects: prediction accuracy and solution stability.

As demonstrated in Table4, the I-NSGA-II significantly enhances the prediction accuracy of models. For instance, the SVR model's $R2$ value improved from 66.74 % with NSGA-II to 81.62 % with I-NSGA-II. Regarding the stability of solutions across different algorithm initializations and data partitioning, Figure 14aillustrates the frequency of feature selection by I-NSGA-II, while Figure 14b shows the selection frequency using classical NSGA-II over 250 iterations in a LOOCV strategy. The results indicate that I-NSGA-II tends to produce more stable solutions compared to classical NSGA-II. For example, in both MLR and SVR models, the importance of features x_1 and x_5 is more pronounced when using I-NSGA-II. These features are also crucial according to the Welch function (Equation 20), demonstrating that I-NSGA-II is more likely to identify significant features with strong interactions with predefined features compared to classical NSGA-II.

Table 4: Comparative analysis of modeling performance on data generated by the Welch test function

using informed feature selection with I-NSGA-II (I-FS) and normal feature selection with NSGA-II (FS). The table displays performance metrics such as R2 , Mean Absolute Error (MAE), and Root Mean Square Error (RMSE) for various machine learning models.

		R^2 in %		MAE in MPa		RMSE in MPa	
Models	Abb.	I-FS	FS	I-FS	FS	I-FS	FS
Multiple Linear Regression	**MLR**	**84.15**	**78.88**	**0.6959**	**0.7973**	**0.8481**	**0.9789**
Kernel Ridge Regression	KRR	81.98	74.08	0.7255	0.8651	0.9042	1.0845
Support Vector Regression	**SVR**	**81.62**	**66.74**	**0.7163**	**0.9329**	**0.9132**	**1.2284**
Decision Tree	DT	49.85	46.76	1.1592	1.1833	1.5085	1.5542
Random Forest	RF	71.01	68.81	0.8814	0.9016	1.1470	1.1896

References:

T. Czigler, S. Reiter, P. Schulze, K. Somers, Laying the foundation for zero-carbon cement, volume 9, McKinsey & Company, 2020.

A. Abrishambaf, M. Pimentel, S. Nunes, C. Costa, Multi-level study on uhpfrc incorporating ecat, volume 318, Construction and Building Materials, 2022, p. 125976. URL: http://dx.doi.org/10.1016/j.conbuildmat.2021.125976. doi:10.1016/j.conbuildmat.2021.125976.

S. Abbas, M. L. Nehdi, M. A. Saleem, Ultra-high performance concrete: Mechanical performance, durability, sustainability and implementation challenges, volume 10, International Journal of Concrete Structures and Materials, 2016, pp. 271–295. URL: http://dx.doi.org/10.1007/s40069-016-0157-4. doi:10.1007/s40069-016-0157-4.

J. Du, W. Meng, K. H. Khayat, Y. Bao, P. Guo, Z. Lyu, A. Abu-Obeidah, H. Nassif, H. Wang, New development of ultra-high-performance concrete (UHPC), volume 224, Composites Part B: Engineering, 2021, p. 109220.

M. T. Marvila, A. R. G. de Azevedo, P. R. de Matos, S. N. Monteiro, C. M. F. Vieira, Materials for production of high and ultra- high performance concrete: Review and perspective of possible novel materials, volume 14, Materials, 2021, p. 4304. URL: http://dx.doi.org/10.3390/ma14154304. doi:10.3390/ma14154304.

F. Rezazadeh P., A. Dürrbaum, G. Zimmermann, A. Kroll, Leveraging ensemble structures to elucidate the impact of factors that influence the quality of ultra-high performance concrete, 2023 IEEE Symposium Series on Computational Intelligence (SSCI), 2023, pp. 180–187. URL: http://dx.doi.org/10.1109/ssci52147.2023.10371800. doi:10.1109/ssci52147.2023.10371800.

M. Amoozegar, B. Minaei-Bidgoli, Optimizing multi-objective PSO based feature selection method using a feature elitism mechanism, volume 113, Expert Systems with Applications, 2018, pp. 499–514. URL: http://dx.doi.org/10.1016/j.eswa.2018.07.013. doi:10.1016/j.eswa.2018.07.013.

R. Jiao, B. H. Nguyen, B. Xue, M. Zhang, A survey on evolutionary multiobjective feature selection in classification: Approaches, applications, and challenges, IEEE Transactions on Evolutionary Computation, 2024, p. 1. URL: http://dx.doi.org/10.1109/tevc.2023.3292527. doi:10.1109/tevc.2023.3292527.

E. N. Ben-Ari, D. M. Steinberg, Modeling data from computer experiments: An empirical comparison of kriging with mars and projection pursuit regression, volume 19, Quality Engineering, 2007, p. 327–338. URL: http://dx.doi.org/10.1080/08982110701580930. doi:10.1080/08982110701580930.

W. J. Welch, R. J. Buck, J. Sacks, H. P. Wynn, T. J. Mitchell, M. D. Morris, Screening, predicting, and computer experiments, volume 34, Technometrics, 1992, p. 15. URL: http://dx.doi.org/10.2307/1269548. doi:10.2307/1269548.

P. Ziolkowski, M. Niedostatkiewicz, Machine learning techniques in concrete mix design, volume 12, Materials, 2019, p. 1256. URL: http://dx.doi.org/10.3390/ma12081256. doi:10.3390/ma12081256.

S. Popovics, Analysis of concrete strength versus water-cement ratio relationship, volume 87, ACI Materials Journal, 1990, pp. 517–529. URL: http://dx.doi.org/10.14359/1944. doi:10.14359/1944.
M. Zain, S. M. Abd, K. Sopian, M. Jamil, A. Che-Ani, Mathematical regression model for the prediction of concrete strength, volume 10, WSEAS, Mathematics and Computers in Science and Engineering, 2008, pp. 313–330.
D. Dutta, S. V. Barai, Prediction of compressive strength of concrete: machine learning approaches, Springer, Recent Advances in Structural Engineering, Volume 1: Select Proceedings of SEC 2016, 2019, pp. 503–513.
E. Ozbay, A. Oztas, A. Baykasoglu, H. Ozbebek, Investigating mix proportions of high strength self compacting concrete by using Taguchi method, volume 23, Construction and Building Materials, 2009, pp. 694–702. URL: http://dx.doi.org/10.1016/j.conbuildmat.2008.02.014. doi:10.1016/j.conbuildmat.2008.02.014.
A. Farzampour, Compressive behavior of concrete under environmental effects, in: Compressive Strength of Concrete, IntechOpen, 2020, pp. 92–104. URL: http://dx.doi.org/10.5772/intechopen.85675. doi:10.5772/intechopen.85675.
K. Safranek, Influence of different mixing processes on the strength of ultra-high strength concretes, repositUm, Vienna University of Technology, 2007. URL: https://repositum.tuwien.at/handle/20.500.12708/10881.
I.-C. Yeh, Modeling of strength of high-performance concrete using artificial neural networks, volume 28, Cement and Concrete Research, 1998, pp. 1797–1808. URL: http://dx.doi.org/10.1016/s0008-8846(98)00165-3. doi:10.1016/s0008-8846(98)00165-3.
I.-C. Yeh, Modeling slump flow of concrete using second-order regressions and artificial neural networks, volume 29, Cement and Concrete Composites, 2007, pp. 474–480. URL: http://dx.doi.org/10.1016/j.cemconcomp.2007.02.001. doi:10.1016/j.cemconcomp.2007.02.001.
A. Kabir, M. Hasan, K. Miah, Predicting 28 days compressive strength of concrete from 7 days test result, Proceedings of the International Conference on Advances in Design and Construction of Structures, 2012, pp. 18–22.
H. Y. Aydogmus, H. I. Erdal, O. Karakurt, E. Namli, Y. S. Turkan, H. Erdal, A comparative assessment of bagging ensemble models for modeling concrete slump flow, volume 16, Computers and Concrete, 2015, pp. 741–757. URL: http://dx.doi.org/10.12989/cac.2015.16.5.741. doi:10.12989/cac.2015.16.5.741.
R. Jin, Q. Chen, A. B. Soboyejo, Non-linear and mixed regression models in predicting sustainable concrete strength, volume 170, Construction and Building Materials, 2018, pp. 142–152. URL: http://dx.doi.org/10.1016/j.conbuildmat.2018.03.063. doi:10.1016/j.conbuildmat.2018.03.063.
H. U. Ahmed, A. A. Abdalla, A. S. Mohammed, A. A. Mohammed, Mathematical modeling techniques to predict the compressive strength of high-strength concrete incorporated metakaolin with multiple mix proportions, volume 5, Cleaner Materials, 2022, p. 100132. URL: http://dx.doi.org/10.1016/j.clema.2022.100132. doi:10.1016/j.clema.2022.100132.
A. Marani, A. Jamali, M. L. Nehdi, Predicting ultra-high-performance concrete compressive strength using tabular generative adversarial networks, volume 13, Materials, 2020, p. 4757. URL: http://dx.doi.org/10.3390/ma13214757. doi:10.3390/ma13214757.
M. H. Rafiei, W. H. Khushefati, R. Demirboga, H. Adeli, Neural network, machine learning, and evolutionary approaches for concrete material characterization, volume 113, ACI Materials Journal, 2016, pp. 781–789. URL: http://dx.doi.org/10.14359/51689360. doi:10.14359/51689360.
B. A. Young, A. Hall, L. Pilon, P. Gupta, G. Sant, Can the compressive strength of concrete be estimated from knowledge of the mixture proportions?: New insights from statistical analysis and machine learning methods, volume 115, Cement and Concrete Research, 2019, pp. 379–388. URL: http://dx.doi.org/10.1016/j.cemconres.2018.09.006. doi:10.1016/j.cemconres.2018.09.006.
M. Shi, W. Shen, Automatic modeling for concrete compressive strength prediction using auto-sklearn,

volume 12, Buildings, 2022, p. 1406. URL: http://dx.doi.org/10.3390/buildings12091406. doi:10.3390/buildings12091406.

J. Yu, R. Pan, Y. Zhao, High-dimensional, small-sample product quality prediction method based on MIC-stacking ensemble learning, volume 12, Applied Sciences, 2021, p. 23. URL: http://dx.doi.org/10.3390/app12010023. doi:10.3390/app12010023.

F. Rezazadeh, A. Kroll, Predicting the compressive strength of concrete up to 28 days-ahead: comparison of 16 machine learning algorithms on benchmark datasets, 32. Workshop Computational Intelligence, 2022, pp. 53–75.

N.-H. Nguyen, J. Abellán-García, S. Lee, E. Garcia-Castano, T. P. Vo, Efficient estimating compressive strength of ultra-high performance concrete using xgboost model, volume 52, Journal of Building Engineering, 2022, p. 104302. URL: http://dx.doi.org/10.1016/j.jobe.2022.104302. doi:10.1016/j.jobe.2022.104302.

E. Saleh, A. Tarawneh, M. Naser, M. Abedi, G. Almasabha, You only design once (yodo): Gaussian process-batch Bayesian optimization framework for mixture design of ultra high performance concrete, volume 330, Construction and Building Materials, 2022, p. 127270. URL: http://dx.doi.org/10.1016/j.conbuildmat.2022.127270. doi:10.1016/j.conbuildmat.2022.127270.

S. Mahjoubi, W. Meng, Y. Bao, Auto-tune learning framework for prediction of flowability, mechanical properties, and porosity of ultra- high-performance concrete (UHPC), volume 115, Applied Soft Computing, 2022, p. 108182. URL: http://dx.doi.org/10.1016/j.asoc.2021.108182. doi:10.1016/j.asoc.2021.108182.

E. M. Golafshani, A. Behnood, T. Kim, T. Ngo, A. Kashani, A framework for low-carbon mix design of recycled aggregate concrete with supplementary cementitious materials using machine learning and optimization algorithms, volume 61, Structures, 2024, p. 106143. URL: http://dx.doi.org/10.1016/j.istruc.2024.106143. doi:10.1016/j.istruc.2024.106143.

Z. Li, J. Yoon, R. Zhang, F. Rajabipour, W. V. Srubar III, I. Dabo, A. Radlińska, Machine learning in concrete science: applications, challenges, and best practices, volume 8, npj Computational Materials, 2022. URL: http://dx.doi.org/10.1038/s41524-022-00810-x. doi:10.1038/s41524-022-00810-x.

F. Rezazadeh P., A. Dürrbaum, G. Zimmermann, A. Kroll, Holistic modeling of ultra-high performance concrete production process: Synergizing mix design, fresh concrete properties, and curing conditions, 33. Workshop Computational Intelligence, 2023, pp. 215–237. URL: http://dx.doi.org/10.58895/ksp/1000162754-15. doi:10.58895/ksp/1000162754-15.

P. Chopra, R. K. Sharma, M. Kumar, Prediction of compressive strength of concrete using artificial neural network and genetic programming, volume 2016, Advances in Materials Science and Engineering, 2016, pp. 1–10. URL: http://dx.doi.org/10.1155/2016/7648467. doi:10.1155/2016/7648467.

R. Mustapha, E. A. Mohamed, High-performance concrete compressive strength prediction based weighted support vector machines, volume 07, International Journal of Engineering Research and Applications, 2017, pp. 68–75. URL: http://dx.doi.org/10.9790/9622-0701016875. doi:10.9790/9622-0701016875.

T. T. Nguyen, L. T. Ngoc, H. H. Vu, T. P. Thanh, Machine learning-based model for predicting concrete compressive strength, volume 20, International Journal of GEOMATE, 2021, pp. 197–204. URL: http://dx.doi.org/10.21660/2020.77.j2019. doi:10.21660/2020.77.j2019.

R. Rajeshwari, S. Mandal, Prediction of compressive strength of high-volume fly ash concrete using artificial neural network, Sustainable Construction and Building Materials, 2018, pp. 471–483. URL: http://dx.doi.org/10.1007/978-981-13-3317-0_42. doi:10.1007/978-981-13-3317-0_42.

P. Chopra, R. Sharma, M. Kumar, Ridge regression for the prediction of compressive strength of concrete, volume 2, Int. J. Innov. Eng. Technol, 2013, pp. 106–111.

K. Güçlüer, A. Özbeyaz, S. Göymen, O. Günaydın, A comparative investigation using machine learning methods for concrete compressive strength estimation, volume 27, Materials Today Communications, 2021, p. 102278. URL: http://dx.doi.org/10.1016/j.mtcomm.2021.102278. doi:10.1016/j.mtcomm.2021.102278.

H. Naderpour, A. H. Rafiean, P. Fakharian, Compressive strength prediction of environmentally friendly concrete using artificial neural networks, volume 16, Journal of Building Engineering, 2018, pp. 213–219. URL: http://dx.doi.org/10.1016/j.jobe.2018.01.007. doi:10.1016/j.jobe.2018.01.007.

I. B. Topcu, M. Sarıdemir, Prediction of compressive strength of concrete containing fly ash using artificial neural networks and fuzzy logic, volume 41, Computational Materials Science, 2008, pp. 305–311. URL: http://dx.doi.org/10.1016/j.commatsci.2007.04.009. doi:10.1016/j.commatsci.2007.04.009.

J. Sun, J. Zhang, Y. Gu, Y. Huang, Y. Sun, G. Ma, Prediction of permeability and unconfined compressive strength of pervious concrete using evolved support vector regression, volume 207, Construction and Building Materials, 2019, pp. 440–449. URL: http://dx.doi.org/10.1016/j.conbuildmat.2019.02.117. doi:10.1016/j.conbuildmat.2019.02.117.

R. Kumar, B. Rai, P. Samui, A comparative study of prediction of compressive strength of ultra-high performance concrete using soft computing technique, volume 24, Structural Concrete, 2023, pp. 5538–5555. URL: http://dx.doi.org/10.1002/suco.202200850. doi:10.1002/suco.202200850.

M. K. Keleş, A. E. Keleş, Ü. Kiliç, Prediction of concrete strength with data mining methods using artificial bee colony as feature selector, 2018 International Conference on Artificial Intelligence and Data Processing (IDAP), 2018, pp. 1–4. URL: http://dx.doi.org/10.1109/idap.2018.8620905. doi:10.1109/idap.2018.8620905.

T. Nguyen, A. Kashani, T. Ngo, S. Bordas, Deep neural network with high-order neuron for the prediction of foamed concrete strength, volume 34, Computer-Aided Civil and Infrastructure Engineering, 2018, pp. 316–332. URL: http://dx.doi.org/10.1111/mice.12422. doi:10.1111/mice.12422.

J. Zhang, G. Ma, Y. Huang, J. sun, F. Aslani, B. Nener, Modelling uniaxial compressive strength of lightweight self-compacting concrete using random forest regression, volume 210, Construction and Building Materials, 2019, pp. 713–719. URL: http://dx.doi.org/10.1016/j.conbuildmat.2019.03.189. doi:10.1016/j.conbuildmat.2019.03.189.

A. Sharafati, S. B. Haji Seyed Asadollah, N. Al-Ansari, Application of bagging ensemble model for predicting compressive strength of hollow concrete masonry prism, volume 12, Ain Shams Engineering Journal, 2021, pp. 3521–3530. URL: http://dx.doi.org/10.1016/j.asej.2021.03.028. doi:10.1016/j.asej.2021.03.028.

N.-D. Hoang, A.-D. Pham, Q.-L. Nguyen, Q.-N. Pham, Estimating compressive strength of high performance concrete with Gaussian process regression model, volume 2016, Advances in Civil Engineering, 2016, pp. 1–8. URL: http://dx.doi.org/10.1155/2016/2861380. doi:10.1155/2016/2861380.

N. K. Nagwani, S. V. Deo, Estimating the concrete compressive strength using hard clustering and fuzzy clustering based regression techniques, volume 2014, The Scientific World Journal, 2014, pp. 1–16. URL: http://dx.doi.org/10.1155/2014/381549. doi:10.1155/2014/381549.

D. Chakraborty, I. Awolusi, L. Gutierrez, An explainable machine learning model to predict and elucidate the compressive behavior of high-performance concrete, volume 11, Results in Engineering, 2021, p. 100245. URL: http://dx.doi.org/10.1016/j.rineng.2021.100245. doi:10.1016/j.rineng.2021.100245.

A. Kandiri, F. Sartipi, M. Kioumarsi, Predicting compressive strength of concrete containing recycled aggregate using modified ann with different optimization algorithms, volume 11, Applied Sciences, 2021, p. 485. URL: http://dx.doi.org/10.3390/app11020485. doi:10.3390/app11020485.

P. Refaeilzadeh, L. Tang, H. Liu, Cross-validation, Encyclopedia of Database Systems, 2009, pp. 532–538. URL: http://dx.doi.org/10.1007/978-0-387-39940-9_565. doi:10.1007/978-0-387-39940-9_565.

J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, H. Liu, Feature selection: A data perspective, volume 50, ACM computing surveys (CSUR), 2017, pp. 1–45.

B. Ahadzadeh, M. Abdar, F. Safara, A. Khosravi, M. B. Menhaj, P. N. Suganthan, SFE: A simple, fast, and efficient feature selection algorithm for high-dimensional data, volume 27, IEEE Transactions on

Evolutionary Computation, 2023, pp. 1896–1911. URL: http://dx.doi.org/10.1109/tevc.2023.3238420. doi:10.1109/tevc.2023.3238420.

L. Li, M. Xuan, Q. Lin, M. Jiang, Z. Ming, K. C. Tan, An evolutionary multitasking algorithm with multiple filtering for high-dimensional feature selection, volume 27, IEEE Transactions on Evolutionary Computation, 2023, pp. 802–816. URL: http://dx.doi.org/10.1109/tevc.2023.3254155. doi:10.1109/tevc.2023.3254155.

K. Chen, B. Xue, M. Zhang, F. Zhou, Evolutionary multitasking for feature selection in high-dimensional classification via particle swarm optimization, volume 26, IEEE Transactions on Evolutionary Computation, 2022, pp. 446–460. URL: http://dx.doi.org/10.1109/tevc.2021.3100056. doi:10.1109/tevc.2021.3100056.

K. Chen, B. Xue, M. Zhang, F. Zhou, An evolutionary multitasking-based feature selection method for high-dimensional classification, volume 52, IEEE Transactions on Cybernetics, 2022, pp. 7172–7186. URL: http://dx.doi.org/10.1109/tcyb.2020.3042243. doi:10.1109/tcyb.2020.3042243.

J. Luo, D. Zhou, L. Jiang, H. Ma, A particle swarm optimization based multiobjective memetic algorithm for high-dimensional feature selection, volume 14, Memetic Computing, 2022, pp. 77–93. URL: http://dx.doi.org/10.1007/s12293-022-00354-z. doi:10.1007/s12293-022-00354-z.

L. von Rueden, S. Mayer, K. Beckh, B. Georgiev, S. Giesselbach, R. Heese, B. Kirsch, M. Walczak, J. Pfrommer, A. Pick, R. Ramamurthy, J. Garcke, C. Bauckhage, J. Schuecker, Informed machine learning - a taxonomy and survey of integrating prior knowledge into learning systems, IEEE Transactions on Knowledge and Data Engineering, 2021, pp. 614–633. URL: http://dx.doi.org/10.1109/tkde.2021.3079836. doi:10.1109/tkde.2021.3079836.

I. Kropp, A. P. Nejadhashemi, K. Deb, Benefits of sparse population sampling in multi-objective evolutionary computing for large-scale sparse optimization problems, volume 69, Swarm and Evolutionary Computation, 2022, p. 101025. URL: http://dx.doi.org/10.1016/j.swevo.2021.101025. doi:10.1016/j.swevo.2021.101025.

H. Xu, B. Xue, M. Zhang, A duplication analysis-based evolutionary algorithm for biobjective feature selection, volume 25, IEEE Transactions on Evolutionary Computation, 2021, pp. 205–218. URL: http://dx.doi.org/10.1109/tevc.2020.3016049. doi:10.1109/tevc.2020.3016049.

X.-F. Song, Y. Zhang, D.-W. Gong, X.-Z. Gao, A fast hybrid feature selection based on correlation-guided clustering and particle swarm optimization for high-dimensional data, volume 52, IEEE Transactions on Cybernetics, 2022, pp. 9573–9586. URL: http://dx.doi.org/10.1109/tcyb.2021.3061152. doi:10.1109/tcyb.2021.3061152.

J. Ren, F. Qiu, H. Hu, Multiple sparse detection-based evolutionary algorithm for large-scale sparse multiobjective optimization problems, volume 9, Complex and Intelligent Systems, 2023, pp. 4369–4388. URL: http://dx.doi.org/10.1007/s40747-022-00963-8. doi:10.1007/s40747-022-00963-8.

P. Wang, B. Xue, J. Liang, M. Zhang, Multiobjective differential evolution for feature selection in classification, volume 53, IEEE Transactions on Cybernetics, 2023, pp. 4579–4593. URL: http://dx.doi.org/10.1109/tcyb.2021.3128540. doi:10.1109/tcyb.2021.3128540.

F. Rezazadeh P., A. Abrishambaf, A. Dürrbaum, G. Zimmermann, A. Kroll, Systematic data generation to study measurement errors and environmental impacts on ultra-high performance concrete, Submitted, Unpublished Results.

A. Dürrbaum, F. Rezazadeh P., A. Kroll, Automatic camera-based advanced slump flow testing for improved reliability, 2023 IEEE SENSORS, 2023, pp. 1–4. URL: http://dx.doi.org/10.1109/sensors56945.2023.10325030. doi:10.1109/sensors56945.2023.10325030.

G. Taguchi, System of experimental design; engineering methods to optimize quality and minimize costs, New York: UNIPUB/Kaus International, 1987.

M. Stein, Large sample properties of simulations using Latin hypercube sampling, volume 29, Technometrics, 1987, p. 143. URL: http://dx.doi.org/10.2307/1269769. doi:10.2307/1269769.

M. Mitchell, Genetic algorithms: An overview, An Introduction to Genetic Algorithms, 1998, pp. 31–39.

URL: http://dx.doi.org/10.7551/mitpress/3927.003.0003. doi:10.7551/mitpress/3927.003.0003.
A. Hedayat, Study of optimality criteria in design of experiments, Statistics and Related Topics, North-Holland Publishing Company, 1981, pp. 39–56.
J. Benesty, J. Chen, Y. Huang, I. Cohen, Pearson correlation coefficient, Springer Topics in Signal Processing, 2009, pp. 1–4. URL: http://dx.doi.org/10.1007/978-3-642-00296-0_5. doi:10.1007/978-3-642-00296-0_5.
A. R. T. Donders, G. J. van der Heijden, T. Stijnen, K. G. Moons, Review: A gentle introduction to imputation of missing values, volume 59, Journal of Clinical Epidemiology, 2006, pp. 1087–1091. URL: http://dx.doi.org/10.1016/j.jclinepi.2006.01.014. doi:10.1016/j.jclinepi.2006.01.014.
O. Troyanskaya, M. Cantor, G. Sherlock, P. Brown, T. Hastie, R. Tibshirani, D. Botstein, R. B. Altman, Missing value estimation methods for DNA microarrays, volume 17, Bioinformatics, 2001, pp. 520–525. URL: http://dx.doi.org/10.1093/bioinformatics/17.6.520. doi:10.1093/bioinformatics/17.6.520.
S. F. Buck, A method of estimation of missing values in multivariate data suitable for use with an electronic computer, volume 22, Journal of the Royal Statistical Society: Series B (Methodological), 1960, pp. 302–306. URL: http://dx.doi.org/10.1111/j.2517-6161.1960.tb00375.x. doi:10.1111/j.2517-6161.1960.tb00375.x.
S. v. Buuren, K. Groothuis-Oudshoorn, mice: Multivariate imputation by chained equations inr, volume 45, Journal of Statistical Software, 2011, pp. 1–67. URL: http://dx.doi.org/10.18637/jss.v045.i03. doi:10.18637/jss.v045.i03.
R. J. Little, D. B. Rubin, Statistical analysis with missing data, volume 793, John Wiley & Sons, 2019.
F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al., Scikit-learn: Machine learning in python, volume 12, Journal of Machine Learning Research, 2011, pp. 2825–2830.
C. M. Bishop, N. M. Nasrabadi, Pattern recognition and machine learning, volume 4, Springer, 2006.
D. J. C. MacKay, Bayesian Interpolation, Maximum Entropy and Bayesian Methods, 1992, pp. 39–66. URL: http://dx.doi.org/10.1007/978-94-017-2219-3_3. doi:10.1007/978-94-017-2219-3_3.
M. E. Tipping, Sparse Bayesian learning and the relevance vector machine, volume 1, Journal of machine learning research, 2001, pp. 211–244.
A. Smiti, A critical overview of outlier detection methods, volume 38, Computer Science Review, 2020, p. 100306. URL: http://dx.doi.org/10.1016/j.cosrev.2020.100306. doi:10.1016/j.cosrev.2020.100306.
X. Su, C.-L. Tsai, Outlier detection, volume 1, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2011, pp. 261–268.
V. Hodge, J. Austin, A survey of outlier detection methodologies, volume 22, Artificial Intelligence Review, 2004, pp. 85–126. URL: http://dx.doi.org/10.1023/b:aire.0000045502.10941.a9. doi:10.1023/b:aire.0000045502.10941.a9.
U. Habib, G. Zucker, M. Blochle, F. Judex, J. Haase, Outliers detection method using clustering in buildings data, IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society, 2015. URL: http://dx.doi.org/10.1109/iecon.2015.7392181. doi:10.1109/iecon.2015.7392181.
A. Boukerche, L. Zheng, O. Alfandi, Outlier detection: Methods, models, and classification, volume 53, ACM Computing Surveys, 2020, pp. 1–37. URL: http://dx.doi.org/10.1145/3381028. doi:10.1145/3381028.
H. Wang, M. J. Bah, M. Hammad, Progress in outlier detection techniques: A survey, volume 7, IEEE Access, 2019, pp. 107964–108000. URL: http://dx.doi.org/10.1109/access.2019.2932769. doi:10.1109/access.2019.2932769.
O. Sagi, L. Rokach, Ensemble learning: A survey, volume 8, WIREs Data Mining and Knowledge Discovery, 2018. URL: http://dx.doi.org/10.1002/widm.1249. doi:10.1002/widm.1249.
A. Krogh, J. Vedelsby, Neural network ensembles, cross validation, and active learning, volume 7, Advances in neural information processing systems, 1994.
D. Wood, T. Mu, A. M. Webb, H. W. Reeve, M. Lujan, G. Brown, A unified theory of diversity in

ensemble learning, volume 24, Journal of Machine Learning Research, 2023, pp. 1–49.
J. Surowiecki, The wisdom of crowds, Anchor, 2005.
K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, volume 6, IEEE Transactions on Evolutionary Computation, 2002, pp. 182–197. URL: http://dx.doi.org/10.1109/4235.996017. doi:10.1109/4235.996017.
N. Gunantara, A review of multi-objective optimization: Methods and its applications, volume 5, Cogent Engineering, 2018, p. 1502242. URL: http://dx.doi.org/10.1080/23311916.2018.1502242. doi:10.1080/23311916.2018.1502242.
F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, et al., Scikit-learn: Machine learning in Python, volume 12, 2011, pp. 2825–2830.
J. A. Wegelin, et al., A survey of partial least squares (PLS) methods, with emphasis on the two-block case, University of Washington, Tech. Rep, 2000.
K. P. Murphy, Machine learning: a probabilistic perspective, MIT press, 2012.
E. Fix, J. L. Hodges, Discriminatory analysis. nonparametric discrimination: Consistency properties, volume 57, International Statistical Review / Revue Internationale de Statistique, 1989, p. 238. URL: http://dx.doi.org/10.2307/1403797. doi:10.2307/1403797.
C.-C. Chang, A library for support vector machines, 2023. URL: https://www.csie.ntu.edu.tw/~cjlin/libsvm/.
M. Dumont, R. Marée, L. Wehenkel, P. Geurts, VISAPP, Proceedings of the Fourth International Conference on Computer Vision Theory and Applications, 2009. URL: http://dx.doi.org/10.5220/0001800001960203. doi::10.5220/0001800001960203.
L. Breiman, Random forests, volume 45, Machine learning, 2001, pp. 5–32.
J. H. Friedman, Greedy function approximation: a gradient boosting machine, JSTOR, 2001, pp. 1189–1232.
T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 785–794. URL: http://dx.doi org/10 1145/2939672 . doi...2939785:10.1145/ 2939672.2939785.
C. E. Rasmussen, Gaussian processes in machine learning, Summer school on machine learning, 2003, pp. 63–71.
Z. Ge, Z. Song, S. X. Ding, B. Huang, Data mining and analytics in the process industry: The role of machine learning, volume 5, IEEE Access, 2017, p. 20590–20616. URL:http://dx.doi.org/10.1109/access.2017.2756872. doi:10.1109/access.2017.2756872.