

Developing Machine Learning Algorithm for Investigating Features Affecting Relative Density and Shrinkage of Printed Parts in Binder Jetting Additive Manufacturing of Microjet TP-80 Ceramic Composite Powder

Suleiman Obeidat

Rahagir Ridwan Anik

Rakib Hasan

Syed Hasib Akhter Faruqui

shf006@shsu.edu

Sam Houston State University <https://orcid.org/0000-0002-5073-8690>

Research Article

Keywords: Binder Jetting Additive Manufacturing (BJAM), Ceramic Composites, Machine Learning, SHAP, Relative Density, Shrinkage Optimization

Posted Date: March 16th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-9073284/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at The International Journal of Advanced Manufacturing Technology on June 20th, 2026. See the published version at <https://doi.org/10.1007/s00170-026-18483-x>.

Developing Machine Learning Algorithm for Investigating Features Affecting Relative Density and Shrinkage of Printed Parts in Binder Jetting Additive Manufacturing of Microjet TP-80 Ceramic Composite Powder

Suleiman Obeidat

Department of Engineering Technology
Sam Houston State University
Huntsville, Tx, USA

<https://orcid.org/0000-0003-3856-8139>

Rahagir Ridwan Anik

Department of Engineering Technology
Sam Houston State University
Huntsville, Tx, USA

<https://orcid.org/0009-0005-4324-670X>

Rakib Hasan

Department of Mechanical, Aerospace,
and Industrial Engineering
University of Texas at San Antonio
San Antonio, Tx, USA

<https://orcid.org/0000-0001-6728-6638>

Syed Hasib Akhter Faruqui*

Department of Engineering Technology
Sam Houston State University
Huntsville, Tx, USA

<https://orcid.org/0000-0002-5073-8690>

Abstract

Binder jetting is a versatile 3D printing technique capable of producing complex geometries with various materials, including ceramics. However, the final properties of binder-jetted parts are affected by multiple process parameters, making optimization challenging and time-consuming. To address this, we developed a machine learning model to predict and optimize the key factors affecting part quality. This study explores the application of machine learning (ML) algorithms to investigate the factors influencing relative density and shrinkage in binder jetting additive manufacturing of ceramic composite powders. The ML algorithm was designed to identify the optimal printing parameters that maximize the relative density of printed parts while minimizing shrinkage in the final parts. Importantly, we employed SHAP algorithms to determine which parameters significantly impact shrinkage and part density. By leveraging this data-driven approach, we aim to reduce the need for extensive trial-and-error experiments, thereby saving time and resources in the manufacturing process. The results demonstrate the effectiveness of ML Algorithms in predicting and optimizing binder jetting outcomes for ceramic composite powders. Furthermore, the study revealed that ensemble methods, such as Bagging and Random Forest, were most effective in predicting relative density, with Bagging achieving the lowest MSE (0.0156 ± 0.085). For shrinkage prediction, simpler models like Linear Regression outperformed more complex approaches, achieving the lowest MSE (0.0028 ± 0.031). Feature importance analysis revealed that *Sample Geometry* and *Sample Actual Volume* were critical determinants of relative density, while *Sample Delay* had the most significant impact on shrinkage.

Keywords: Binder Jetting Additive Manufacturing (BJAM), Ceramic Composites, Machine Learning, SHAP, Relative Density, Shrinkage Optimization

1 Introduction

Binder Jetting (BJ) is a prominent additive manufacturing (AM) technique that creates parts by selectively depositing a binding agent onto a powder bed. Compared to other AM techniques like Fused Deposit Modeling (FDM) or Selective Laser Sintering (SLS), BJ does not require high-energy sources or heated nozzles. This reduces the complexity of the machine and the energy consumption. A wide range of materials, including metals, ceramics, and composites, can be used to print geometries using the BJ

*Corresponding Author. Email: shf006@shsu.edu

method. This makes it one of the most versatile methods to be used for various industrial applications, ranging from aerospace and automotive engineering to biomedical engineering.

One of the primary advantages of Binder Jetting lies in its capacity to produce complex geometries with high precision and surface quality. The layer-by-layer deposition process allows for the creation of parts with intricate internal structures, lightweight designs, and customized features that are challenging to achieve with traditional manufacturing methods. Additionally, BJ is capable of producing multiple parts simultaneously, enhancing production efficiency and reducing lead times for large-scale manufacturing. However, the final properties of binder-jetted parts are significantly influenced by several factors during the manufacturing process, such as binder saturation and number of prints [1, 2]. Additionally, time delays between printing stages, drying, de-powdering, de-binding, and sintering, can lead to inconsistencies and defects if not meticulously managed, while these steps are essential to obtain a part with good mechanical properties [1, 2]. Binder Jetting Additive Manufacturing (BJAM) works by spreading thin layers of powder and using precise droplets of binder to join the particles together. The success of the process depends heavily on how well the powder is spread, how uniformly the binder infiltrates the powder bed, and how the printed part is later debinded and sintered. If the powder layer is not evenly packed, the printed part will have variations in density that can lead to distortion and uneven shrinkage during sintering. Likewise, the amount of binder must be carefully controlled. Too little binder creates weak, brittle green parts, while too much causes binder bleeding and loss of dimensional accuracy. Once printing is complete, the green part goes through debinding and sintering, where ceramic particles bond together through solid-state diffusion. These fundamental steps determine the final density, microstructure, and overall quality of the part. A significant concern is the relative volume shrinkage of the parts during production. Predicting the extent of shrinkage is essential for adjusting the initial geometry to attain the desired final dimensions. Understanding the process parameters that influence shrinkage is also crucial.

To address these challenges, recent studies have been focusing on utilizing machine learning (ML) techniques to predict optimal printing process parameters that maximize desired properties in additively manufactured parts [3, 4]. Foundational works have further demonstrated the utility of ML for process monitoring and predictive modeling, thereby establishing a strong basis for current research directions in this field [5]. Research has demonstrated that by fine-tuning factors such as binder composition, layer thickness, and drying protocols, it is possible to enhance the relative volume shrinkage, density, and overall quality of BJ parts. To prevent repeated experiments that do not yield the best quality of parts, several researchers have implemented ML techniques to enhance part quality. For example, [6] utilized a micro-mechanics model combined with experimental data to determine elastic constants in the extrusion deposition additive manufacturing (EDAM) process. In binder jetting, [7] applied machine learning for automated porosity detection, achieving an 88% F1 score, while [8, 9] developed real-time quality inspection systems using camera images and X-ray computed tomography combined with ML to monitor and adjust the printing process automatically. Additionally, studies such as [10–15] have demonstrated the effectiveness of ML in predicting weld bead characteristics, monitoring material deposition in wire arc additive manufacturing (WAAM), and optimizing parameters in Laser Powder Bed Fusion (LPBF) and metal AM (MAM) processes. ML models have also been employed to predict stress-strain curves [16], quantify geometric imperfections [17], and analyze shrinkage and porosity relationships [18]. These advancements underscore the potential of machine learning to enhance process control, predict defects, and optimize manufacturing parameters across various additive manufacturing techniques, thereby improving the overall quality and efficiency of produced parts.

This study aims to systematically investigate the influence of process parameters on dimensional accuracy, and relative volume shrinkage of BJ printed parts. Specifically,

- **Quantify Shrinkage Effects:** Develop a predictive ML model that can predict the amount of relative volume shrinkage to be expected from a BJ printed part under the current printing settings and geometric specifications.
- **Quantify Relative Density:** Develop a predictive ML model that can predict the amount of relative density to be expected from a BJ printed part under the current printing settings and geometric specifications.
- **Identify Significant Printing Parameters:** Utilize explainable techniques like SHapley Additive exPlanations (SHAP) [19] to determine the process parameters that affect the desired properties in BJ printed parts.

The findings of this research are intended to advance BJ technology by improving its efficiency, reducing associated costs, and improving the quality and reliability of the parts produced. The remainder of the manuscript is structured as follows. Section 2 presents the preliminaries and background for the proposed analysis methodology. Section 3 discusses the results of ML models and identifies features that are important in predicting the shrinkage effect and relative density. Finally, Section 4 provides the concluding remarks for this research work.

2 Methodology

2.1 Binder Jetting Additive Manufacturing Process

The Binder Jetting Additive Manufacturing (BJAM) technology is employed in this work using the ComeTrue M10 CERAMIC & Binder Jetting 3D Printer from MicroJet Technology Co., Ltd. CAD models were created using computer-aided design (CAD) software and then converted into (.stl) format to prepare for printing. During the printing process, parts are constructed layer by layer, with each layer bonded by binder droplets deposited after the powder layer is laid down. The M10 3D printer features two bed systems: a feeding bed and a fabrication bed. The feeding bed prepares the powder through a tapping process to minimize pores and create a smooth surface. The fabrication bed is where the part is built. Printing begins by pouring a specific amount of ceramic powder into the feed bed based on the part's height. Both the feed bed and fabrication bed measure $200 \times 160 \times 150\text{mm}$. A plastic scraper is used to tap the powder in the feed bed, ensuring a smooth surface when the powder is spread to the fabrication bed. Once the powder is smooth, the printing process starts with a rotating roller, spreading the powder from the feed bed to the fabrication bed while the fabrication bed lowers by the thickness of one layer. Binder droplets are then selectively deposited through the printhead. The roller returns to the feed bed, which rises to allow the roller to spread another layer of powder onto the fabrication bed. From a manufacturing engineering point of view, every parameter in the BJAM process affects the final outcome. Layer thickness, powder spreading speed, and powder packing all influence the strength of the green part and how much it will shrink later. Binder saturation is equally important because it controls how well the layers hold together. A part that absorbs too much binder will swell or distort, while too little binder leads to delamination or weak bonding. The drying, debinding, and sintering steps then play a major role in how the part densifies and what defects may form. Heating too quickly can cause cracks, while the sintering temperature and hold time determine how the particles fuse, how much the part shrinks, and how dense and strong it becomes. These engineering principles guide the selection of process parameters and ensure that printed parts are consistent and reliable.

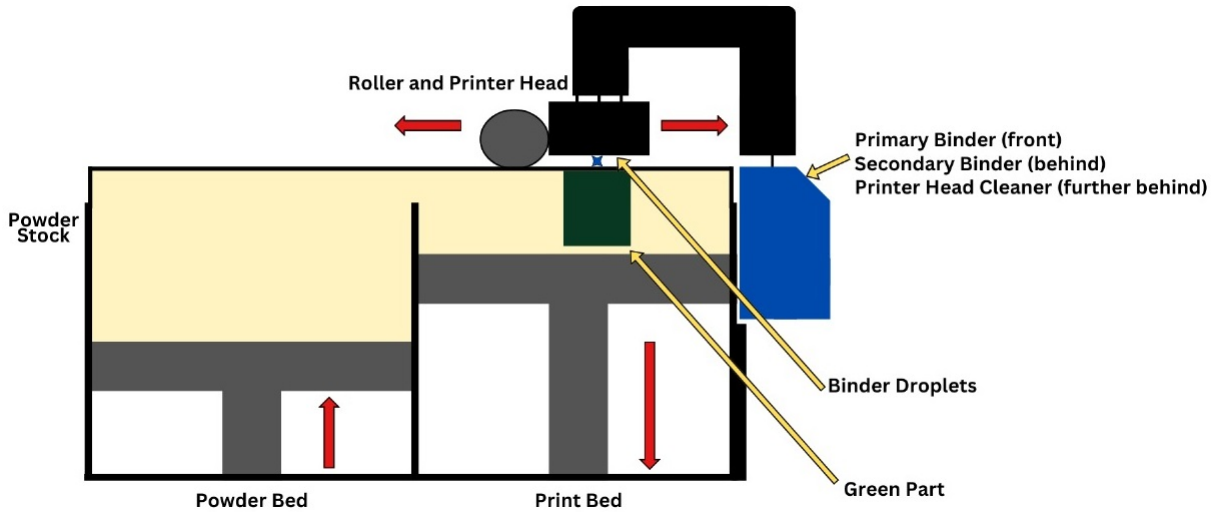


Figure 1: Schematic diagram of the BJAM 3D printer.

Figure 1 illustrates a schematic diagram of the BJAM process. The printing process produces green parts, which are initially very weak and difficult to measure accurately. To enhance their strength, a sintering process is carried out at a high temperature of up to 1250°C . The powder used was TP-80 Ceramic Composite Powders provided by Microjet Technology Co., Ltd. The powder composition was crystal line silica (quartz) 50%, kaolinite 25%, and potassium feldspar 25%. The size range of these powders was $50\text{--}200\text{ }\mu\text{m}$. The binder consisted of 8% diethylene glycol, 2% surfactant, and 90% water. Shrinkage was calculated by comparing between the CAD model volume and the sintered part volume. We printed the shapes mentioned above in two different sizes, big and small sizes. The big size sample means the volume is 5301.44 mm^3 and the small size sample means the volume is 3534.3 mm^3 . After the printing process, the samples are sintered to enhance the samples strength. The sintering cycle consists of: raise the temperature from the room temperature to 500°C in 2.5 hours, raise the temperature to 1150°C in 4.5 hours, raise the temperature to 1250°C in two hours, keep the temperature (dwelling Temperature) at 1250°C for 30 minutes (dwelling time) and then start cooling it down to the room temperature naturally.

2.2 Machine Learning Models

This study develops ML models to predict and optimize critical properties of ceramic BJAM parts, focusing on relative density and shrinkage. The effect of process parameters in BJAM, such as layer thickness, Binder saturation, and time delay, often renders traditional optimization methods inadequate. To address this, we evaluated and will discuss the performance of the following ML models: Bagging, Random Forest, Gradient Boosting, AdaBoost, Decision Tree, Linear Regression, Ridge Regression, and Lasso Regression.

2.2.1 Bagging

Bagging, or Bootstrapping, is an ensemble learning technique designed to reduce variance and improve prediction stability by combining multiple tree models' predictions [20]. This study applied Bagging to predict relative density and shrinkage in BJAM parts. The method involves generating multiple bootstrap samples from the original dataset through random sampling with replacement. Each bootstrap sample is used to train a base model, typically a Decision Tree, which is well-suited for capturing non-linear relationships between process parameters such as layer thickness, binder saturation, and time delay. For regression tasks, the predictions from all trained models are aggregated by averaging, as given by the formula:

$$\hat{y} = \frac{1}{B} \sum_{b=1}^B M_b(x) \quad (1)$$

where, $\hat{y}$ is the final prediction, B represents the total number of bootstrap generated models, and $M_b(x)$ is the individual prediction from the b -th base model trained on its corresponding bootstrap sample for the input x . This aggregation reduces the overall variance in the predictions.

2.2.2 Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees and combines their predictions to enhance accuracy and reduce overfitting [20]. Similar to Bagging, each tree in the forest is trained on a bootstrap sample of the original dataset. However, Random Forest introduces an additional level of variability by selecting a random subset of features at each split within a tree. This feature-level randomness further reduces correlations among trees and increases the overall robustness of the model. For regression tasks, the final prediction is obtained by averaging the output of all trees in the ensemble, mathematically expressed as:

$$\hat{y} = \frac{1}{T} \sum_{t=1}^T h_t(x) \quad (2)$$

where, $\hat{y}$ is the aggregated prediction, T represents the total number of trees in the forest, and $h_t(x)$ is the prediction from the t -th tree for the input x .

2.2.3 Gradient Boosting

Gradient Boosting is a predictive modeling technique that builds a series of decision trees in a sequential manner, where each new tree is trained to correct the errors made by the previous trees [21]. Unlike models that train all components independently, Gradient Boosting emphasizes iterative improvement by refining residuals at each step. The goal is to minimize a predefined loss function $L(y, \hat{y})$, which measures the deviation between the actual values y and the predicted values $\hat{y}$. For regression tasks, the prediction is iteratively updated as follows:

$$\hat{y}^{(m)} = \hat{y}^{(m-1)} + \eta \cdot h_m(x) \quad (3)$$

where $\hat{y}^{(m)}$ is the updated prediction after the m -th iteration, η is the learning rate that controls the contribution of the m -th tree, and $h_m(x)$ is the tree fitted to the residuals.

2.2.4 AdaBoost

AdaBoost (Adaptive Boosting) is a boosting algorithm that builds a strong predictive model by sequentially combining multiple weak learners, typically decision trees [22]. Unlike other methods that assign equal weight to all training samples, AdaBoost dynamically adjusts the weights of each sample to focus on those that were misclassified by previous models [23]. The algorithm starts by assigning equal weights to all samples. At each iteration, a weak learner $h_m(x)$ is trained, and its weighted error ϵ_m is calculated as:

$$\epsilon_m = \frac{\sum_{i=1}^n w_i \cdot \mathbb{I}(y_i \neq h_m(x_i))}{\sum_{i=1}^n w_i} \quad (4)$$

where w_i is the weight of sample i , y_i is the true label, $h_m(x_i)$ is the prediction of the m -th weak learner, and $\mathbb{I}(\cdot)$ is the indicator function. The performance of the weak learner determines its contribution to the final model, quantified by the model weight:

$$\alpha_m = \ln \left(\frac{1 - \epsilon_m}{\epsilon_m} \right) \quad (5)$$

Misclassified samples are assigned higher weights to ensure that subsequent models focus on these challenging examples. The final prediction $H(x)$ is obtained by a weighted majority vote for classification or a weighted average for regression:

$$H(x) = \text{sign} \left(\sum_{m=1}^M \alpha_m \cdot h_m(x) \right) \quad (6)$$

2.2.5 Decision Tree

Decision Tree is a tree-structured nonparametric model for regression and classification tasks [24]. Recursively partitions the dataset into subsets based on feature thresholds to minimize a specific impurity metric or error at each split. The tree grows until a stopping criterion is met, such as a maximum depth, minimum number of samples per leaf, or no further reduction in error. Decision Trees are highly interpretable and capable of capturing non-linear relationships.

2.2.6 Linear Regression

Linear Regression is one of the simplest and most interpretable statistical models used for predicting a continuous outcome. It assumes a linear relationship between the input features and the target variable [25]. The model predicts the target y as a weighted sum of the input features $x_1, x_2, \dots, x_p$ and an intercept term β_0 , represented as:

$$\hat{y} = \beta_0 + \sum_{j=1}^p \beta_j x_j \quad (7)$$

where β_j are the coefficients representing the effect of each feature x_j on the target variable.

2.2.7 Ridge and Lasso Regression

Ridge Regression [26] and Lasso Regression [27] are extensions of Linear Regression that introduce regularization to prevent overfitting and improve model generalization. Both methods modify the objective function of Linear Regression by adding a penalty term.

Ridge Regression Ridge Regression introduces L_2 -regularization to the linear regression model, penalizing the squared magnitude of the coefficients [28]. This helps address multicollinearity and prevents overfitting, making it particularly suitable for scenarios with a high number of correlated features. The Ridge Regression objective is:

$$\text{Loss} = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^p \beta_j^2 \quad (8)$$

where λ is the regularization parameter that controls the trade-off between minimizing RSS and shrinking the coefficients.

Lasso Regression Lasso Regression applies L_1 -regularization, penalizing the absolute magnitude of coefficients. This enables feature selection by shrinking some coefficients to exactly zero, effectively identifying the most influential predictors [27]. The Lasso Regression objective is:

$$\text{Loss} = \sum_{i=1}^n (y_i - \hat{y}_i)^2 + \lambda \sum_{j=1}^p |\beta_j| \quad (9)$$

2.3 Feature Importance in Predictive Model

Understanding feature importance is crucial for interpreting predictive models and identifying key variables driving the target outcomes [29]. In BJAM, process parameters such as layer thickness, core saturation, and time delay significantly affect the target properties of relative density and shrinkage. By analyzing the importance of the features helps optimize these parameters by identifying those with the greatest impact on the model predictions, thereby guiding optimization of the process settings to improve these outcomes. In tree-based models, such as Random Forest and Gradient Boosting, feature importance is determined by evaluating the reduction in impurity (e.g., the Gini index) or gain at each decision split. Likewise, when ensemble models

were applied, feature importance could be ranked based on their contributions to splits across trees [30]. This ranking can provide insights into the most critical predictors, highlighting which features have the greatest impact on the model’s accuracy.

To achieve a more comprehensive and model-agnostic understanding of feature effects, we used SHAP [31] in this study which provides a model-agnostic interpretation of feature contributions in predictive models for relative density and shrinkage in BJAM. SHAP is an interpretability approach grounded in Shapley values from cooperative game theory [32], which provide a fair way to allocate the “payout” (the model’s prediction) among a set of contributors (the input features). In this framework, each feature is conceptually treated as a player in a coalition, and the model’s predicted outcome for a given instance is the result of all players working together. SHAP ensures that the contribution of each feature is assessed in a manner that satisfies desirable properties—namely, local accuracy (the feature contributions sum up to the actual model prediction for the instance), missingness (features that are “absent” or have no influence receive zero contribution), and consistency (if a feature’s influence on the prediction increases in a new model, its attributed importance will not decrease). This makes SHAP a robust explanation method for any machine learning model, as it does not rely on the internal structure of the model and can capture both linear and non-linear effects. SHAP utilizes an additive explanation model, which decomposes the prediction of a machine learning model into contributions from individual features [32]. The additive model is expressed as:

$$g(\mathbf{z}') = \phi_0 + \sum_{i=1}^M \phi_i z'_i \quad (10)$$

where M is the total number of input features, $\mathbf{z}' \in \{0, 1\}^M$ represents the simplified binary representation of input features, and $\phi_i \in \mathbb{R}$ quantifies the contribution of the i -th feature to the prediction. The term ϕ_0 denotes the baseline prediction, which is the model output when no features are present. SHAP ensures local accuracy, missingness, and consistency, making it a robust explanation method.

The SHAP value for a feature i is computed as:

$$\text{expl}_i(f, \mathbf{x}) = \sum_{\mathbf{z}' \subseteq \mathbf{x}'} \frac{|\mathbf{z}'|!(M - |\mathbf{z}'| - 1)!}{M!} [f_{\mathbf{x}}(\mathbf{z}') - f_{\mathbf{x}}(\mathbf{z}' \setminus i)], \quad (11)$$

where $|\mathbf{z}'|$ is the number of active features in the subset $\mathbf{z}'$, and $\mathbf{z}' \subseteq \mathbf{x}'$ represents all subsets of the input features $\mathbf{x}'$. The term $f_{\mathbf{x}}(\mathbf{z}')$ is the model prediction using the subset $\mathbf{z}'$, and $f_{\mathbf{x}}(\mathbf{z}' \setminus i)$ is the prediction when feature i is excluded.

In our study, applying SHAP to the trained predictive models of relative density and shrinkage in BJAM provided a nuanced, model-agnostic explanation of how each process parameter influences the predictions. The SHAP analysis decomposed each model prediction into contributions from individual input factors, offering clear insight into how features such as layer thickness, core saturation, and time delay affected the resulting relative density or shrinkage. For example, a positive SHAP value for layer thickness in a particular prediction indicates that a larger layer thickness (for that sample) pushed the predicted density or shrinkage higher than the baseline, whereas a negative SHAP value would indicate a lowering effect. By examining these feature contributions across all predictions, we could identify which parameters consistently have the greatest impact on the targets. Moreover, SHAP values allowed us to observe interaction and non-linear effects – for instance, how the effect of core saturation on density might differ depending on the level of another feature – which traditional global feature importance measures might miss. This level of interpretability proved instrumental in elucidating the relationships between BJAM process parameters and the part quality metrics, supporting data-driven efforts toward process optimization. In summary, the use of SHAP in this work enhances trust in the model’s predictions and provides actionable insights, as it pinpoints the key process variables that should be tuned to achieve improved relative density and minimized shrinkage in binder jetting production.

2.4 Data Collection Procedure

This is a secondary study of existing research data. The original data was collected by Obeidat et al. [1, 2] to investigate the relevant density and shrinkage of BJAM printed parts. The original tests were with 1 and 2 prints, where the number of prints refers to how many times the print head deposits binder on the same layer before spreading the next powder layer. The time delays used were 0 seconds and 10 seconds, indicating the interval needed to spread the powder on the next layer after binder deposition on the previous layer. Core saturation limits were set at 50%, 75%, and 100%, representing the ratio of binder volume to the volume of open pores in the powder bed. A layer thickness of 0.2mm was maintained. The feed box magnification was 1.2, meaning that if the fabrication box is lowered by 1mm, the feeding box is raised by 1.2mm. We constructed four types of geometries: *spheres*, *cylinders*, *rectangular blocks*, and *cones*. Table 1 outlines the parameters used in the building process. All the post-processing details of this process can be found in [1].

The layer thickness was chosen to be 0.2 mm to balance print resolution and build speed. 0.2 mm is widely used in Binder Jetting literature for producing parts with a good surface finish with lower printing time. Core Saturation 100%, 75%, and 50% were selected to systematically study the effect of binder content on dimensional accuracy and internal porosity. High saturation can

cause bleeding, while low saturation may cause weak bonding, and have a broad range to be tested. Multiple print passes (2 prints) were introduced to evaluate if repeated binder deposition improves layer bonding, densification, and reduces internal defects compared to a single pass. A time delay between layers of 0 sec and 10 sec was implemented to study the impact of drying time on binder migration, dimensional stability, and layer adhesion. The time delay simulates controlled evaporation conditions often used to enhance uniformity.

We printed 3 samples of each shape at each parameter. We measured the density of each sample using an Archimedes method. We measured the dry mass and the wet mass of each sample and used Equation 1 to calculate the apparent density. The ceramics powder used has a density of 2.5 g/ml. We calculated the relative density for each sample by dividing the apparent density by 2.5 g/ml. The shrinkage was calculated as follows: The three dimensions of the sintered part was measured using a caliper. Then the volume of each sintered part was calculated. A comparison between the CAD model volume and the sintered part volume was made. The shrinkage percentage was calculated using the equation: (Sintered part volume- CAD model volume)/ CAD model volume.

$$\rho_a = \rho_{wt} \frac{m_d}{m_d - m_w}$$

- ρ_a : Apparent density of the part.
- ρ_{wt} : Water density at the measurement temperature.
- m_d : Dry mass of the part.
- m_w : Wet mass of the part.

Table 1: The parameters used for printing different geometries.

Batch	Layer thickness (mm)	Binder saturation (%)	Time delay (s)	Number of prints
Batch 1	0.2	100	0	1 and 2
Batch 2	0.2	100	10	1 and 2
Batch 3	0.2	75	0	1 and 2
Batch 4	0.2	75	10	1 and 2
Batch 5	0.2	50	0	1 and 2
Batch 6	0.2	50	10	1 and 2

The parts were printed in batches to maintain consistency. Four trials of each group of parts were printed at a 0.2mm layer thickness. While creating the parts, the overall volume for different geometries was kept consistent. The geometric information for the generated parts is provided in Table 2. At the end of fabrication, the printed parts were left in the printer for approximately 6 hours before moving them to the de-powering station. The parts were then dried and cured at 35°C for 6-8 hours to prevent any cracks during the sintering process. The loose powder was then removed from the surfaces of the parts to prepare them for sintering.

Table 2: Dimensions considered in printing the ceramic part files

Part	Volume (mm ³)	Height (mm)	Diameter (mm)	Width (mm)	Surface Area (mm ²)
Small Samples					
Sphere	3534.3	NA	18.9	NA	1121.63
Cylinder	3534.3	20	15	NA	1295.25
Rectangular Block	3534.3	20	NA	13.29	1416.44
Cone	3534.3	20	25.98	NA	1503.34
Big Samples					
Sphere	5301.44	NA	21.63	NA	1469.82
Cylinder	5301.44	30	15	NA	1767.15
Rectangular Block	5301.44	30	NA	13.29	1948.05
Cone	5301.44	30	25.98	NA	1864.23

3 Results and Discussion

3.1 Predictive Modeling

We used the collected data to train two sets of models for predicting the relative density and shrinkage of AM parts. To ensure robust model evaluation and optimization, we performed k-fold cross-validation ($k = 5$), which provides a more reliable assessment of model performance by reducing the impact of sampling variability. Hyperparameters were optimized through a comprehensive grid search strategy for each model type. As a pre-processing step, all features were normalized using min-max scaling to the range $(0, 1)$ to ensure consistent feature scaling across the dataset. Algorithm 1 details the steps to train the proposed models and select the best one for the purpose of future usage.

Algorithm 1 K-Fold Cross-Validation with Grid Search for Hyperparameter Optimization

Require: Dataset $\mathcal{D}$ of N samples; hyperparameter grid Θ

Ensure: Test MSEs across outer folds

```

1: # Data Preprocessing & Split
2: Partition  $\mathcal{D}$  into  $K$  stratified folds:  $\{F_1, \dots, F_K\}$ 
3: for each  $i \in \{1, \dots, K\}$  do
4:    $T_{\text{outer}} \leftarrow F_i$ 
5:    $D_{\text{train}} \leftarrow \bigcup_{j \neq i} F_j$ 
6:   Normalize each feature of  $T_{\text{outer}}$  and  $D_{\text{train}}$  to  $[0, 1]$  via min-max scaling
7:    $\text{best\_params} \leftarrow \emptyset$ ,  $\text{best\_mse} \leftarrow +\infty$  ▷ For hyperparameter tuning
8:   Partition  $D_{\text{train}}$  into  $K$  stratified folds  $\{I_1, \dots, I_K\}$ 
9:   for each  $\theta \in \Theta$  do
10:     $\text{mse\_sum} \leftarrow 0$ 
11:    for each  $k \in \{1, \dots, K\}$  do
12:       $V_k \leftarrow I_k$ ,  $T_k \leftarrow \bigcup_{\substack{j=1 \\ j \neq k}}^K I_j$ 
13:      Train model  $M_\theta$  on  $T_k$ 
14:       $\text{mse}_k \leftarrow \text{MSE}(M_\theta, V_k)$ 
15:       $\text{mse\_sum} \leftarrow \text{mse\_sum} + \text{mse}_k$ 
16:    end for
17:     $\text{avg\_mse} \leftarrow \text{mse\_sum} / K$ 
18:    if  $\text{avg\_mse} < \text{best\_mse}$  then
19:       $\text{best\_mse} \leftarrow \text{avg\_mse}$ 
20:       $\text{best\_params} \leftarrow \theta$ 
21:    end if
22:  end for
23:  # Retrain and evaluate on outer fold
24:  Train final model  $M_{\text{opt}}$  on  $D_{\text{train}}$  with  $\text{best\_params}$ 
25:   $\text{test\_mse}_i \leftarrow \text{MSE}(M_{\text{opt}}, T_{\text{outer}})$ 
26: end for
27: return  $\{\text{test\_mse}_1, \dots, \text{test\_mse}_5\}$ 

```

This comprehensive training and evaluation algorithm ensures that our model performances are both reliable and reproducible, and that the chosen algorithms are well-matched to the underlying data patterns for each predictive target. Table 3 presents the performance metrics for different ML models in predicting Relative Density and Sample Shrinkage, evaluated using Mean Squared Error (MSE) and its corresponding standard deviation ($\pm$ std).

For predicting *Relative Density*, the Bagging method achieved the lowest MSE (0.0156 ± 0.085), followed closely by Random Forest (0.0163 ± 0.091). The superior performance of ensemble methods suggests that the relationship between input features and relative density involves complex interactions and potentially nonlinear patterns. Ensemble approaches, particularly Bagging, excel at capturing these complex relationships while reducing model variance through aggregated predictions. In contrast, regularization-based models such as Ridge Regression and Lasso Regression demonstrated lower performance, with Lasso yielding an MSE of 0.0311 ± 0.137 . This performance gap indicates that linear models may be insufficient for capturing the intricate relationships present in the relative density prediction task.

Table 3: Model performance in terms of MSE for Relative Density and Shrinkage. The first column shows the trained ML model, and the subsequent columns shows the MSE for predicting Relative Density and Shrinkage

Model	Relative Density MSE ($\pm$ std)	Sample Shrinkage MSE ($\pm$ std)
Bagging	0.0156 ($\pm$ 0.085)	0.0106 ($\pm$ 0.059)
Random Forest	0.0163 ($\pm$ 0.091)	0.0127 ($\pm$ 0.072)
Gradient Boosting	0.0167 ($\pm$ 0.125)	0.0137 ($\pm$ 0.074)
AdaBoost	0.0192 ($\pm$ 0.115)	0.0070 ($\pm$ 0.024)
Decision Tree	0.0194 ($\pm$ 0.111)	0.0097 ($\pm$ 0.072)
Linear Regression	0.0226 ($\pm$ 0.089)	0.0028 ($\pm$ 0.031)
Ridge	0.0232 ($\pm$ 0.093)	0.0030 ($\pm$ 0.026)
Lasso	0.0311 ($\pm$ 0.137)	0.0359 ($\pm$ 0.061)

The prediction of *Sample Shrinkage* revealed a notably different pattern in model performance compared to Relative Density models. Linear Regression was the best-performing model with the lowest MSE (0.0028 ± 0.031), significantly outperforming more complex ensemble methods like Gradient Boosting and Random Forest. This contrast in optimal model types between the two prediction tasks provides valuable insights into the underlying relationships in the data. The success of Linear Regression suggests that sample shrinkage exhibits a predominantly linear relationship with the input features, making complex nonlinear models unnecessary and potentially counterproductive.

Table 4: Optimal hyperparameters determined using grid search for the ML models

Model	Hypr-Parameters
Linear Regression	fit_intercept: True
Random Forest	n_estimators: 100, max_depth: 10, min_samples_split: 2, min_samples_leaf: 4
Decision Tree	max_depth: 30, min_samples_split: 5, min_samples_leaf: 1
Lasso	alpha: 0.1
Gradient Boosting	n_estimators: 100, learning_rate: 0.01, max_depth: 10
Bagging	n_estimators: 100, max_samples: 0.5, max_features: 1.0
Ridge	alpha: 10
AdaBoost	n_estimators: 100

These findings highlight the importance of tailoring model selection to the specific characteristics of each prediction task. While relative density prediction benefits from sophisticated ensemble methods that can capture complex feature interactions, sample shrinkage prediction achieves optimal results with a simpler linear model. This distinction not only impacts prediction accuracy but also has practical implications for designing models that are computationally efficient and interpretable in industrial applications. Table 4 shows the optimal hyper-parameter for the trained models. A detailed analysis of the effect of the features in the best-performing models for both cases is discussed in section 3.2.

3.2 Feature Importance

To gain deeper insights into how different features influence our model predictions, we conducted a feature importance analysis using SHAP values. SHAP values provide a unified measure of feature importance by quantifying each feature’s contribution to individual predictions while accounting for feature interactions. For this analysis, we focused on the best-performing models identified in section 3.1; the Bagging model for Relative Density prediction and the Linear Regression model for Shrinkage prediction. Mean SHAP values are computed by averaging the SHAP values for each feature in the dataset. As Bagging models generate an ensemble of decision trees, calculating the SHAP values for this model requires aggregating the contributions across all trees in the ensemble. The TreeSHAP algorithm is employed to compute SHAP values for each decision tree, and these values are then averaged across all trees in the Bagging ensemble to obtain the final feature importance measures. This approach accounts for both the hierarchical structure of decision trees and the ensemble nature of the Bagging model, providing a robust estimate of feature importance that considers complex interactions and nonlinear relationships in the data.

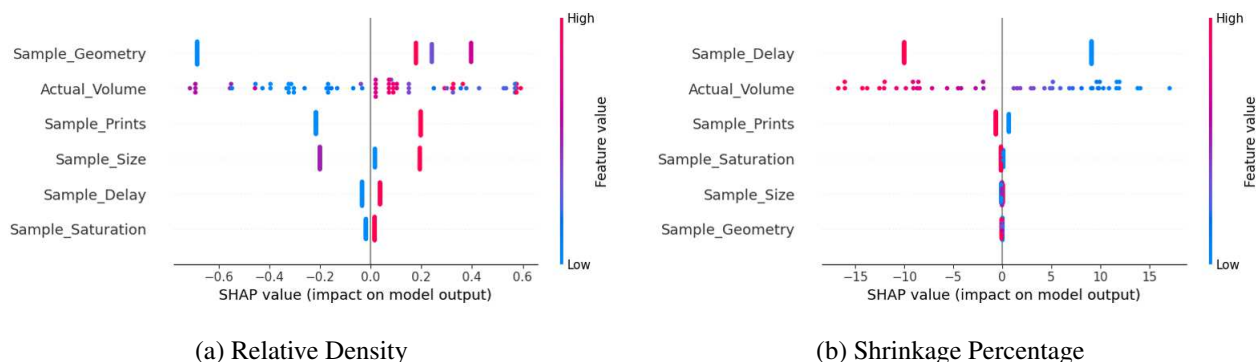


Figure 2: Beeswarm plots showing global SHAP analysis for feature impacts for predicting (a) Relative Density and (b) Shrinkage Percentage.

The bee swarm plots, shown in Figure 2, represent the SHAP values for each feature’s impact on model predictions and are ranked in descending order to determine the global feature relevance. For the Relative Density model, the bee swarm plot highlights that Sample Geometry exhibits a significant range of SHAP values, indicating its strong positive and negative influences on the outcome. Actual Volume also shows a moderate influence, while features like Sample Prints, Sample Delay, and Sample Size contribute less variability. Conversely, Sample Saturation has a negligible impact.

This suggests that geometric factors and material volume are critical determinants of achieving consistent density in the manufacturing process. Our experimental observations support this finding, particularly in the case of conic geometries, which exhibited a tendency to collapse during the cooling phase, primarily at their tips. In contrast, spherical geometries demonstrated more uniform cooling behavior due to their evenly distributed surface area, explaining the significant influence of geometry in our model predictions. The importance of volume in the model aligns with fundamental physical principles, as the initial geometric volume has a direct relationship with relative density. This is due to the high surface area of the big samples compared to the small samples as shown in Table 2. Having a high surface area increases the heat transfer rate during the sintering process, improving the sintered parts’ density. The number of sample prints in turn increases the amount of the deposited binder to double in our case which increases also relative density due to the increase in the binding force between the layers. This relationship enables us to quantify density changes during both the sintering and cooling processes, providing a theoretical foundation for the model’s emphasis on volume as a key predictive feature. For the Shrinkage Percentage model, the bee swarm plot reveals that Sample Delay has the most substantial effect, with a wide spread of SHAP values, indicating that delays significantly influence shrinkage outcomes. Sample delay, in this case, is the time delay between printing two consecutive layers. Time delay gives more time for the deposited binder to penetrate through the printed layer and to create a good binding force between the layers. Sample volumes have the same effect as the prior model, while features such as Sample Geometry and Sample Size show minimal impact on model predictive capabilities. This highlights the need for precise control of processing delays to mitigate material shrinkage.

The bar plots, presented in Figure 3, provide an aggregated view of feature importance by showing the mean absolute SHAP values. The results derived from the bar plot align with the observations in the bee swarm plot. For the Relative Density model, Sample Geometry is the most influential feature, followed by Actual Volume and Sample Prints, while features like Sample Delay and Sample Saturation have smaller impacts. This enforces the previous conclusion that geometric and volumetric considerations should be prioritized in processes requiring high material density, as they likely affect material packing and structural integrity. For the Shrinkage Percentage model, Sample Delay stands out as the dominant feature, with Actual Volume contributing moderately.

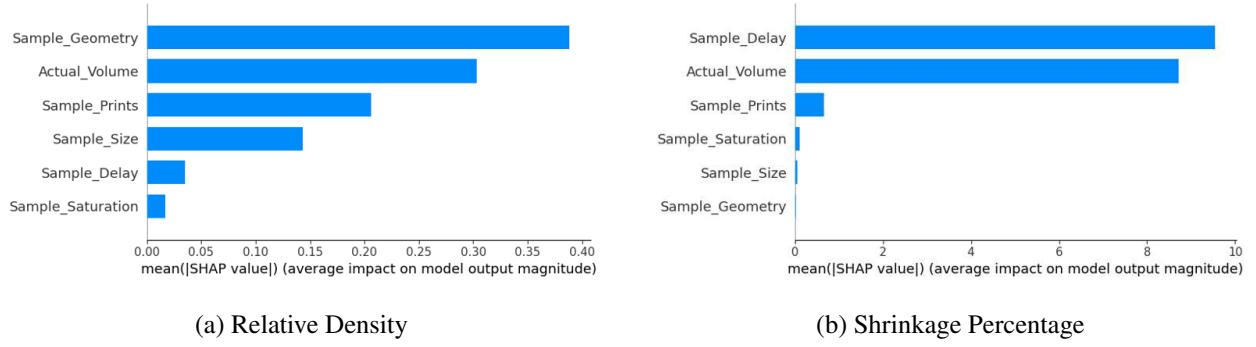


Figure 3: A bar chart for the importance of different features on (a) Relative Density and (b) Shrinkage Percentage.

The remaining features have negligible influence. This identifies the importance of managing printing delays and deciding on appropriate material volumes to control shrinkage and ensure dimensional accuracy in BJAM.

3.3 Physical Validation

To physically validate the developed framework, we conducted a small controlled experiment using the Bagging model. To assess its predictive accuracy under real printing conditions, a targeted validation study was performed. Specific combinations of printing parameters (binder saturation, layer thickness, time delay, and number of prints) were selected along with a desired part geometry and nominal volume. The trained Bagging model was then used to predict the expected shrinkage and final relative density for each parameter set. These predictions were subsequently used to adjust the geometry volume prior to printing.

After fabrication and sintering, the actual volume and density of each specimen were measured using calipers and Archimedes method, respectively. Table 5 summarizes the predicted and measured values, while Figure 4 presents photographic examples of the sintered specimens (sphere and rectangular geometries). New print heads were used to emulate out-of-distribution conditions relative to the original training data. These validation samples were completely independent from the dataset used for model training.



Figure 4: Out of distribution printed parts for testing the developed model.

The generalization capability of the trained models was assessed using the generated out-of-distribution (OOD) dataset. The evaluation results indicate that model performance varies depending on the target variable. For predicting sample shrinkage, linear models such as Linear Regression and Ridge Regression demonstrated the strongest OOD performance, achieving the lowest mean squared error (≈ 0.061). In contrast, ensemble-based models produced substantially larger prediction errors under distribution shift, suggesting reduced generalization for this target. For relative density prediction, however, ensemble models performed better, with the Bagging model achieving the lowest MSE (≈ 0.0156) among the tested models. These results indicate that shrinkage behavior exhibits a largely linear dependency on the selected process parameters, while relative density is influenced by more complex nonlinear interactions that are better captured by ensemble learning methods.

Table 5: Out-of-Distribution (OOD) performance comparison using Mean Squared Error (MSE). Lower values indicate better predictive accuracy.

Model	MSE (Sample Shrinkage)	MSE (Relative Density)
Linear Regression	<u>0.0612</u>	0.0525
Random Forest	0.3345	0.0165
Decision Tree	0.2856	0.0189
Ridge	0.0608	0.0454
Lasso	0.2500	0.0234
Gradient Boosting	0.2515	<u>0.0157</u>
AdaBoost	0.2434	0.0193
Bagging	0.2960	0.0156

3.4 Practical Implementation and Future Direction

In this study we developed ML models to predict the targeted quality metrics (relative density and shrinkage) for parts printed using BJAM method for ceramic composites. Utilizing these developed model, specially (Bagging and Linear Regression) models. The trained models can be used to (i) adjust the printing parameters (e.g., time delay, layer thickness, etc.) and/or model geometries if needed to improve the part quality; (ii) identify the critical features for future part production where the printing protocol can focus on the identified factors. For example, SHAP value analysis of the model shows that the sample delay is crucial for controlling part shrinkage. So the printing operations can be adjusted to achieve the desired volume. This reduces trial-and-error experiments, reduces material consumption, and lowers production costs; and finally, (iii) By integrating the developed ML models into an inverse design pipeline (not part of the current study), manufacturers can specify desired part attributes (like target density or minimal shrinkage) and obtain optimized printing settings, which directly translates to enhanced process reliability and product consistency.

The insights obtained from both the predictive models and the SHAP analysis provide a foundation for developing future inverse design strategies for BJAM printed components. For achieving specific relative density targets, the SHAP analysis revealed that Sample Geometry and Actual Volume have the strongest influence on the final output. Therefore, these parameters should be the primary variables adjusted within an optimization framework to achieve the desired relative density. A constrained optimization formulation can be developed in which Sample Geometry and Actual Volume are treated as controllable design variables. Similarly, for controlling shrinkage behavior, the features Sample Delay and Actual Volume can be incorporated as key decision variables within the optimization model. For designs that must simultaneously satisfy both Relative Density and Shrinkage Percentage requirements, a multi-objective constrained optimization framework can be formulated that integrates the respective predictive models for each target response.

4 Conclusion

This study demonstrates the potential of ML in predicting and optimizing key quality metrics in BJAM of ceramic composite powders. By employing a range of ML models, we effectively captured the relationships between process parameters and the resulting properties of relative density and shrinkage. Ensemble methods, particularly Bagging, outperformed other models in predicting relative density, suggesting that density exhibits a non-linear relationship with the input features. Conversely, Linear Regression was the most effective model for predicting shrinkage, suggesting that shrinkage exhibits a more linear relationship with the input features. Feature importance analysis using SHAP revealed that Sample Geometry and Actual Volume were the most influential factors for predicting relative density, reflecting the critical role of geometric and volumetric considerations in material packing and structural consistency while for shrinkage, Sample Delay was identified as the dominant factor, indicating the importance of precise timing in layer deposition may minimize dimensional inaccuracies. These insights can be particularly valuable for developing inverse design approaches, where specific part characteristics can be targeted through careful control of the most influential parameters. Furthermore, clearly identifying important parameters can help streamline the optimization process, reducing the need for extensive experimental trials while printing new products using BJAM technology and ceramic powders. We plan to use this as a pre-processing stage during CAD model generation to suggest users what type of shrinkage to expect from the designed model. Future work could focus on expanding this approach to include additional process parameters and quality metrics, as well as developing automated optimization algorithms that leverage these insights. The methodologies developed in this study could also be adapted to other additive manufacturing processes, potentially accelerating the development and optimization of new materials and applications in the broader field of advanced manufacturing.

Statements and Declarations

All authors certify that they have no affiliations with or involvement in any organization or entity with any financial or non-financial interest in the subject matter or materials discussed in this manuscript.

Acknowledgments

The Sam Houston State University Office of Research & Sponsored Programs Internal Grant funded the initial data collection efforts and their *Research & Development Program* (Project Id: 25-0131-P0001) supported the experimentation.

Data Statement

The data will be made available upon request to the authors.

Author contributions

Authors SO, RH, RRA, and SHAF, contributed to the manuscript's writing, conception, and design of the study. SO & RRA performed the BJAM printing and data collection steps. SHAF and RH wrote the ML codes, analyzed the results, and generated all the tables and visualizations. All authors read and approved the final manuscript.

References

- [1] S. Obeidat, J. Ma, S. Himelstein, and A. Acharya, "The impact of the printed part geometry on the shrinkage and relative density in binder jetting additive manufacturing of ceramics powder," in *ASME International Mechanical Engineering Congress and Exposition*, vol. 86632. American Society of Mechanical Engineers, 2022, p. V02AT02A026.
- [2] S. Obeidat, I. Basith, and U. Dakeev, "Impact of Number of Prints and Time Delay on Density and Shrinkage Percentage of Parts Printed by Binder Jetting 3D Printers," *International Journal of Engineering Research & Innovation*, vol. Spring, no. 2024, Aug. 2024. [Online]. Available: <https://zenodo.org/records/13256816>
- [3] Z. Hu and W. Yan, "Data-driven modeling of process-structure-property relationships in metal additive manufacturing," *npj Advanced Manufacturing*, vol. 1, no. 1, p. 3, 2024.
- [4] A. I. Saimon, E. Yangue, X. Yue, Z. Kong, and C. Liu, "Advancing additive manufacturing through deep learning: A comprehensive review of current progress and future challenges," *IISE Transactions*, pp. 1–24, 2025.
- [5] M. Safdar, P. P. Paul, G. Lamouche, G. Wood, M. Zimmermann, F. Hannesen, C. Bescond, P. Wanjara, and Y. F. Zhao, "Fundamental requirements of a machine learning operations platform for industrial metal additive manufacturing," *Computers in Industry*, vol. 154, p. 104037, 2024.
- [6] A. J. Thomas, E. Barocio, and R. B. Pipes, "A machine learning approach to determine the elastic properties of printed fiber-reinforced polymers," *Composites Science and Technology*, vol. 220, p. 109293, Mar. 2022. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0266353822000355>
- [7] N. Satterlee, E. Torresani, E. Olevsky, and J. S. Kang, "Automatic detection and characterization of porosities in cross-section images of metal parts produced by binder jetting using machine learning and image augmentation," *Journal of Intelligent Manufacturing*, vol. 35, no. 3, pp. 1281–1303, Mar. 2024. [Online]. Available: <https://doi.org/10.1007/s10845-023-02100-9>
- [8] D. Günther, M. F. Pirehgalin, I. Weiß, and B. Vogel-Heuser, "Condition monitoring for the Binder Jetting AM-process with machine learning approaches," in *2020 IEEE Conference on Industrial Cyberphysical Systems (ICPS)*, vol. 1, Jun. 2020, pp. 417–420. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/9274716>
- [9] Y. Zhu, Z. Wu, W. D. Hartley, J. M. Sietins, C. B. Williams, and H. Z. Yu, "Unraveling pore evolution in post-processing of binder jetting materials: X-ray computed tomography, computer vision, and machine learning," *Additive Manufacturing*, vol. 34, p. 101183, Aug. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214860420305558>
- [10] A. Shaik, S. K. Kenchugonde, S. Kuruva, D. Sabbu, A. K. R. Y, and V. K. CH R, "Prediction of weld bead cross-sectional area in wire arc additive manufacturing using vision system integrated with machine learning approach," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, Jul. 2024. [Online]. Available: <https://doi.org/10.1007/s12008-024-01952-6>

- [11] M. A. Rahman, S. Jamal, M. V. Cruz, B. Silwal, and H. Taheri, "In situ process monitoring of multi-layer deposition in wire arc additive manufacturing (WAAM) process with acoustic data analysis and machine learning," *The International Journal of Advanced Manufacturing Technology*, vol. 132, no. 9, pp. 5087–5101, Jun. 2024. [Online]. Available: <https://doi.org/10.1007/s00170-024-13641-5>
- [12] L.-K. Chang, R.-S. Chen, M.-C. Tsai, R.-M. Lee, C.-C. Lin, J.-C. Huang, T.-W. Chang, and M.-H. Horng, "Machine learning applied to property prediction of metal additive manufacturing products with textural features extraction," *The International Journal of Advanced Manufacturing Technology*, vol. 132, no. 1, pp. 83–98, May 2024. [Online]. Available: <https://doi.org/10.1007/s00170-024-13165-y>
- [13] H. Wang, B. Li, S. Zhang, and F. Xuan, "Traditional machine learning and deep learning for predicting melt-pool cross-sectional morphology of laser powder bed fusion additive manufacturing with thermographic monitoring," *Journal of Intelligent Manufacturing*, Mar. 2024. [Online]. Available: <https://doi.org/10.1007/s10845-024-02356-9>
- [14] P. Akbari, M. Zamani, and A. Mostafaei, "Machine learning prediction of mechanical properties in metal additive manufacturing," *Additive Manufacturing*, vol. 91, p. 104320, Jul. 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S221486042400366X>
- [15] Q. Liu, W. Chen, V. Yakubov, J. J. Kruzic, C. H. Wang, and X. Li, "Interpretable machine learning approach for exploring process-structure-property relationships in metal additive manufacturing," *Additive Manufacturing*, vol. 85, p. 104187, Apr. 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2214860424002331>
- [16] B. K. Chigilipalli and A. Veeramani, "A machine learning approach for the prediction of tensile deformation behavior in wire arc additive manufacturing," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, Nov. 2023. [Online]. Available: <https://doi.org/10.1007/s12008-023-01617-w>
- [17] W. Tian, Q. Li, Q. Wang, D. Chen, and W. Gao, "Additive manufacturing error quantification on stability of composite sandwich plates with lattice-cores through machine learning technique," *Composite Structures*, vol. 327, p. 117645, Jan. 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0263822323009911>
- [18] S. Nohut and M. Schwentenwein, "Machine learning assisted material development for lithography-based additive manufacturing of porous alumina ceramics," *Open Ceramics*, vol. 18, p. 100573, Jun. 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2666539524000373>
- [19] X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi, "A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability," *Computer Science Review*, vol. 37, p. 100270, Aug. 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1574013719302527>
- [20] G. James, D. Witten, T. Hastie, R. Tibshirani *et al.*, *An introduction to statistical learning*. Springer, 2013, vol. 112.
- [21] Y. Freund, R. Schapire, and N. Abe, "A short introduction to boosting," *Journal-Japanese Society For Artificial Intelligence*, vol. 14, no. 771-780, p. 1612, 1999.
- [22] Y. Freund, R. E. Schapire *et al.*, "Experiments with a new boosting algorithm," in *icml*, vol. 96. Citeseer, 1996, pp. 148–156.
- [23] Y. Gao, G. Ji, Z. Yang, and J. Pan, "A dynamic adaboost algorithm with adaptive changes of loss function," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1828–1841, 2012. [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/6392455>
- [24] V. G. Costa and C. E. Pedreira, "Recent advances in decision trees: An updated survey," *Artificial Intelligence Review*, vol. 56, no. 5, pp. 4765–4800, 2023. [Online]. Available: <https://link.springer.com/article/10.1007/s10462-022-10275-5>
- [25] G. James, D. Witten, T. Hastie, R. Tibshirani, and J. Taylor, "Linear regression," in *An introduction to statistical learning: With applications in python*. Springer, 2023, pp. 69–134. [Online]. Available: https://link.springer.com/chapter/10.1007/978-1-0716-1418-1_3
- [26] G. C. McDonald, "Ridge regression," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 1, no. 1, pp. 93–100, 2009. [Online]. Available: <https://wires.onlinelibrary.wiley.com/doi/full/10.1002/wics.14>
- [27] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society Series B: Statistical Methodology*, vol. 58, no. 1, pp. 267–288, 1996. [Online]. Available: <https://academic.oup.com/jrsssb/article/58/1/267/7027929>

- [28] R. de Vlaming and P. J. Groenen, "The current and future use of ridge regression for prediction in quantitative genetics," *BioMed research international*, vol. 2015, no. 1, p. 143712, 2015. [Online]. Available: <https://onlinelibrary.wiley.com/doi/full/10.1155/2015/143712>
- [29] J. Chen, L. Q. R. Ooi, T. W. K. Tan, S. Zhang, J. Li, C. L. Asplund, S. B. Eickhoff, D. Bzdok, A. J. Holmes, and B. T. Yeo, "Relationship between prediction accuracy and feature importance reliability: An empirical and theoretical study," *NeuroImage*, vol. 274, p. 120115, 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1053811923002616>
- [30] C.-H. Chen, K. Tanaka, M. Kotera, and K. Funatsu, "Comparison and improvement of the predictability and interpretability with ensemble learning models in qspr applications," *Journal of cheminformatics*, vol. 12, pp. 1–16, 2020. [Online]. Available: <https://link.springer.com/article/10.1186/s13321-020-0417-9>
- [31] S. Lundberg, "A unified approach to interpreting model predictions," *arXiv preprint arXiv:1705.07874*, 2017.
- [32] X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi, "A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability," *Computer Science Review*, vol. 37, p. 100270, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1574013719302527#b146>