

PREDICTION OF MATERIAL REMOVAL RATE FOR CHEMICAL MECHANICAL PLANARIZATION USING BAYESIAN ADDITIVE REGRESSION TREES

Han Bao and Xiyuan Qian*

School of Mathematics
East China University of Science and Technology
Shanghai, China

*Corresponding author's e-mail: xyqian@ecust.edu.cn

Chemical Mechanical Planarization (CMP) is a critical surface processing technique in semiconductor manufacturing, where the Material Removal Rate (MRR) plays a vital role in determining process quality. Traditional physics-driven MRR models based on Preston's Equation have difficulty capturing the complex nonlinear relationships and high-order interactions present in CMP data. This paper applies Bayesian Additive Regression Trees (BART) to CMP MRR prediction. BART uses a Bayesian sum-of-trees framework to model nonlinear effects and interactions while providing posterior uncertainty estimates and variable-importance information. Quantitative comparisons on the validation set show that BART achieves the best predictive performance, outperforming the strongest baseline model by 11% in terms of RMSE. Model reliability and interpretation are examined through residual diagnostics, MCMC convergence checks, hyperparameter-sensitivity analysis, omnibus covariate testing, and variable-importance analysis. These results indicate that BART is effective for CMP MRR prediction and can help identify process variables that warrant monitoring and diagnostic attention.

Keywords: Bayesian Additive Regression Trees (BART), Chemical Mechanical Planarization (CMP), Material Removal Rate (MRR)

(Received on June 11, 2025; Accepted on August 31, 2026)

1. INTRODUCTION

Chemical Mechanical Planarization (CMP) is now widely employed in the semiconductor industry. This process combines mechanical polishing with chemical reactions to achieve enhanced polishing efficiency for various wafer surfaces. As a surface smoothing technique, CMP induces a chemical reaction between the wafer and the abrasive solution through the rotation of the wafer carrier and relative motion with the polishing pad, thereby achieving controlled material removal. The CMP system primarily comprises a platen for mounting the polishing pads, a wafer carrier, an abrasive fluid dispensing unit, and a pad conditioner. During operation, the wafers are pressed against the rotating pads by the carriers, while the abrasive fluid flows at a specified rate onto the pads. This mechanical action facilitates the chemical reaction between the grinding liquid and the wafer surface, removing excess material and flattening the wafer surface, thus reducing roughness. On the opposite side of the platen, a trimmer prevents clogging of the abrasive fluid, ensuring smooth process execution. (See Figure 1 for the diagram of the CMP process)

In the process of CMP, the material removal rate (MRR) is a critical parameter that significantly influences the final processing outcome. During the milling stage, wafers are clamped between polishing pads and carriers, making it challenging to predict the extent of surface smoothing until the process concludes. The planarization process can be adversely affected by factors such as contaminated chemicals, non-uniform slurry particles, and delamination at the chip cross-section. To mitigate these issues, accurately predicting the MRR in CMP is essential, as it indirectly determines the smoothness of the wafers post-processing.

Based on existing literature, various approaches have been devised to predict the MRR during the CMP process. These approaches can be categorized into two main types: model-based and data-driven methods. Most model-based methods rely on the Preston Equation (Xie *et al.*, 2006) to describe the MRR. This Equation expresses the MRR as a function of three factors: the Preston coefficient, the polishing pressure applied, and the relative velocities between the polishing pad and the wafer. The Preston coefficient is generally influenced by the characteristics of the abrasive fluid and the hardness of the wafer material. However, over extended periods of operation, the surface of the polishing pad may deteriorate due to wear and deformation. This deterioration can result in micropores being filled with abrasive fluid and the formation of a glassy thin layer on the pad's surface. Such changes diminish the pad's capacity to hold and transport abrasive materials, leading to a reduction in the MRR and impacting the accuracy of predicted outcomes.

Over the past few years, data-driven methods have been increasingly used to predict MRR during the CMP process. Existing studies include Hypergraph Convolutional Network (Xia *et al.*, 2021), Conditional Probabilistic Autoencoder (Wei and Wu, 2024), Artificial Neural Networks (Nagaraj *et al.*, 2021), and stacking (Li *et al.*, 2019). Standard ensemble

methods such as Random Forest, Gradient Boosting Machine (GBM) (Friedman, 2001), and eXtreme Gradient Boosting (XGBoost) (Chen and Guestrin, 2016) are strong regression baselines because they can capture nonlinear relationships and interactions through multiple decision trees. However, their predictions are usually reported as point estimates, and additional procedures are required to obtain posterior uncertainty information or to interpret process-variable effects in a probabilistic framework. BART is therefore considered in this study because it combines flexible nonlinear regression with Bayesian uncertainty quantification and variable-importance analysis. This capability is useful for CMP applications in which prediction accuracy, uncertainty awareness, and interpretation of influential process variables are all important.

In this paper, a data-driven method based on BART is proposed for MRR prediction during the CMP process. The study addresses the gap between physics-driven models, which may be difficult to adapt to complex nonlinear process data, and standard ensemble-learning methods, which provide strong prediction but limited built-in uncertainty characterization. Section 2 reviews related methods for MRR prediction during CMP. Section 3 presents the principles of BART and the details of MRR prediction. Section 4 conducts a case study using the 2016 PHM Challenge CMP dataset, compares BART with representative baseline methods, and summarizes the experimental findings. Section 5 discusses the limitations of the results. Section 6 concludes and discusses this study.

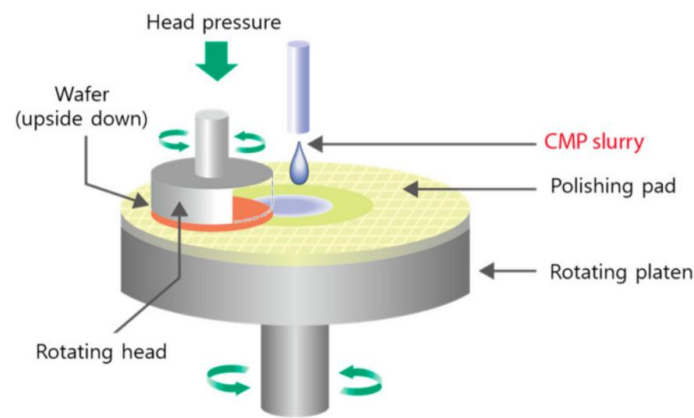


Figure 1. Schematic diagram of CMP

2. RELATED WORK

Most traditional studies on predicting MRR rely on numerical or analytical models to forecast system or process behavior. Evans *et al.* (2003) examined the chemo-mechanical flattening factors influencing MRR, reviewed the fundamental mechanisms of material removal during grinding and polishing, and highlighted critical areas for future research. Lee and Jeong (2011) utilized a simulation approach based on the modified Arrhenius Equation and pad temperature to derive an average dynamic etch rate for the CMP process. They subsequently integrated this rate with the modified Preston Equation to develop an average MRR model. Lim and Dutta (2020) outlined common chemical metrics employed for monitoring and controlling CMP polishing fluids and post-polish rinse chemicals, illustrating how variations in these metrics can impact polishing performance. Kart-Leong *et al.* (2020) introduced a categorical regression technique that, unlike other studies which either neglect wear or assume known wear characteristics for unknown wafers, incorporates prior knowledge of wear during training to enhance MRR predictions.

With the advent of multimodal CMP data, time series data collected during each wafer's CMP process can serve as a foundation for data-driven methods to predict MRR under specified conditions. Wang *et al.* (2017) utilized a deep belief network to analyze the correlation between MRR and parameters such as pressure and rotational speed, and employed a particle swarm optimization algorithm to evaluate how network architecture and learning rate influence prediction accuracy. Zhang *et al.* (2021) adopted a deep learning-based data-driven approach to predict MRR, identified influential process variables using Random Forest, and subsequently proposed an MRR prediction method combining residual convolutional neural networks (ResCNN) with traditional convolutional neural networks, comparing their respective performances.

3. PREDICTION OF MRR WITH BART

3.1. Principle of Bayesian Additive Regression Trees (BART)

Bayesian Additive Regression Trees (BART) were first introduced by Chipman *et al.* (2007). Zhang *et al.* (2007) extended this model by incorporating spatial components, allowing for simultaneous imputation of missing variables and inference of relationships between variables through posterior inference.

The core concept of BART is to transform the information embedded in the data into a "sum-of-trees" model. This approach begins by imposing a prior on the model parameters to regularize the fit and ensure that each regression tree contributes only a small fraction to the overall function $f(x)$. Subsequently, after obtaining observations, the posterior distribution is updated from the prior knowledge. Finally, the posterior distribution is sampled using a Markov Chain Monte Carlo (MCMC) algorithm to estimate the parameters of the tree structure and leaf nodes.

BART constitutes an ensemble learning method for nonparametric function estimation through regression trees. In BART, the prediction space is recursively partitioned into hyperrectangles via binary splits to approximate the unknown function f . The method employs a Bayesian probabilistic framework to estimate $f(x) = E(Y|x)$ as a sum of weak learners represented by decision trees. Specifically, the BART model can be expressed as follows.

$$Y=f(X)+\varepsilon \approx T_1^M(X)+T_2^M(X)+\cdots+T_m^M(X)+\varepsilon, \varepsilon \sim N_n(0, \sigma^2 I_n), \quad (1)$$

where Y is the $n \times 1$ response vector, X is the $n \times p$ design matrix (where p represents the dimensionality of the prediction space, i.e., the number of variables), and ε is the $n \times 1$ noise vector following an n -dimensional normal distribution.

This model incorporates m distinct regression trees, where T denotes each regression tree, comprising a tree structure and parameters M for the terminal nodes (also known as leaf nodes). The notation T^M is used in this paper to represent a tree that fully encompasses both its structure and the set of leaf node parameters. Here, $M = \{u_1, \dots, u_k\}$ represents the parameter values associated with the k leaf nodes in the decision tree. The structure of the tree T signifies that at each internal non-leaf node, there is a splitting rule which selects a variable x_i based on the condition $\{x_i \leq t\}$ or $\{x_i > t\}$, determining the path to a specific leaf node. The prediction is then made using the parameter values of the reached leaf node, denoted as $T^M(X)$. BART algorithm is initialized with m single-node trees, setting the initial value as the average of the predicted values. It iteratively refines the structure of these m trees by incorporating new parameter values obtained from posterior sampling. Each iteration yields the sum of the m returned values from each decision tree, and the final estimate is derived by averaging all results after the burn-in period.

3.1.1. Prior distributions

BART first specifies a prior on the tree structure, terminal node parameters, and variance, assuming independence of σ^2 and $(T_i^M, i = 1, 2, \dots, m)$ and between all m tree structures and terminal node parameters. The joint prior distribution is:

$$P(T_1^M, \dots, T_m^M, \sigma^2) = [\prod_{t,k} P(u_{t,k}|T_t)P(T_t)]P(\sigma^2). \quad (2)$$

The prior of the tree structure can be specified by three aspects: (1) the probability that a node at depth d is an internal node, which is $\alpha(1+d)^{-\beta}$ where $\alpha \in (0,1), \beta \in [0, \infty)$. Here, α controls how likely a terminal node in the tree would split and β controls the number of terminal nodes. (2) the distribution used to choose which covariate is selected for the decision rule in an internal node. (3) the distribution for the value of the selected covariate for the decision rule in an internal node. The default distribution used in (2) and (3) is the discrete uniform distribution for the available covariates.

The prior distribution of leaf node parameters $u_{t,k}$ and the error variance σ^2 can be specified as:

$$u_{t,k} \sim N(0, \frac{1}{4k^2m}), \quad (3)$$

$$\sigma^2 \sim IG(v/2, v\lambda/2). \quad (4)$$

where $\alpha, \beta, k, m, v, \lambda$ are hyperparameters. These hyperparameters can be constructed as a mix of a priori fixed and data-driven values according to Chipman and McCulloch (2024).

3.1.2. Posterior distribution calculation

Along with the set of priors, BART specifies the likelihood of responses in the terminal nodes. A Metropolis-within-Gibbs sampler (Geman S and Geman D, 1984) is employed to generate draws from the posterior distribution $P(T_1^M, \dots, T_m^M, \sigma^2|y)$. The key feature of this sampler for BART is to employ a form of "Bayesian backfitting" (Hastie and Tibshirani, 2000) where the j^{th} tree is fit iteratively, holding all other $m-1$ trees constant by exposing only the residual response that remains unfitted:

$$R_{-j} \equiv y - \sum_{t \neq j} T_t^M(X). \quad (5)$$

The sampling procedure consists of $2m+1$ steps, each of which is detailed as follows:

- (1) sampling from $P(T_1|R_{-1}, \sigma^2)$
- (2) sampling from $P(M_1|T_1, R_{-1}, \sigma^2)$
- (3) sampling from $P(T_2|R_{-2}, \sigma^2)$
- (4) sampling from $P(M_2|T_2, R_{-2}, \sigma^2)$
- $\vdots$
- $(2m-1)$ sampling from $P(T_m|R_{-m}, \sigma^2)$
- $(2m)$ sampling from $P(M_m|T_m, R_{-m}, \sigma^2)$
- $(2m+1)$ sampling from $P(\sigma^2|T_1, M_1, \dots, T_m, M_m, \epsilon)$

Steps (1), (3), ..., $(2m-1)$ involve posterior sampling of the structural parameters of each tree. At each step, one of three possible tree alterations (GROW, PRUNE and CHANGE) is selected based on their respective probabilities of modifying the tree structure. The chosen method dictates the direction of state transition for the newly proposed tree. Subsequently, it relies on the Metropolis-Hastings algorithm. The state transition probability r for GROW, PRUNE and CHANGE is the product of the proposal ratio, likelihood ratio, and tree structure ratio, as shown in Equation (6). Here, T_* denotes the newly proposed tree structure.

$$r = \frac{P(T_* \rightarrow T)}{P(T \rightarrow T_*)} \times \frac{P(R|T_*, \sigma^2)}{P(R|T, \sigma^2)} \times \frac{P(T_*)}{P(T)}. \quad (6)$$

Steps (2), (4), ..., $(2m)$ involve sampling the posterior distribution of the leaf node parameters for each tree, given the tree structure that has been modified in the preceding step. Since both the prior and likelihood are normally distributed, the posterior distribution of the leaf node parameters is conjugate normal, with its mean being a weighted combination of the likelihood and prior parameters (Gelman *et al.*, 2013).

Step $(2m+1)$ involves sampling the posterior distribution of the error variance σ^2 . Due to the Normal-inverse-gamma conjugacy, the posterior of σ^2 is inverse gamma as well.

All $(2m+1)$ steps represent a single Gibbs iteration. Generally, certain iterations are used as “burn-in”, and the number that can be modified is set to be 250 for convenience. An additional 1,000 iterations are taken as samples from the posterior distributions, which can be used to estimate the predicted value. In summary, the detailed sampling process of the posteriors in Bayesian additive regression trees is outlined in Table 1.

Table 1. Posterior sampling algorithm in BART

Algorithm 1: BART Posterior Sampling Algorithm
Initialization: $T_1^M(X) = T_2^M(X) = \dots = T_m^M(X) = \frac{\sum_{i=1}^n y_i}{mn}, y'^{(0)} = T_1^M(X) + T_2^M(X) + \dots + T_m^M(X)$
For $k = 1, 2, \dots, 1250$
For $t = 1, 2, \dots, m$
Calculate partial residuals $R_{-t}^{(k)} = \frac{\sum_{i=1}^n y_i}{n} - \sum_{i=1}^{t-1} T_{i^{(k)}}^M(X) - \sum_{i=t+1}^m T_{i^{(k-1)}}^M(X)$
Sample new structure T_t^* from $P(T_t R_{-t}^{(k)}, \sigma^2)$ and determine the size of μ drawn from the standard normal distribution and r ; if $\mu < r$ then $T_t \leftarrow T_t^*$
Sample from a normal distribution $P(M_t T_t^{(k)}, R_{-t}^{(k)}, \sigma^2)$ to yield $M_t^{(k)}$
End of inner loop
Calculate $y'^{(k)} = T_1^{M^{(k)}}(X) + T_2^{M^{(k)}}(X) + \dots + T_m^{M^{(k)}}(X)$
Calculate $\epsilon^{(k)} = \frac{\sum_{i=1}^n y_i}{n} - y'^{(k)}$
Sampling σ^2 from the inverse gamma distribution $P(\sigma^2 T_1^{(k)}, M_1^{(k)}, \dots, T_m^{(k)}, M_m^{(k)}, \epsilon^{(k)})$
end of outer loop
The predict value can be estimated by averaging to have $\hat{y} = \frac{\sum_{k=251}^{1250} y'^{(k)}}{1000}$

There are some implementations of BART in the R program environment. Chipman and McCulloch (2024) developed the BayesTree package. Kapelner and Bleich (2013) further developed bartMachine, an enhanced R package for BART that includes external prediction functions, mechanisms for handling missing data and built-in cross-validation (BART-CV). BART-CV employs a grid search approach to systematically explore all potential hyperparameter combinations, training models under each combination and selecting the optimal set based on error metrics, thereby further optimizing the BART model.

3.2. BART for MRR Prediction during CMP

In this paper, the MRR prediction task is solved with BART for the first time. The raw CMP process variables are time-series, and they are converted into time-independent features by statistic extraction. There are high-dimensional features, and not all of them contribute to the prediction of MRR. Therefore, feature selection is introduced to select the core feature subset for MRR prediction. Then the selected features are fed into BART for the prediction of MRR during the CMP process.

3.3. Evaluation Indicators

Mean absolute deviation (MAD), mean square error (MSE), root mean square error (RMSE) and coefficient of determination (denoted as R^2) are introduced to evaluate the MRR prediction performance. The expressions of the indicators are shown as follows.

$$MAD = \frac{1}{N} \sum_{i=1}^N |y_i - \bar{y}|, \quad (7)$$

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2, \quad (8)$$

$$RMSE = \sqrt{MSE}, \quad (9)$$

$$R^2 = 1 - \frac{\sum_{i=1}^N (y_i - \hat{y}_i)^2}{\sum_{i=1}^N (y_i - \bar{y})^2}, \quad (10)$$

where y_i is the true value of MRR for i^{th} wafer, $\bar{y}$ is the mean value of MRR for all the testing wafers, and $\hat{y}_i$ is the predictive value for i^{th} wafer from the regression model. RMSE is the squared root of the average squared difference between the estimated values and the actual values, which measures the quality of a regressor, and a value closer to zero is better. R^2 is an indication of goodness of fit and a measure of how well unseen samples are likely to be predicted by the model, and a value closer to 1.0 is better.

4. CASE STUDY

4.1. Data Description

In this section, we employ the dataset sourced from the 2016 PHM Data Challenge. This dataset includes variables related to specific wafers during the CMP process, with the primary predictive target being MRR. The raw training dataset consists of over 600,000 wafer run records, while the raw test dataset contains more than 100,000 wafer run records. A total of 25 variables are included in the dataset. Among these, the first six represent identification parameters and sampling timestamps, whereas the remaining 19 correspond to measurement variables during the CMP process. The identification parameters, such as Chamber ID and Stage, specify the processing location of the wafer. Wafer ID serves as a unique identifier for individual wafer process data. The measurement variables fall into five categories: usage, pressure, flow, rotation, and others. These variables serve as inputs to the model, while the output is defined as the mean value of all material removal rate results at a specific stage and wafer number.

4.2. Data Pre-processing

It is important to highlight that the dataset used in this study was gathered from actual CMP devices by the PHM society committee. As a result, the raw signal includes a considerable amount of outliers and missing data, necessitating proper treatment. To effectively prepare the data for feature extraction, preprocessing steps such as outlier detection and missing value imputation are critical. Generally, outliers are defined as unusually high or low values and can be identified using specific criteria. In this research, the 3σ rule is applied to detect outliers. Regarding missing values, this paper employs a random forest-based method for data imputation (Tang and Ishwaran, 2017). This approach transforms the missing data issue into a prediction problem aimed at estimating missing entries. Specifically, it performs regression analysis by considering each variable in turn as the dependent variable, with all other variables acting as predictors. Subsequently, a trained random forest model is utilized to predict the missing data for the target variable. The variable with the fewest missing values is selected as the dependent variable, while the remaining $n-1$ variables and the original dependent variable are combined to form a new feature matrix. This process completes the imputation for one independent variable, and a similar iterative procedure is applied based on the number of missing values in each column.

4.3. Feature Extraction & Selection

In the fields of machine learning and statistics, identifying meaningful features from raw data is crucial for simplifying intelligent models and improving prediction accuracy. In most application scenarios, selecting relevant features while excluding irrelevant ones plays an equally significant role in ensuring final prediction performance, as both can substantially impact the results. Consequently, feature extraction and selection are typically regarded as one of the most critical steps in modern machine learning methodologies. Optimal subsets of features are often derived through iterative processes, frequently requiring extensive trial-and-error efforts. Extracting useful features from raw data is a challenging task that usually demands deep domain expertise and experience in data analysis. Based on existing literature in the semiconductor industry, statistical features for each process variable are commonly extracted for modeling purposes (Laine, 2003). In this research, the mean, median, maximum, minimum, and standard deviation (STD) of usage variables were calculated. Following the extraction of all features from the raw data, a suitable feature selection method was employed to identify features with stronger correlations to the MRR. The goal of feature selection is to minimize the dimensionality of the feature set by pinpointing and preserving the most effective subset of features, thereby boosting prediction accuracy. Feature selection holds particular significance in semiconductor applications, as the number of dimensions can exceed hundreds after feature extraction, necessitating the removal of redundant features to streamline the model.

This paper excludes variables that are irrelevant to prediction outcomes. Specifically, machine ID, mechanical date, timestamp, wafer ID, and stage are excluded from model consideration. The MRR observations corresponding to a specific wafer ID and stage may include multiple time-stamped records; therefore, records under identical wafer-stage conditions are aggregated before modeling. To avoid relying only on visual inspection, the bimodal structure of the average MRR distribution is first examined using Hartigan's dip test (Hartigan and Hartigan, 1985). The test indicates a significant departure from unimodality ($D = 0.0627, p < 0.001$), supporting the presence of a multimodal distribution. A Gaussian mixture model (GMM) (Dempster *et al.*, 1977) is then fitted to the MRR density distribution, and the intersection of the two fitted Gaussian components is used as a quantitative reference for segmentation. Based on this procedure, $MRR = 128.59$ is selected as the cutoff between the low- and high-MRR regimes.

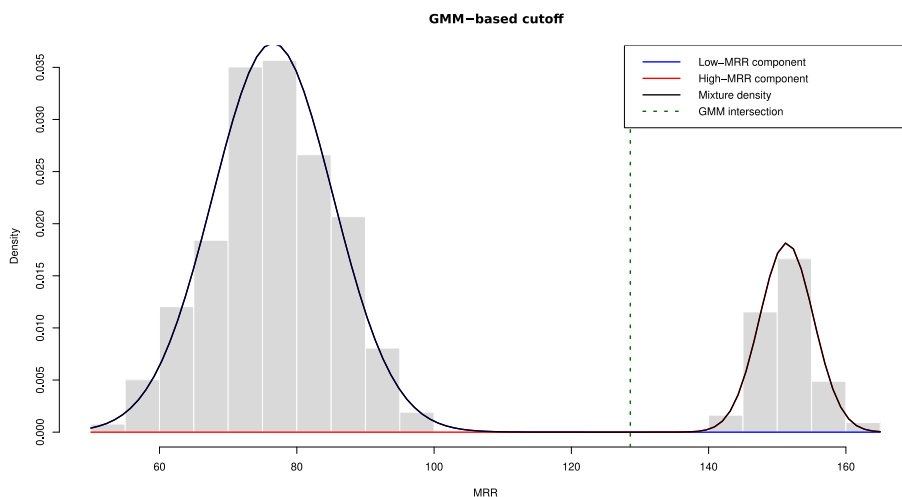


Figure 2. Gaussian mixture model-based identification of low- and high-MRR regimes

Each data point is subsequently classified into one of the two regimes according to whether its MRR is below or above 128.59 (Figure 2). Following modal segmentation, the training data are applied separately to each regime. After completing modal classification, the MRR within each regime is fitted using BART, which improves the interpretability of model performance across different MRR levels.

4.4. Hyperparameter Settings

To specify the priors of BART, the following hyperparameters need to be set: α , β , k , v , q , m . α and β are hyperparameters controlling the tree depth, usually set to 0.95 and 2, respectively. A common choice is $k = 2$, which ensures a 95% prior probability that the average MRR lies within the interval $(MRR_{\min}, MRR_{\max})$. A typical value for v ranges between 3 and 10, which determines the shape of the distribution. Common choices for q are 0.75, 0.90, or 0.99. The default values for (v, q) are (3, 0.90); Additionally, the number of trees m used in the BART model is another hyperparameter, set to 50. We refer the reader to Chipman *et al.* (2010) for a detailed discussion of these hyperparameters.

After completing the hyperparameter settings, the sampler in the form of “Bayesian backfitting” was first applied to sample the posterior distribution to obtain the average of 1,000 samples of the fitted values after the burn-in period, which was used as the final fitted values for the BART training set. The article’s evaluation of the BART model mainly focuses on the default parameter settings of the tree and model, as well as the training set that has undergone data processing, with 100 variables and 1977 observations, and the test set and validation set, each with the same variables and 424 observations.

4.5. Analysis of Results

The data points in the training dataset were used as input for the BART model. The predictive models generated by BART were subsequently validated on both the validation and test datasets. The performance metrics for MRR prediction, including MAD, RMSE, and R^2 , are 2.35, 9.24, and 0.98, respectively.

Figure 3 presents a plot comparing the actual average MRR values against the fitted values, along with 95% confidence intervals and prediction intervals derived from BART. Specifically, Figure 3(a) illustrates the comparison between the distribution of 95% credible intervals for the fitted values and the actual values, while Figure 3(b) depicts the comparison between the distribution of 95% prediction intervals for the fitted values and the actual values. In these plots, red dots indicate instances where the actual values fall outside their respective intervals, and blue lines represent the range of the respective intervals. Based on the interval coverage analysis of both graphs, it is evident that the BART model exhibits high prediction interval coverage, suggesting a strong likelihood that true values will fall within the prediction intervals. This underscores the robust predictive capability of the BART model.

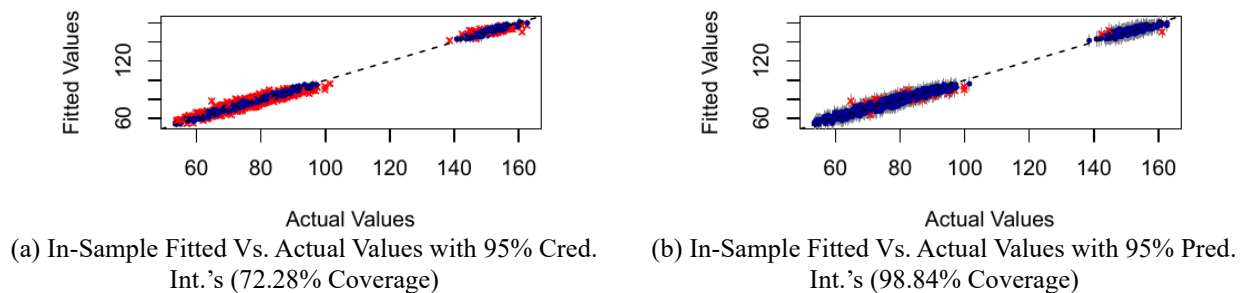


Figure 3. In-sample fitted vs. actual values

As shown in Figure 3(a) In-sample fitted vs. actual values with 95% credible intervals under the BART model; coverage = 76.28% due to incorporation of a noise term; and In-sample fitted vs. actual values with 95% predictive intervals; coverage improves to 98.84% is shown in Figure 3(b). Red points indicate observations where the true value falls outside the interval; blue vertical lines show the relevant credible or prediction intervals.

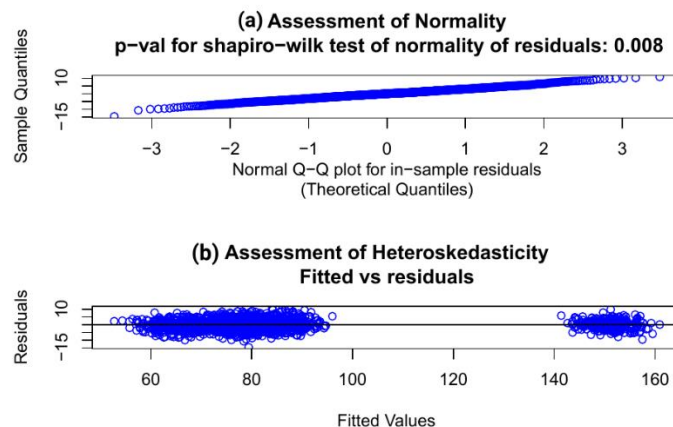


Figure 4. (a): Normal Q-Q plot and Shapiro–Wilk test ($p = 0.008$) assessing normality of in-sample residuals. (b): Fitted values vs. residuals plot used to assess heteroskedasticity in the BART model.

The normality and homoskedasticity of the residuals in the BART model are also analyzed. Figure 4(a) presents the normal Q-Q plot and the Shapiro-Wilk test result. Because the Shapiro-Wilk test has the null hypothesis that the residuals follow a normal distribution, $p = 0.008$ indicates a statistically significant deviation from normality. Since BART is a nonparametric predictive model and this study evaluates predictive performance primarily with RMSE-based metrics, this residual-normality result does not by itself invalidate the predictive comparison. The Q-Q plot suggests that most residuals are close to the theoretical quantile line, with the main deviations appearing in the tails. Figure 4(b) shows that the residual fluctuation range remains relatively stable as fitted values change, providing no strong visual evidence of severe heteroskedasticity.

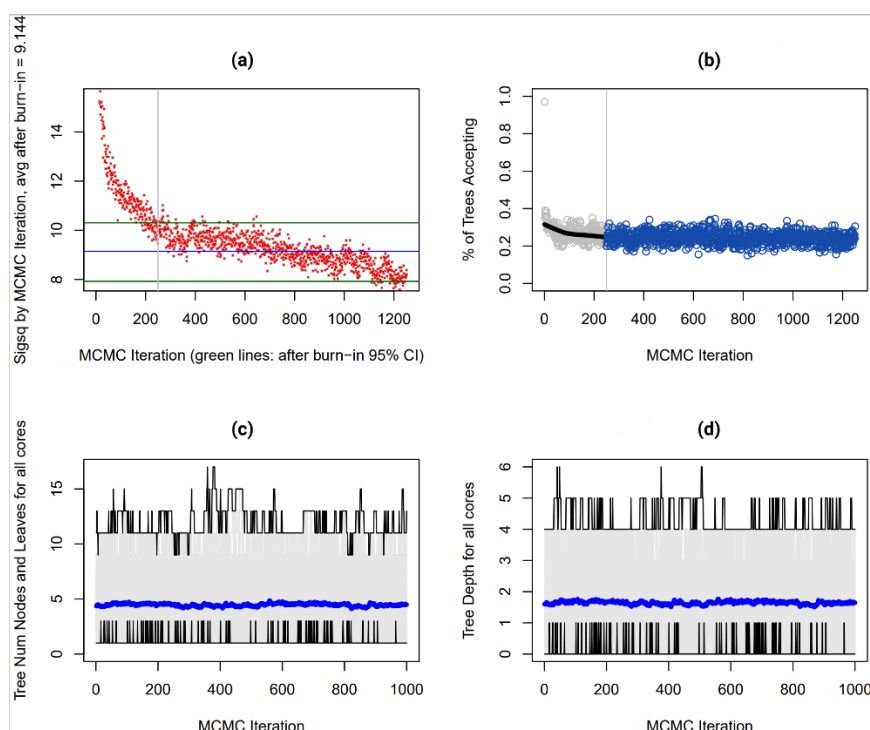


Figure 5. MCMC convergence diagnostics and tree complexity assessment for the BART model. (a) Posterior estimates of σ^2 across MCMC iterations; green lines denote the 95% credible interval after the burn-in period. (b) Percentage of accepting tree proposals across MCMC iterations. (c) Number of tree nodes and leaves after burn-in. (d) Tree depth across MCMC iterations after burn-in.

The diagnostic plots in Figure 5 summarize the stability and sampling behavior of the BART model during MCMC estimation. The estimate of σ^2 decreases during the early iterations and then stabilizes after burn-in, as shown in Figure 5(a). The acceptance-rate plot in Figure 5(b) shows that the percentage of accepted trees remains within a stable range after burn-in, indicating consistent MCMC sampling behavior. Figure 5(c) and Figure 5(d) further examine model complexity: the number of tree nodes/leaves and the tree depth fluctuate around stable levels across iterations, suggesting that the ensemble structure does not show systematic growth or collapse. Overall, these plots indicate that the BART model achieves acceptable post-burn-in stability while maintaining relatively consistent tree complexity.

The predictive performance of BART on the validation set is illustrated in Figure 6. Figure 6(a) and Figure 6(b) separately display the performance of the validation set in regimes where the average MRR is below 128.59 and above 128.59, while Figure 6(c) and Figure 6(d) present scatter plots of predicted values sorted by the magnitude of MRR. The RMSE equals 4.47 in the low-MRR regime, whereas the RMSE equals 3.22 in the high-MRR regime.

These RMSE values are calculated on the validation set and therefore differ from the overall test-set RMSE reported earlier. The test-set RMSE of 9.24 is relatively larger because RMSE gives greater weight to large errors, and a small number of high-error observations in the test set increase the overall RMSE. Therefore, the test-set RMSE of 9.24 and the validation-set RMSE values of 4.47 and 3.22 should not be directly compared as identical metrics, because they are calculated on different datasets and under different aggregation levels.

Next, this paper evaluates the overall contribution of the selected covariates to BART predictions using the omnibus test. The null hypothesis of the omnibus test is that the covariates do not provide statistically meaningful predictive information for average MRR. The p -value is calculated by comparing the observed Pseudo- R^2 with the distribution of Pseudo- R^2 values obtained after randomly shuffling the covariates. As shown in Figure 7, the replacement distribution lies entirely to the left of the blue line representing the observed Pseudo- R^2 , resulting in a p -value of zero. This result

rejects the null hypothesis and indicates that the selected covariates make a statistically meaningful overall contribution to the BART prediction model.

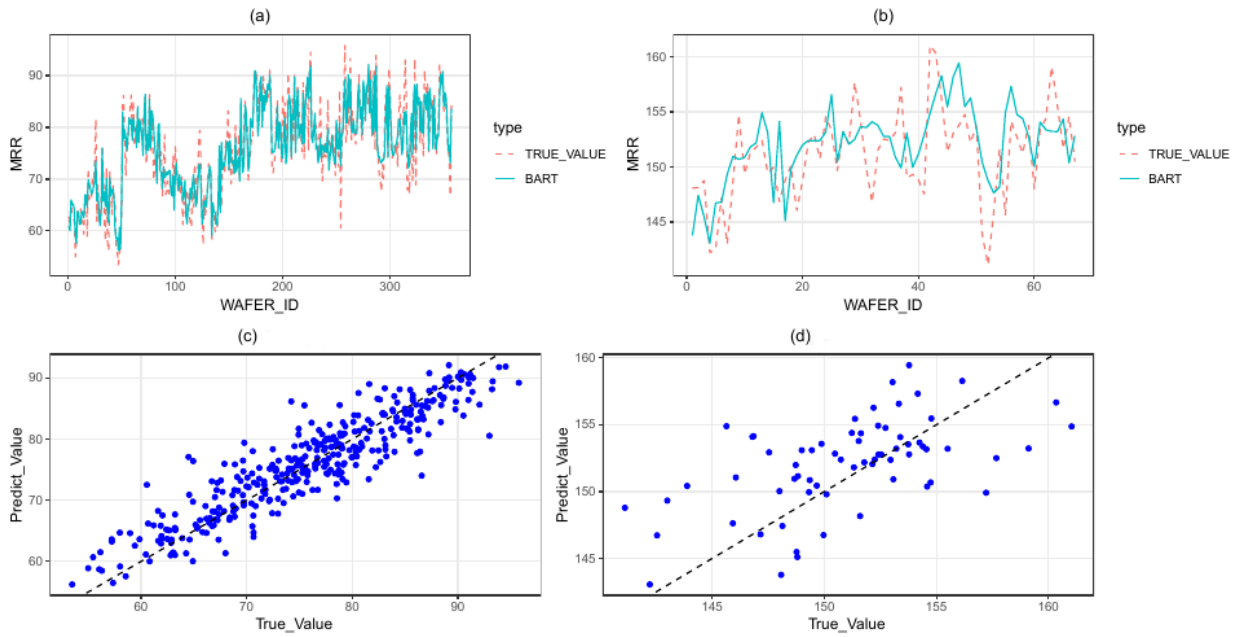


Figure 6. Predicted versus true material removal rate (MRR) on the validation set from BART, stratified by MRR level.

Figure 6(a) displays comparisons for wafers with average MRR < 128.59, and Figure 6(b) displays time-series comparisons for wafers with average MRR > 128.59. Figure 6(c) shows the corresponding scatter plot of predicted against true values for the low-MRR regime (average MRR < 128.59, RMSE = 4.47), and Figure 6(d) shows the scatter plot for the high-MRR regime (average MRR > 128.59, RMSE = 3.22).

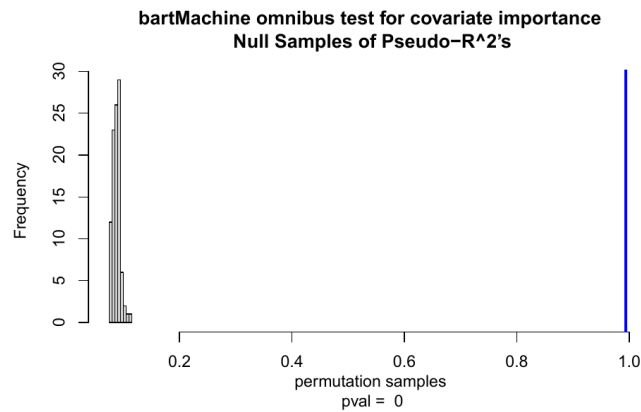


Figure 7. Omnibus test of covariate importance using the BART model.

Figure 8 presents the sensitivity of the BART model's out-of-sample RMSE to two hyperparameters, m and k . The left figure shows that as the number of trees m increases, the RMSE declines substantially at first and then levels off after around 100 trees, indicating improved predictive accuracy with diminishing returns from adding more trees. The right figure shows that RMSE varies across different values of k , but without a strong monotonic pattern, suggesting that model performance is less sensitive to k than to m in this experiment.

In BART, variable importance is typically assessed by the frequency of its appearance in the decision tree's partition rules. Therefore, this paper ranks variable importance using a local thresholding strategy and calculates the inclusion ratio (Bleich *et al.*, 2013) for each covariate based on its null distribution under substitution adjustments. This approach effectively identifies predictors with strong correlations to predicted values.

Figure 9 illustrates the top 20 variables ranked by variable inclusion ratio from highest to lowest. The blue bars represent a locally determined threshold derived from the aligned distribution, while the plotted points correspond to the average variable inclusion ratios under multiple null distributions. A variable is classified as relatively important only when its average inclusion ratio exceeds the local threshold. The results show that variables related to chamber conditions, dresser-table usage, and dresser usage exhibit relatively high importance for MRR prediction.

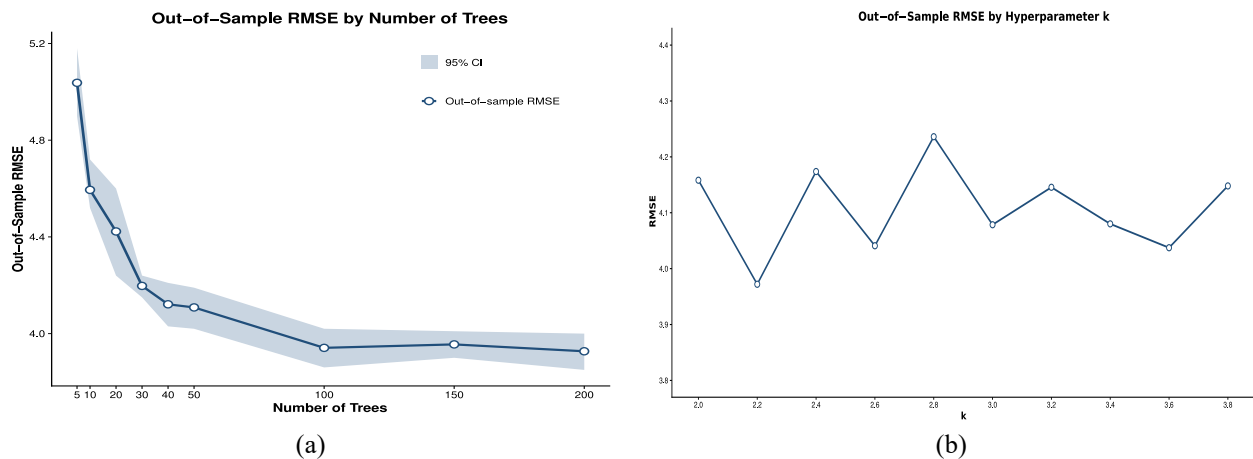


Figure 8. Sensitivity Analysis of Hyperparameters m and k in BART

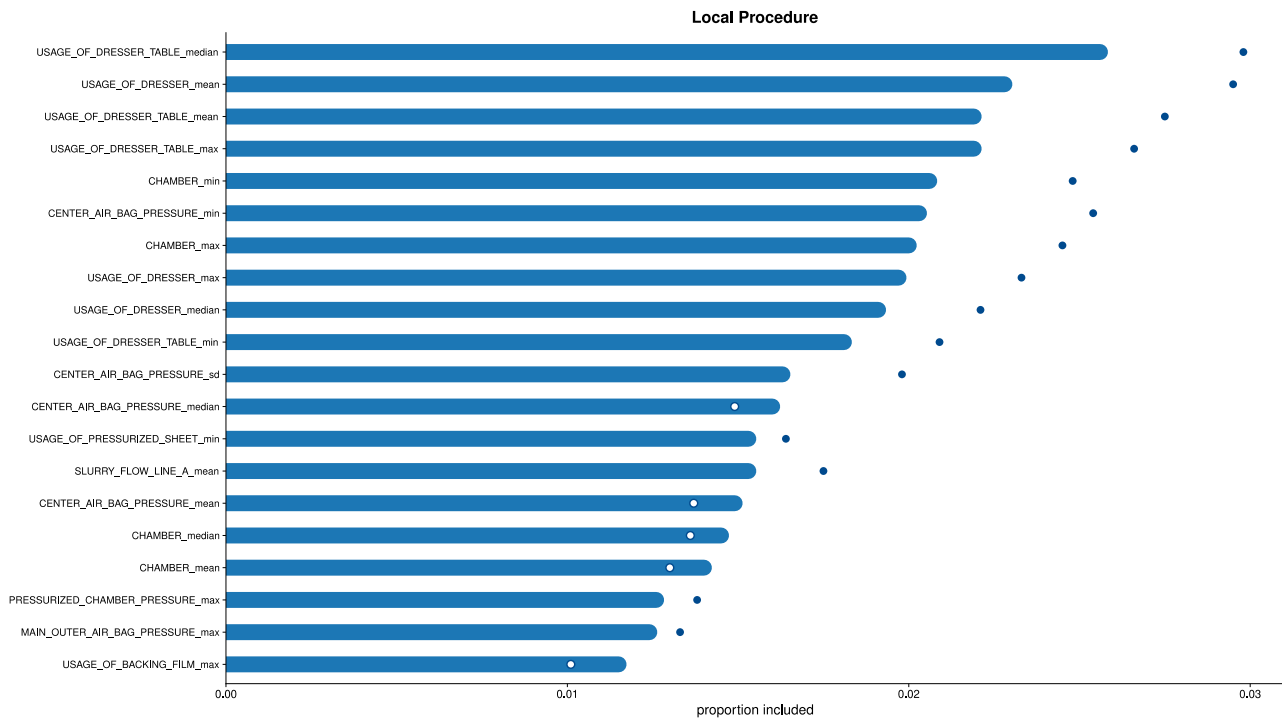


Figure 9. Variable importance ranking based on BART model inclusion proportions

4.6. Comparative Results

BART with default hyperparameters is further utilized as a control group to evaluate and compare the performance of various models, including BART-CV, Random Forest, GBM, and XGBoost. The root mean square error (RMSE) is adopted as the loss function for all models.

After grid search, the optimal hyperparameter combination for BART-CV is determined as $(k, v, q, m) = (5, 3, 0.9, 200)$. GBM and XGBoost are included as gradient-boosting baselines because they are competitive tree-based ensemble methods for nonlinear regression. For both models, the number of boosting iterations was set to 1,000, with a learning rate of 0.01 and a maximum tree depth of 3. For GBM, the minimum number of observations in each terminal node was set to 10, the bagging fraction was set to 0.8, and the full training set was used for model fitting. For XGBoost, the

subsampling ratio and column subsampling ratio were both set to 0.8, and the squared-error objective was used for regression. For the Random Forest, the number of trees was also set to 200, and the minimum number of samples required to split a node was set to 10.

The prediction results for MRR under different regimes are presented in Figure 10. Vertically, the first column reports results for the low-MRR regime, and the second column reports results for the high-MRR regime. Horizontally, each row illustrates the fitted curves of one model under the two regimes. In each subplot, the red line represents the true value of average MRR, while the blue line represents the predicted value.

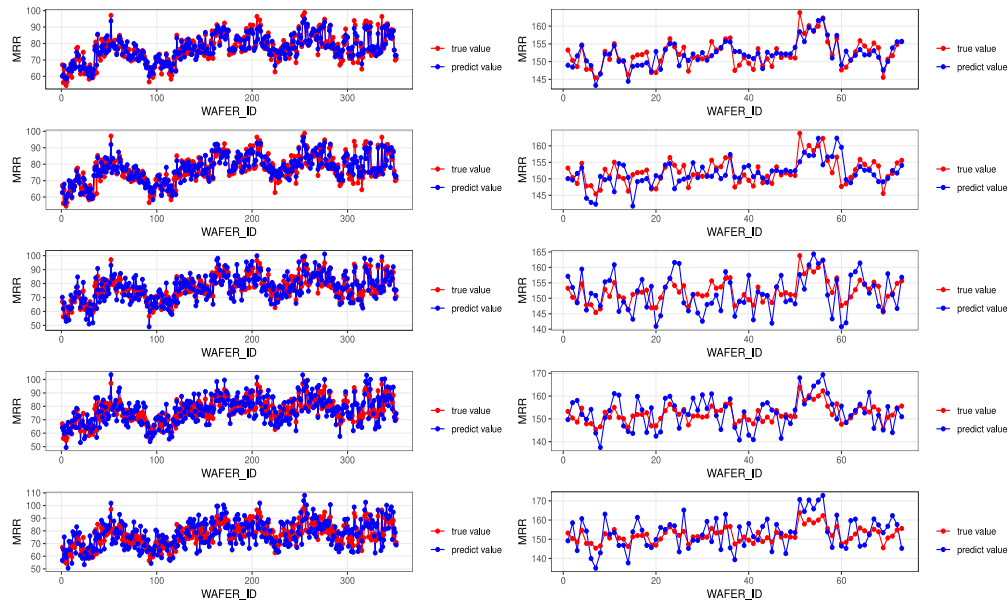


Figure 10. Predicted vs. true material removal rate (MRR) under different algorithms and MRR regimes. Left column: prediction results for the low-MRR regime using BART, BART-CV, GBM, Random Forest, and XGBoost. Right column: corresponding results for the high-MRR regime under the same five models. In each subplot, red lines represent true MRR values, and blue lines represent predicted MRR values.

To further compare these methods, this study evaluates RMSE across the two MRR regimes in Figure 11. In the low-MRR regime, BART-CV achieves the lowest RMSE of 3.6698, followed by XGBoost with 4.1175, Random Forest with 4.1438, BART with 4.2292, and GBM with 4.4122. These results indicate that BART-CV provides the strongest predictive accuracy in this regime, while BART remains comparable to the baseline models and additionally offers posterior uncertainty estimates and variable-importance information.

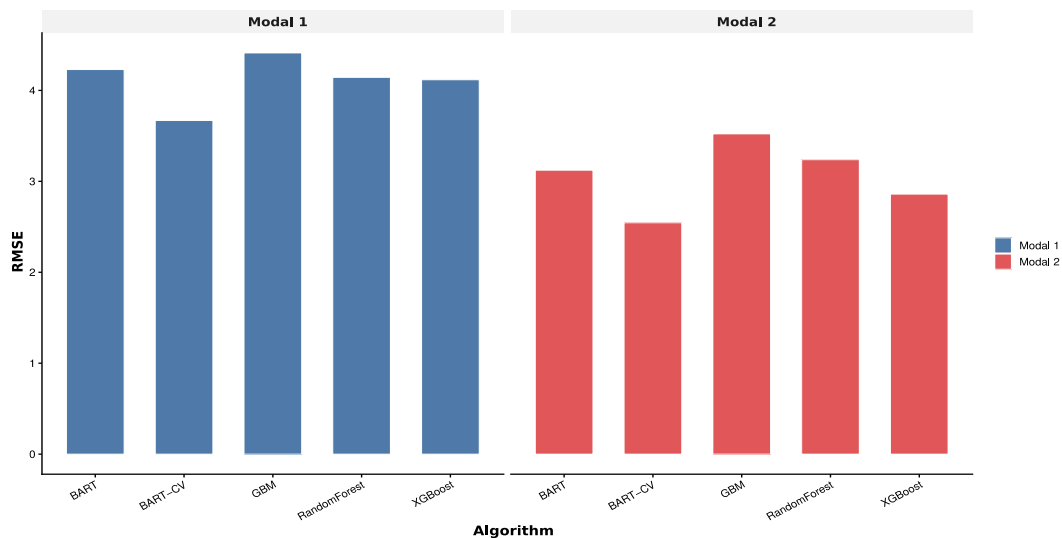


Figure 11. RMSE comparison of five models (BART, BART-CV, GBM, Random Forest, and XGBoost) across the low- and high-MRR regimes.

To assess whether the performance differences among the compared models are statistically meaningful, a Friedman test (Demšar, 2006) is conducted across repeated experimental results in Figure 12. The test indicates significant differences among the models ($p = 2.68\text{e-}07 < 0.05$). Combined with the RMSE comparison, this result supports the conclusion that BART provides a useful predictive modeling option for CMP MRR prediction under the present experimental setting.

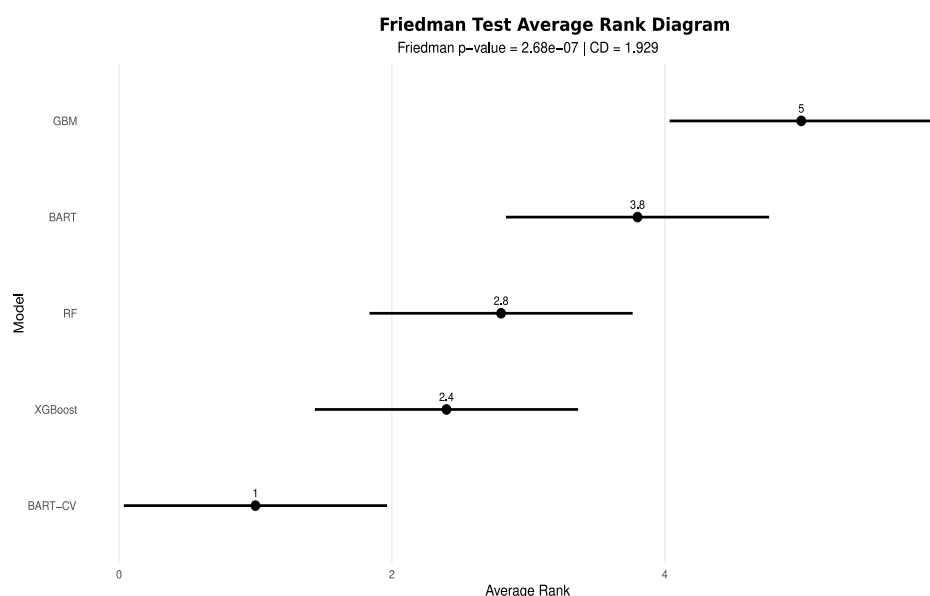


Figure 12. Friedman Test Average Rank Comparison Across Models

5. LIMITATIONS

Several limitations remain. First, although the data split avoids placing samples from the same batch simultaneously in the training and test sets where possible, future studies could use stricter temporal or production-batch extrapolation designs to better evaluate generalization in real manufacturing settings. Second, the current BART framework learns statistical relationships from process variables but does not explicitly encode polishing-unit geometry, Preston-type physical relationships, or process-feasibility constraints. Future research could incorporate physics-informed features, combine mechanistic CMP models with BART-based residual learning, or impose monotonicity, non-negativity, and feasibility constraints during modeling or post-processing.

6. CONCLUSION AND DISCUSSION

Machine learning models have been widely applied to predict the average material removal rate (MRR) for wafers in chemical mechanical planarization (CMP) processes within semiconductor manufacturing. This study applies Bayesian Additive Regression Trees (BART) to CMP MRR prediction after outlier handling, missing-value interpolation, statistical feature extraction, and feature selection. The model is evaluated on the 2016 PHM Challenge dataset and compared with representative baselines including BART-CV, Random Forest, GBM, and XGBoost. The results show that BART provides competitive predictive performance while offering posterior uncertainty estimates and variable-importance analysis. The bimodal MRR segmentation is further supported by Hartigan's dip test and a GMM-based threshold, improving the reproducibility of the experimental design. The findings indicate that BART can serve as a useful data-driven tool for CMP process prediction and diagnostic analysis.

The variable-importance analysis indicates that chamber-related variables and dresser-related usage variables are strongly associated with MRR prediction in the present dataset. These findings may help engineers focus diagnostic attention on relevant process conditions. Nevertheless, the variable-importance results should be viewed as references for process monitoring and future intervention design rather than as direct evidence of causal effects. Future work should further integrate physical CMP constraints, conduct stricter temporal or batch-based validation, and examine model robustness under broader production conditions.

REFERENCES

- Bleich, J., Kapelner, A., George, E. I., and Jensen, S. T. (2013). Variable Selection for BART: An Application to Gene Regulation. *The Annals of Applied Statistics*, 8, 1750-1781.
- Chen, T., and Guestrin, C. (2016). *Xgboost: A Scalable Tree Boosting System*. in Proceedings of The 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (Pp. 785-794). ACM.
- Chipman, H., and McCulloch, R. (2024). Bayestree: Bayesian Additive Regression Trees. CRAN. Retrieved from: <https://cran.r-project.org/web/packages/bayestree/bayestree.pdf>.
- Chipman, H. A., George, E. I., and McCulloch, R. E. (2007). Bayesian Ensemble Learning. In B. Schölkopf, J. Platt, and T. Hoffman (Eds.), *Advances in Neural Information Processing Systems* 19 (Pp. 265-272). MIT Press.
- Chipman, H. A., George, E. I., and McCulloch, R. E. (2010). BART: Bayesian Additive Regression Trees. *The Annals of Applied Statistics*, 4(1), 266-298.
- Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data Via The EM Algorithm. *Journal of The Royal Statistical Society: Series B*, 39(1), 1-38.
- Demšar, J. (2006). Statistical Comparisons of Classifiers Over Multiple Data Sets. *Journal of Machine Learning Research*, 7, 1-30.
- Evans, C. J., Paul, E., Dornfeld, D. A., Lucca, D. A., Byrne, G., Tricard, M., Klocke, F., Dambon, O., and Mullany, B. A. (2003). Material Removal Mechanisms in Lapping and Polishing. *CIRP Annals - Manufacturing Technology*, 52(2), 611-633.
- Friedman, J. H. (2001). Greedy Function Approximation: A Gradient Boosting Machine. *The Annals of Statistics*, 29(5), 1189-1232.
- Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis* (3rd Ed.). Chapman and Hall/CRC.
- Geman, S., and Geman, D. (1984). Stochastic Relaxation, Gibbs Distributions, and The Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 721-741.
- Hartigan, J. A., and Hartigan, P. M. (1985). The Dip Test of Unimodality. *The Annals of Statistics*, 13(1), 70-84.
- Hastie, T., and Tibshirani, R. (2000). Bayesian Backfitting. *Statistical Science*, 15(3), 196-213.
- Kapelner, A., and Bleich, J. (2013). *Bartmachine: Machine Learning with Bayesian Additive Regression Trees*. Arxiv Preprint Arxiv:1312.2171.
- Laine, S. (2003). *Using Visualization, Variable Selection and Feature Extraction to Learn from Industrial Data*. Helsinki University of Technology.
- Lee, H., and Jeong, H. (2011). A Wafer-Scale Material Removal Rate Profile Model for Copper Chemical Mechanical Planarization. *International Journal of Machine Tools and Manufacture*, 51(5), 395-403.
- Li, Z., Wu, D., and Yu, T. (2019). Prediction of Material Removal Rate for Chemical Mechanical Planarization Using Decision Tree-Based Ensemble Learning. *Journal of Manufacturing Science and Engineering*, 141(3), 031003.
- Lim, K.-L., and Dutta, R. (2020). Material Removal Rate Prediction Using The Classification-Regression Approach. In 2020 IEEE 22nd Electronics Packaging Technology Conference (EPTC) (Pp. 172-175).
- Nagaraj, Y., Jagannatha, N., Sathisha, N., and Niranjana, S. J. (2021). Prediction of Material Removal Rate and Surface Roughness in Hot Air Assisted Hybrid Machining on Soda-Lime-Silica Glass Using Regression Analysis and Artificial Neural Network. *Silicon*, 13, 4163-4175.

- Pate, K., and Safier, P. (2022). *Chemical Metrology Methods for CMP Quality*. In *Advances in Chemical Mechanical Planarization (CMP)* (2nd Ed., Pp. 355-383).
- Tang, F., and Ishwaran, H. (2017). Random Forest Missing Data Algorithms. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 10, 363-377.
- Wang, P., Gao, R. X., and Yan, R. (2017). A Deep Learning-Based Approach to Material Removal Rate Prediction in Polishing. *CIRP Annals*, 66(1), 429-432.
- Wei, Y., and Wu, D. (2024). Material Removal Rate Prediction in Chemical Mechanical Planarization with Conditional Probabilistic Autoencoder and Stacking Ensemble Learning. *Journal of Intelligent Manufacturing*, 35, 115-127.
- Xia, L., Zheng, P., Huang, X., and Liu, C. (2021). A Novel Hypergraph Convolution Network-Based Approach for Predicting The Material Removal Rate in Chemical Mechanical Planarization. *Journal of Intelligent Manufacturing*, 33(8), 1-12.
- Xie, G. D., Gao, B., Yao, X. Y., and Yuan, Z. J. (2006). Study of Local Material Removal Model of Bonnet Tool Polishing. *Key Engineering Materials*, 536(304-305), 335-339.
- Zhang, J. S., Jiang, Y. C., Luo, H., and Yin, S. (2021). Prediction of Material Removal Rate in Chemical Mechanical Polishing Via Residual Convolutional Neural Network. *Control Engineering Practice*, 107, 104673.
- Zhang, S., Shih, T. Y., and Müller, P. (2007). A Spatially-Adjusted Bayesian Additive Regression Tree Model to Merge Two Datasets. *Bayesian Analysis*, 2(3), 611-633.