

Article

Accelerating Elastic Property Prediction in Fe-C Alloys through Coupling of Molecular Dynamics and Machine Learning

Sandesh Risal ¹, Navdeep Singh ^{2,*}, Yan Yao ^{3,4}, Li Sun ¹, Samprash Risal ³ and Weihang Zhu ^{1,5,*}

¹ Department of Mechanical Engineering, University of Houston, Houston, TX 77204, USA; srisal@cougarnet.uh.edu (S.R.); lsun3@central.uh.edu (L.S.)

² Department of Mechanical Engineering, School of Engineering and Computer Science, University of the Pacific, Stockton, CA 95211, USA

³ Materials Science and Engineering Program, University of Houston, Houston, TX 77204, USA; yyao4@central.uh.edu (Y.Y.); srisal2@cougarnet.uh.edu (S.R.)

⁴ Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77204, USA

⁵ Department of Engineering Technology, University of Houston, Houston, TX 77204, USA

* Correspondence: nsingh1@pacific.edu (N.S.); wzhu21@central.uh.edu (W.Z.)

Abstract: The scarcity of high-quality data presents a major challenge to the prediction of material properties using machine learning (ML) models. Obtaining material property data from experiments is economically cost-prohibitive, if not impossible. In this work, we address this challenge by generating an extensive material property dataset comprising thousands of data points pertaining to the elastic properties of Fe-C alloys. The data were generated using molecular dynamic (MD) calculations utilizing reference-free Modified embedded atom method (RF-MEAM) interatomic potential. This potential was developed by fitting atomic structure-dependent energies, forces, and stress tensors evaluated at ground state and finite temperatures using ab-initio. Various ML algorithms were subsequently trained and deployed to predict elastic properties. In addition to individual algorithms, super learner (SL), an ensemble ML technique, was incorporated to refine predictions further. The input parameters comprised the alloy's composition, crystal structure, interstitial sites, lattice parameters, and temperature. The target properties were the bulk modulus and shear modulus. Two distinct prediction approaches were undertaken: employing individual models for each property prediction and simultaneously predicting both properties using a single integrated model, enabling a comparative analysis. The efficiency of these models was assessed through rigorous evaluation using a range of accuracy metrics. This work showcases the synergistic power of MD simulations and ML techniques for accelerating the prediction of elastic properties in alloys.

Keywords: machine learning; molecular dynamics; density functional theory; MEAMfit; RF-MEAM; elastic properties; mechanical properties



Citation: Risal, S.; Singh, N.; Yao, Y.; Sun, L.; Risal, S.; Zhu, W. Accelerating Elastic Property Prediction in Fe-C Alloys through Coupling of Molecular Dynamics and Machine Learning. *Materials* **2024**, *17*, 601. <https://doi.org/10.3390/ma17030601>

Academic Editors: Jaka Burja and Grega Klančnik

Received: 22 December 2023

Revised: 18 January 2024

Accepted: 24 January 2024

Published: 26 January 2024



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

1. Introduction

The prediction of material properties using computational tools can serve as a crucial guide to the experimental material design process [1]. Computational methods like density functional theory (DFT) and molecular dynamics (MD) are able to compute material properties accurately, allowing researchers to gain meaningful and prompt insights into the material properties of interest [2]. However, their applicability is constrained by limitations in length (restricted to a few nanometers [3,4]) and time scale (limited to a few nanoseconds [5–7]) that can be explored efficiently and effectively.

In recent years, machine learning (ML) has gained popularity for predicting material properties as evidenced by notable studies [8–12]. For ML, the data availability is a significant hurdle. The integration of ML with these multi-level computational material tools addresses the individual limitations of each method, accelerating the novel material

discovery. This synergy solves the data scarcity in ML by generating a plethora of data using DFT and MD and the time limitation of the computational tools by using ML that further expedites the materials exploration process.

Numerous studies have harnessed ML models to predict mechanical properties, including tensile strength, yield strength, hardness, and elongation, across various alloy systems. Here, we present a literature review that underscores the diverse applications of ML in predicting alloy mechanical properties and subsequently focus on its application in predicting steel properties.

The quest to predict the mechanical properties of alloys using machine learning has seen significant advancements across various alloy systems. Hu et al. [13] proposed a novel feature engineering method integrating chemical compositions and manufacturing processes, achieving high prediction accuracy for wrought aluminum alloys. Deng et al. [14] further refined this approach, employing a sequential minimal optimization algorithm to predict copper-aluminum alloys' tensile strength and hardness. With the guidance of their model, they successfully developed an alloy meeting specific mechanical property targets.

Devi et al. [15] compared multiple ML models for predicting aluminum alloys' tensile strength and hardness, demonstrating the efficacy of algorithms like K-nearest neighbors (KNN) and artificial neural networks (ANN) in this domain. Similarly, Xu et al. [16] employed ANN and support vector machine (SVM) models to predict properties in magnesium alloys, showcasing the potential for relating composition, processing, and mechanical properties.

Choyi [17] applied ML techniques to screen and identify promising alloy compositions for experimental testing in aluminum alloys. Ling [18] extended this approach by utilizing ensemble ML to predict mechanical properties and phase transformation temperatures for various alloy classes, enabling an inverse design framework for creating new alloys meeting specific specifications.

Lu et al. [19] focused on magnesium-rare earth alloys, utilizing SVR-based ML models to predict multiple mechanical properties with considerable accuracy. Tan et al. [20], employing a simpler ML model, provided a foundational tool for predicting mechanical properties in multi-principal element alloys based solely on composition data.

Zhang et al. [21] and Lee et al. [22] explored steel alloys, utilizing ML to predict tensile strength and yield strength while employing various algorithms and input parameters. Shen et al. [23] pioneered a physical metallurgy-guided ML approach, integrating equilibrium volume fraction and driving force for precipitation into their models for predicting hardness, leading to the design of ultrahigh-strength stainless steel with remarkable accuracy. Further extending these methodologies, Mahfouf [24], Gaffour et al. [25], Dutta et al. [26], and Pattanayak et al. [27] utilized a range of ML techniques such as artificial neural networks (ANN), genetic algorithms (GA), and hybrid models to model mechanical properties, optimize compositions, and predict mechanical performance across various steel alloys, showcasing the potential of these methods in alloy design and optimization.

A thorough review of the literature [13–27] above demonstrates that various ML algorithms such as LR, SVM/SVR, RF, ANN, KNN, and GPR are effective at making predictions. Notably, some studies have also explored ensemble techniques such as XGBoost, AdaBoost, and SL to enhance predictive accuracy. The majority of these studies utilized composition and process parameters as their primary inputs. Among the commonly employed process parameters were heat treatment temperature, heating duration, choice of cooling medium, and others. Hence, a broad spectrum of strategies have been applied in materials property prediction through ML methodologies.

In this work, we aim to demonstrate a framework to predict the elastic properties of iron-carbon systems using the synergistic effect of multiscale computational material tools (DFT and MD) and ML. MD calculations using the interatomic potential, produced by fitting on DFT-calculated data, generate the dataset, which is subsequently utilized to train the ML model for the expedited prediction of elastic properties.

2. Data and Methods

2.1. Potential Development

Potential development methodology and calculation details have been discussed in our previous publication [28]. Here, we briefly summarize the process to provide a complete picture of the framework that integrates the previous work with the ML model for elastic properties prediction.

The inter-atomic potential was developed by fitting the potential parameters to the forces, energies, and stress tensors of various ordered structures of Fe-C calculated by first principle calculations using Vienna Ab initio Simulation Package (VASP) [29] with the projector augmented wave (PAW) [30] pseudopotentials. The exchange-correlation function was calculated using the Perdew–Burke–Ernzerhof Generalized-Gradient Approximation (GGA-PBE) [31]. These precise data were fitted into 78 parameters of RF-MEAM potential [32] using a potential fitting code called MEAMfit [33]. Thus, the produced potential was used to perform MD calculation on LAMMPS [34] to reproduce the properties of the structures used during the fitting process. The potential was further verified by calculating elastic moduli for the disordered structures at various finite temperatures.

2.2. Data Collection/Calculation

The training data for ML were generated through MD simulations using the developed RF-MEAM potential. These calculations were performed on various ordered and disordered structures. The ordered structures used are B1, B3, and Cementite, whereas the disordered alloy structures used were Fe-C with fcc and bcc base structures, each with octahedral and tetrahedral interstitial carbon atoms. For these disordered alloys, the composition of carbon was varied. The carbon concentrations used were 1, 2, 3, 4, 5, 6, 8, 10, 12, 14, 16, 18, and 20 atomic percentage. These structures were subjected to temperatures from 0 K to 1200 K in 50 K increments and elastic properties were calculated at each temperature. In total, 1375 data points were obtained and used for this study.

The atomic structure is initially equilibrated at the desired temperature using the NPT ensemble, allowing the system to equilibrate under constant temperature and pressure conditions. The equilibrated structures are then subjected to positive and negative deformations (i.e., deformation applied along specific axes in two opposite directions) in six directions, corresponding to the Voigt deformation components. These deformations were applied to investigate the material's response to mechanical stress. After each deformation, the structures were equilibrated again using the NVE ensemble. The resultant change in stress within the structure was measured after each deformation. These data were used to compute the elastic stiffness tensor, which characterizes the material's response to mechanical deformation. Once we have the elastic stiffness tensors, Voigt equation

$$B_V = \frac{1}{9}(c_{11} + c_{22} + c_{33}) + \frac{2}{9}(c_{12} + c_{13} + c_{23}), \quad (1)$$

and

$$G_V = \frac{1}{15}(c_{11} + c_{22} + c_{33} - c_{12} - c_{13} - c_{23}) + \frac{1}{5}(c_{44} + c_{55} + c_{66}), \quad (2)$$

Reus equation

$$B_R = \frac{1}{(s_{11} + s_{22} + s_{33}) + 2(s_{12} + s_{13} + s_{23})}, \quad (3)$$

and

$$G_R = \frac{1}{4(s_{11} + s_{22} + s_{33}) - 4(s_{12} + s_{13} + s_{23}) + 3(s_{44} + s_{55} + s_{66})}, \quad (4)$$

and Hill equation

$$B_H = \frac{1}{2}(B_R + B_V) \quad (5)$$

and

$$G_H = \frac{1}{2}(G_R + G_H) \quad (6)$$

are used to calculate the macro elastic constants like bulk modulus, and rigidity modulus. The Young's modulus can be calculated using equation

$$E_H = \frac{9B_H G_H}{3B_H + G_H}. \quad (7)$$

These three elastic constants are used as the target properties for our ML study.

2.3. Input Parameters

The input parameters for ML should be meticulously selected to correlate well with the output target properties. So, we initially did a simple visual analysis by plotting the target properties vs. the possible input parameters. The purpose of this visual test is to observe if there is any relationship between the input and the output no matter how complicated the relation is. The input parameters we used are temperature, composition, base lattice structure type, interstitial sites, Ordered or Disordered, and three lattice constants a , b , and c .

Next, feature importance is calculated to see the effect of each input feature on the target properties. We used an RF regressor to calculate and feature importance library from yellow-brick to visualize feature importance.

2.4. ML Algorithms

Various ML algorithms were employed in this study to perform predictions using two distinct approaches, as shown in Figure 1. The first approach uses a multi-variate model to predict both the bulk and rigidity modulus and subsequently calculate Young's modulus from these simultaneously predicted results. The second approach utilizes two separate models, each trained independently to predict their respective target properties, and then Young's modulus is computed from the calculated bulk (B) and rigidity (G) modulus values.

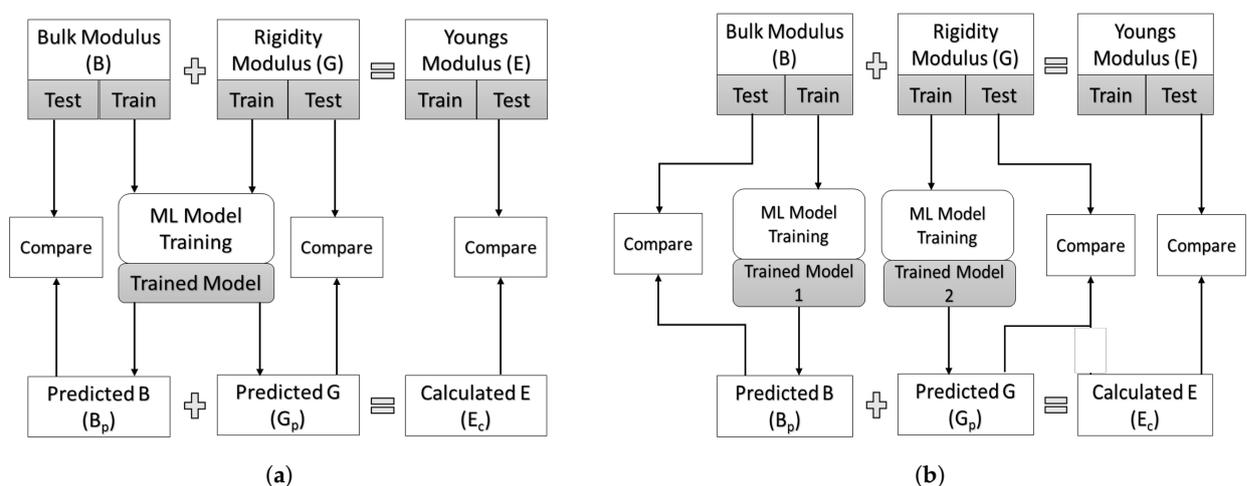


Figure 1. Two different approaches used for the properties prediction: (a) multivariate method, and (b) independent method.

This study employed a selection of individual algorithms known for their effectiveness in predicting material properties. Additionally, the ensemble algorithm named SL is utilized, which has demonstrated efficacy in combining the strengths of individual algorithms, making it particularly well-suited for small to medium-sized datasets, such as the one used in this study. The specific algorithms used in this study, along with the SL, are discussed below.

2.4.1. Random Forest (RF)

RF is a bagging algorithm technique that combines multiple decision trees to determine the final result rather than relying on individual decision trees [35]. RF has numerous decision trees as base learning models. In RF, row and feature sampling are performed from the dataset creating sample datasets for every model. This part is called Bootstrap. Every decision tree has high variance, but when we combine them all in parallel, the resultant variance is low as each decision tree is perfectly trained on that particular sample data. Hence, the output does not depend on one decision tree but on multiple decision trees [36]. For a classification problem, the final output is taken using the majority voting classifier. Likewise, for a regression problem, the final output is the mean of all the outputs. This part is Aggregation [37].

2.4.2. Extreme Gradient Boosting (XGBoost)

XGBoost is a ML method that combines the predictions of multiple weak Decision Tree (DT) models to produce a more robust prediction [38]. Initially, a model is built from the training data. Then, the second model is built to rectify the errors in the first model. The procedure is continued, and models are added until the complete training data set has been predicted or classified correctly or the maximum number of models has been added.

XGBoost, operating through a sequential process, constructs decision trees. All the independent variables fed into the decision tree are assigned weights, which play an essential role in XGBoost. The weight of variables predicted wrong by the tree is increased, and these variables are then fed to the second decision tree. These individual classifiers/predictors then ensemble to give a strong and more precise model. It can work on regression, classification, ranking, and user-defined prediction problems [39].

2.4.3. Support Vector Machine (SVM)

SVM is a robust supervised ML algorithm used for both classification and regression tasks [40]. SVM seeks the optimal hyperplane that best separates data points belonging to different classes while maximizing the margin between the classes. Support vectors, which are the data points closest to the decision boundary, are crucial in determining the hyperplane [41]. SVM works well for both linear and non-linear classification problems using kernel functions such as linear, polynomial, sigmoid kernels, and radial basis functions (RBF).

SVM is known for its ability to perform well in high-dimensional spaces and its effectiveness in cases where the data are not linearly separable. It is widely used in applications like image classification, text classification, and bioinformatics [42].

2.4.4. K-Nearest Neighbor (KNN)

KNN is a simple yet effective supervised ML algorithm used for classification and regression tasks [43]. KNN operates on the principle of identifying the 'K' nearest data points within the training set to a new, unseen data point and making predictions based on the majority class (for classification) or averaging (for regression) among those 'K' neighbors [44]. The choice of K, the number of nearest neighbors, is a critical parameter that affects the algorithm's performance.

KNN is a non-parametric and instance-based algorithm, meaning it does not make explicit assumptions about the underlying data distribution. It works well for linear and non-linear data and is particularly useful for problems with complex or irregular decision boundaries [45]. However, its computational complexity increases with the size of the training dataset, making it less suitable for large-scale applications [46].

2.4.5. MultiLayer Perceptron (MLP)

MLP, the most widely used neural network structure, consists of multiple layers of three types, i.e., input layer, output layer, and multiple hidden layers. The input layer receives the input signals to be processed. The output layer performs the required tasks,

such as prediction and classification [47]. The hidden layers, placed in between the input and output layers, are the true computational engine of the MLP. Data passes in a forward path from the input to the output layer in MLP, equivalent to a network in feed-forward. The backpropagation learning technique is used to train all the nodes in the MLP. MLPs can fix issues that are not linearly separable and are structured to approximate every continuous function [48].

As with all neural networks, the dimension of the input vector dictates the number of neurons in the input layer, while the number of classes to be learned dictates the number of neurons in the output layer. The number of chosen hidden layers and neurons in each layer must be empirically determined. As a rule of thumb, the neurons in the hidden layers are chosen as a fraction of those in the input layer. However, there is a trade-off regarding the number of neurons: Too many neurons produce over-training; too few neurons affect generalization capabilities [49]. In this study, three hidden layers are used and the number of neurons in each layer is determined by the hyperparameter optimization.

2.4.6. Gaussian Process Regression (GPR)

GPR is a probabilistic ML algorithm that, unlike many other ML models, allows for the prediction of the underlying uncertainties in its predictions [50]. GPR's adaptability is another standout feature; it excels at modeling complex relationships in data without requiring rigid functional forms or predefined features. This adaptability is highly beneficial when dealing with noisy or limited datasets, enabling GPR to capture intricate patterns effectively [51].

However, GPR's suitability depends on the specific characteristics of the problem. While it excels in scenarios with small datasets and noisy observations, it performs optimally in low-dimensional problems, stable design spaces, and moderately sized datasets [52]. Despite this specialization, GPR remains an invaluable tool in various domains, where its ability to provide probabilistic predictions and account for uncertainties makes it a vital asset in critical decision-making processes. Whether in geostatistics [53], finance [54], robotics [55], or materials [51], GPR's probabilistic nature ensures that decision-makers receive accurate predictions and gain crucial insights into the confidence level associated with those predictions, ultimately enhancing the quality of informed decisions.

2.4.7. Super Learner (SL)

SL is an ensemble ML approach designed to improve predictive accuracy and robustness by combining multiple base models or learning algorithms, encompassing a wide array of ML algorithms, from decision trees to neural networks. The concept aims to find the optimal combination of models to achieve the best possible prediction performance [56]. These single algorithm (base) models are trained on the same dataset, and their predictions are recorded as "meta-features". Then, a meta-learner is introduced to the mix, which learns how to combine or weight the base models' predictions optimally. This meta-learner, which can be as simple as linear regression or more complex like a neural network, aims to extract the best predictive insights from the collective knowledge of the base models.

SL's appeal lies in its ability to deliver superior predictive performance consistently. It often outperforms any single base model in isolation by leveraging various models and learning algorithms. Furthermore, it enhances the robustness of predictions by reducing the risk of overfitting, as it avoids relying too heavily on the peculiarities of any one model [57]. Its flexibility is also noteworthy, as it can adapt to various data types and domains, offering a versatile solution for diverse ML challenges. Overall, the SL serves as a valuable tool for data scientists and ML practitioners seeking to maximize prediction accuracy and automate the model selection process while benefiting from the collective intelligence of multiple models [58].

3. Results and Discussion

3.1. Potential Verification

The developed interatomic potentials (i.e., different potentials fitted on various combinations of structures) were used to reproduce the elastic properties of various ordered and disordered alloys of Fe-C.

At first, energy as a function of volume was calculated for the ordered structures and fitted to the Birch–Murnaghan equation of state to obtain the bulk modulus of elasticity and its pressure derivative. These outputs were compared to experimental and DFT results in order to assess the accuracy and reliability of the potentials, allowing us to identify the best potential. Upon comparing the results from the potentials fitted on various distinct dataset combinations, it was found that the potential fitted on the dataset containing B1 and Cementite reproduced most of the alloy results faithfully. The RF-MEAM interatomic potential, referred to hereafter, is the potential fitted on the B1 and Cementite datasets. Table 1 details the results for equilibrium volume (V_0), bulk modulus (B_0), and pressure derivative of bulk modulus (B_0') produced by our best DFT potential in comparison to the experimental and DFT results from the literature. From the table, it is apparent that the results from our best potential are in very good agreement with the literature.

Table 1. Comparison of our results with the published DFT data for equilibrium volume, bulk modulus, and pressure derivative calculated by the Birch–Murnaghan equation of state.

Structure	V_0 (\AA^3)		B_0 (GPa)		B_0'	
	RF-MEAM	Literature	RF-MEAM	Literature	RF-MEAM	Literature
B1 (FeC)	65	64 ^{a,b}	338	329 ^a	5.28	4.40 ^a
Cementite (Fe ₃ C)	155	154 ^a	252	234 ^a	4.05	4.00 ^a

^a Henriksson et al. [59]; ^b Chihi et al. [60].

Figure 2 compares the cohesive energy versus volume curve for (a) B1 and (b) Cementite with the DFT and experimental results, respectively, as reported by Lalitha et al. [61]. As seen in the figure, our calculated curves are in good agreement with the reported literature, with the equilibrium volume deviation of 2.4%, and 1.2% for B1, and Cementite, respectively.

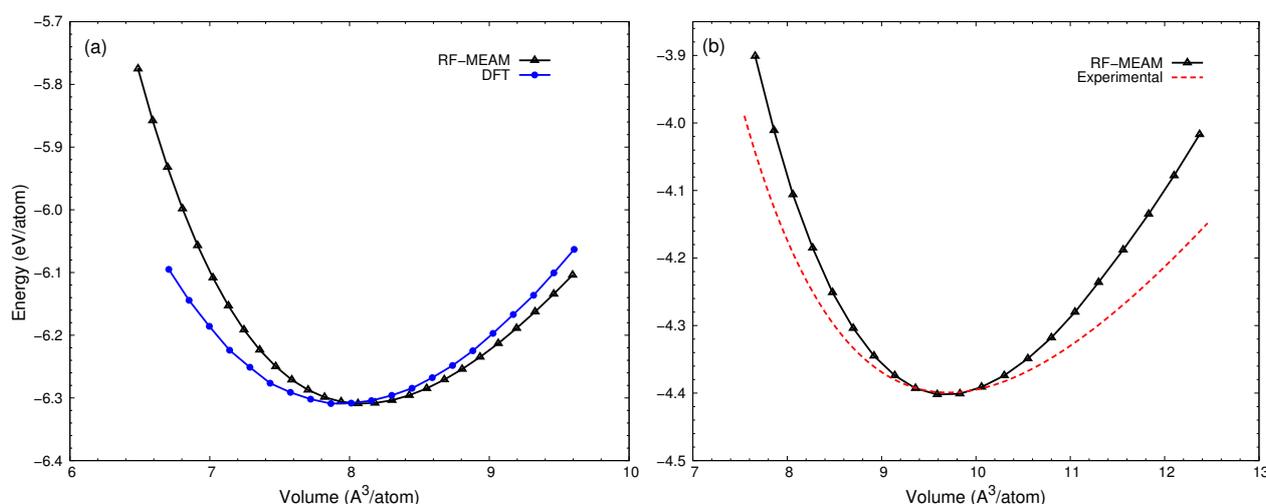


Figure 2. Comparison of energy vs. volume curves for (a) B1, and (b) Cementite with experimental and DFT results. Both DFT and experimental results used for the comparison are from Lalitha et al. [61].

The best RF-MEAM potential was then used to reproduce the elastic properties of the disordered alloys of Fe-C at various temperatures. Young's modulus of elasticity for FeC-0.2% and FeC-0.4% were calculated and compared with the experimental data reported by [62] and the SMM (Statistical Moment Method) calculated data reported by Tinh et al. [63]. The comparison is depicted using the graph in Figure 3. As seen in the figure, Young's modulus decreased with increasing temperatures and closely follows the experimental curve in the case of both disordered alloys. For both alloys, our potential was able to reproduce Young's modulus for a wide range of temperatures more accurately (i.e., closely following the experimental result) than the previously reported SMM result. Detailed results on potential verification can be found in our previous work [28].

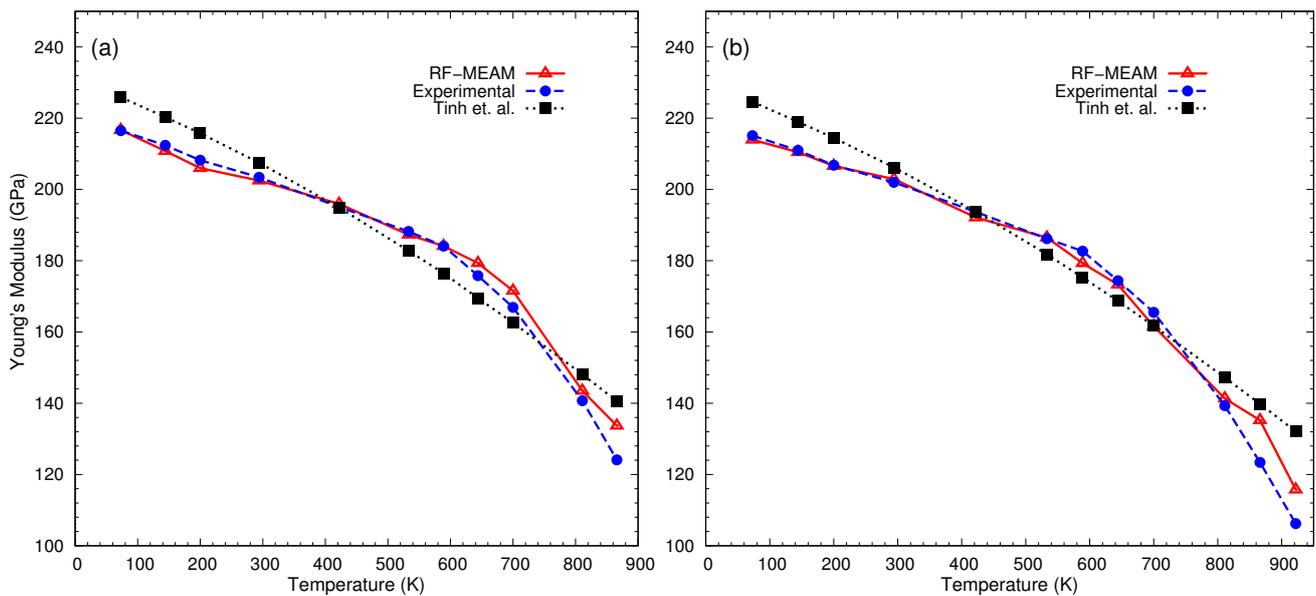


Figure 3. Young's moduli of interstitial Fe-C alloys: (a) FeC-0.2% and (b) FeC-0.4% at various temperatures. For comparison, experimental [62] and SMM-calculated data by Tinh et al. [63] are also included.

3.2. Data Analysis and Feature Selection

From the MD calculation result, we extracted temperature, carbon composition, and other structural parameters, including lattice constants, base structure, interstitial site placements, alloy type (i.e., ordered or disordered), and the calculated elastic moduli. To gain insights into potential input features for our ML study, we plotted the elastic moduli against these possible input features.

As shown in Figure 4, bulk and rigidity modulus tends to decrease as the temperature increases for all ordered structures (i.e., B1, B3, and Cementite). This figure also sheds light on the influence of structural features on elastic properties. However, it is not clear whether this effect is due to the lattice type or other structural parameters like lattice constants. Hence, both the lattice type and the lattice constants are taken as input features for our prediction model.

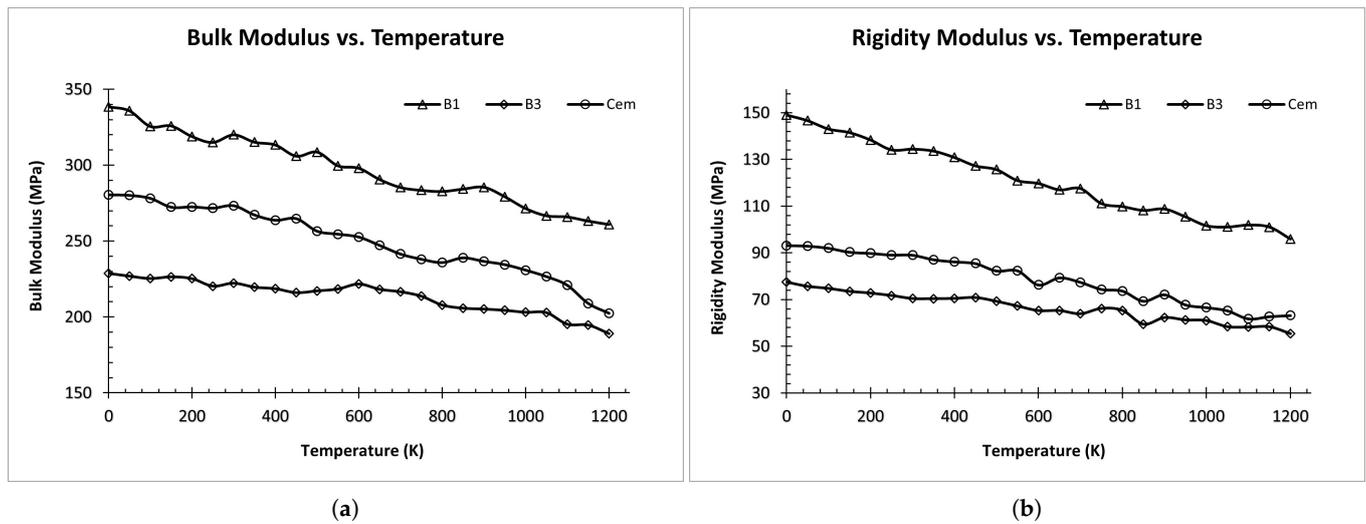


Figure 4. Elastic Modulus vs. Temperature for ordered structures: (a) Bulk Modulus vs. Temperature. (b) Rigidity Modulus vs. Temperature.

Figure 5 shows the relationship between the elastic moduli and temperatures for the fcc-FeC (1% C) structure with carbon occupying octahedral and tetrahedral interstitial sites. Notably, we discerned that these properties exhibited small variations depending on the position of the carbon interstitial site. When carbon was situated at octahedral sites, both the bulk and rigidity modulus exhibited higher values than when carbon occupied tetrahedral sites. Consequently, we included the carbon interstitial site locations in our pool of potential input features.

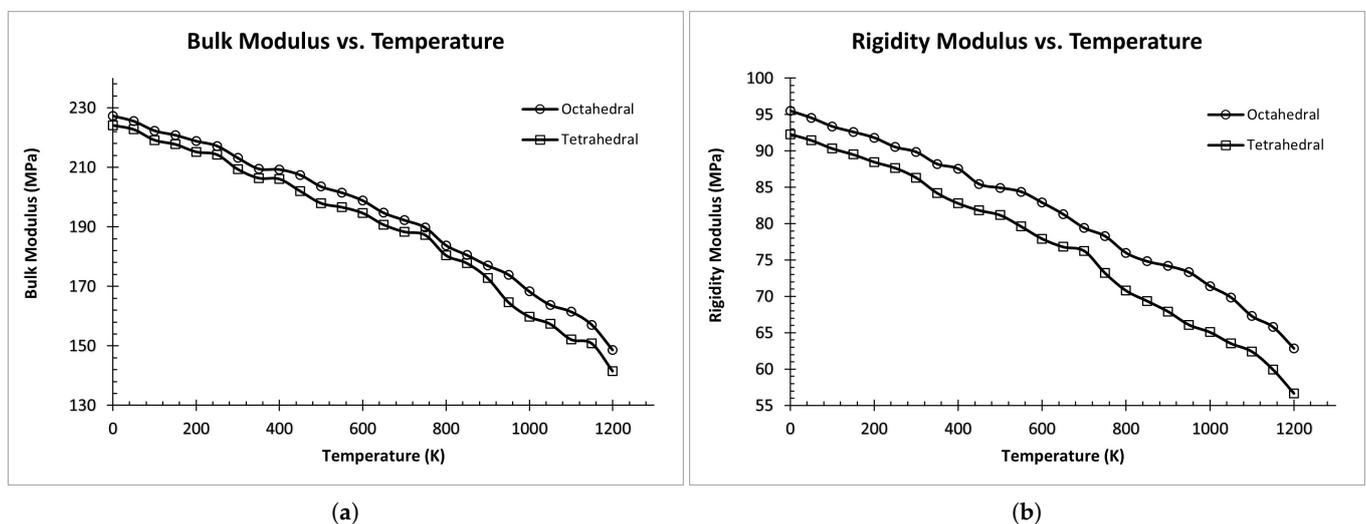


Figure 5. Elastic Modulus vs. Temperature for Fe-C (1% C) disordered alloy with C at different interstitial sites. (a) Bulk Modulus vs. Temperature, (b) Rigidity Modulus vs. Temperature.

Figure 6 represents the relationship between the elastic moduli and carbon composition at different temperatures for the fcc-FeC alloy. In this figure, we observed that increasing carbon composition led to an increase in the bulk modulus, up to approximately 15%, before exhibiting a gradual decrease at higher concentrations. Simultaneously, the rigidity modulus decreased with both an increase in carbon concentration and elevated temperatures. Young's modulus is not presented in any of the three cases (Figures 4–6), as it exhibited a consistent and parallel pattern to the shear modulus, owing to the greater weightage of shear modulus compared to bulk modulus in Equation (7).

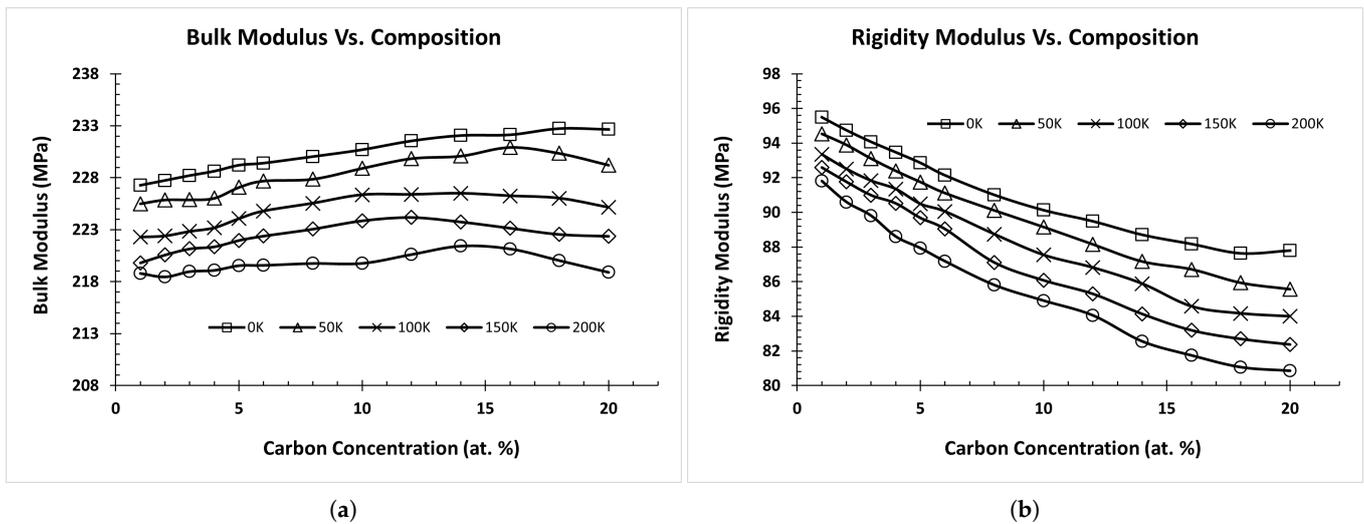


Figure 6. Elastic Moduli vs. Composition for various disordered Fe-C alloys with octahedral interstitial C atoms. (a) Bulk Modulus vs. Composition, (b) Rigidity Modulus vs. Composition.

Based on the data analysis above, we selected eight key input features to capture the essential variables that influence the properties of our materials. These features include temperature (Temp), carbon composition (CC), alloy type (Alloy), i.e., order or disorder alloy, interstitial sites of carbon (Int. Sites), lattice structure type (Str), and lattice constants: L_a , L_b , and L_c .

Temperature, which is a critical parameter, ranges from 0 to 1200 K, with increments of 50 K, providing a comprehensive range of temperature conditions to evaluate material behavior. Carbon composition is another vital input, varying from 1 to 20% for disordered alloys and up to 50% for ordered alloys, encompassing a wide spectrum of compositions relevant to our study. Another important input feature is lattice type which covers four major lattice types: FCC, BCC, orthorhombic, and simple cubic, ensuring that we account for a diverse set of structural configurations. Additionally, we have included interstitial sites of carbon as a feature, with variations in octahedral, tetrahedral, planar, and basal interstitial sites. This inclusion allows us to explore the impact of carbon placement within the lattice on material characteristics.

Next, feature analysis was performed using the ML algorithm, and RF regressor, where feature importance scores were calculated for each input feature as shown in Figure 7. Upon calculating the feature importance of the eight features against the target output, it was observed that temperature (Temp) was the most important feature, with the feature important score of 0.52. Following Temp, lattice constants (L_c , L_a , and L_b), type of alloy (Alloy), carbon concentration (CC), and interstitial sites (IntSites) were also found to be significant. Lattice structure (Str) had a feature importance score of less than 1%, which shows that it has less impact on the target output.

The interatomic potential used to perform the MD calculation was developed on the data of just two structures of Fe-C, i.e., B1 and Cementite. So, the result obtained from MD calculation using such potential might not be sensitive to the lattice structure type. This might have resulted in the lower feature importance of lattice structure (Str) while predicting the elastic moduli.

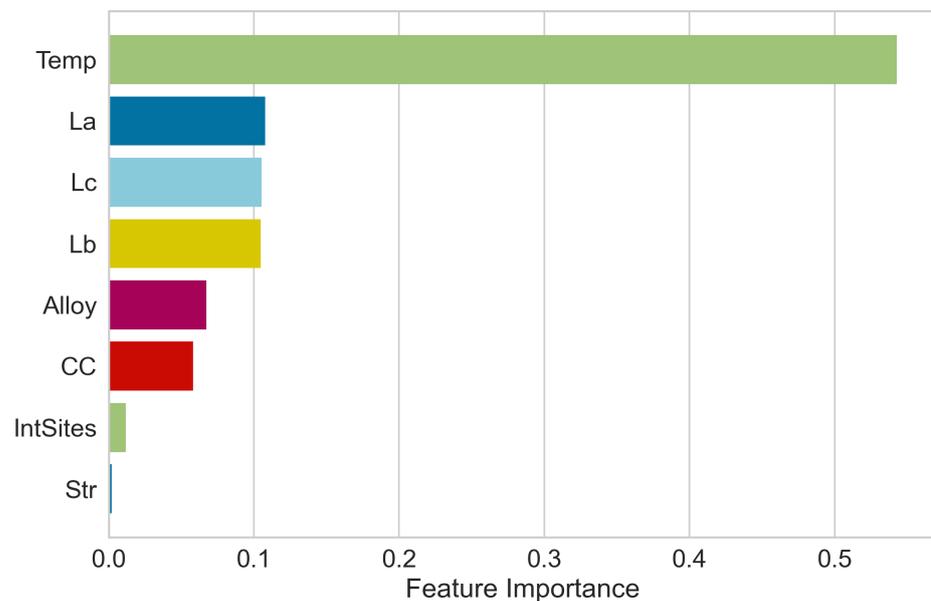


Figure 7. Feature Importance of eight input features for elastic properties prediction.

3.3. Feature Elimination and Hyperparameter Tuning

Before making decisions about eliminating features based on their importance scores, we conducted another test to understand how the removal of less important features would impact the predictive accuracy of our ML models. Hence, we prepared two different datasets, one with seven features and another with all (eight features). As discussed in the “Data and Method” section, various algorithms were used on these datasets to make the prediction. Furthermore, two different approaches were tried: (a) training two independent models to predict B and G, and (b) training a multi-variate model to predict both B and G simultaneously. For all cases, the data were split into 80:20 training and test sets. The training set was used to train the models, and the test set was used to assess the models’ accuracy.

In parallel, we also performed hyperparameter tuning, which involves exploring various hyperparameters for each algorithm and selecting the most optimum solution based on the cross-validation (CV) score. Five fold CV was applied on the training set. First, individual algorithms were tuned for each dataset and method. Subsequently, for SL, the hyperparameter-tuned individual algorithms were taken as the base predictors, and at the same time, all of these were tried one by one for meta learners while tuning the hyperparameters of the meta estimator. Among all the individual algorithms tried, MLP gave the least mean square error CV score, thus was finalized as the meta estimator.

Table 2 presents the results of our experiments, showcasing the best CV scores, measured in terms of mean squared error (MSE), for each of the cases. The table shows a consistent pattern across all algorithms and models: the dataset with seven parameters consistently yields the same or superior performance compared to the dataset with eight parameters. It can be deduced that the eighth parameter, i.e., lattice structure (L) produces a neutral or negative impact on the prediction. Hence, the feature was removed from the dataset, and the results hereafter use the seven-feature dataset.

We also did a comparative analysis of two distinct approaches: (a) employing individual models for the separate prediction of the elastic constants B and G, and (b) employing a multi-variate model to predict these properties simultaneously. Figure 8 provides a comprehensive overview of this comparison, utilizing six individual models and one ensemble algorithm, focusing on the MSE as the evaluation metric. These MSE values, unlike the one presented in Table 2, are calculated using the test dataset, which has not been used during the model training. Notably, the results shown in the figure clearly favor the

independent models, which consistently had smaller MSEs across all algorithms. This highlights the effectiveness of the independent model when contrasted with the multi-variate model approach.

Table 2. Best CV scores across different ML models and feature sets during hyperparameter tuning.

Algorithms	Independent Models				Multi-Variate Model	
	B_7	B_8	G_7	G_8	S_7	S_8
RF	8.311	8.377	3.878	3.885	7.046	7.050
KNN	11.051	11.051	3.447	3.447	7.250	7.250
SVM	8.427	8.427	1.646	1.646	10.175	10.175
MLP	5.470	5.485	1.514	1.616	3.520	3.596
GPR	5.036	5.036	1.453	1.453	3.244	3.244
XGBoost	4.868	4.868	1.517	1.517	3.259	3.259
SL	3.876	4.695	1.360	1.441	3.440	3.696

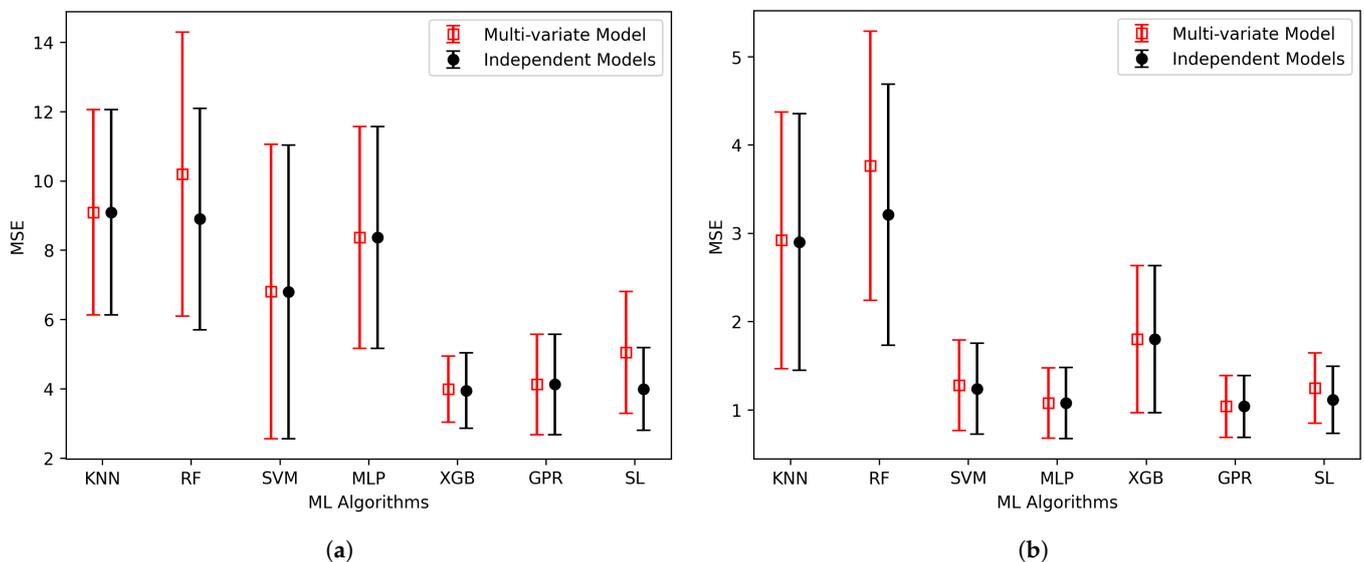


Figure 8. Assessment of elastic moduli, (a) Bulk modulus and (b) Rigidity modulus, calculated by independent and simultaneous models using various algorithms.

We then also calculated Young's modulus of elasticity (E) utilizing Equation (7), and subjected it to a similar comparative analysis, as illustrated in Figure 9. Once again, the independent model demonstrated superior performance, showcasing each algorithm's lowest MSE when predicting Young's modulus.

Furthermore, when examining the performance of each algorithm in Figures 8 and 9 concerning the prediction of B , G , and E , some noteworthy trends emerged. In the case of B prediction, XGBoost, GPR, and SL consistently outperformed others, displaying both lower prediction errors and reduced error variation. Similarly, when estimating G , MLP, GPR, and SL demonstrated superior predictive accuracy, with GPR standing out as the top performer.

Likewise, the prediction of E , MLP, GPR, and SL proved to be the most effective, with GPR being the most accurate result. Notably, in both Figures 8 and 9, we can also see that GPR and SL have been consistently effective in predicting all elastic moduli, exhibiting a narrow range of error variation and reinforcing their reliability in our analysis. The MSE errors and 95% confidence interval (CI) shown in Figures 8 and 9 are also detailed in Table 3.

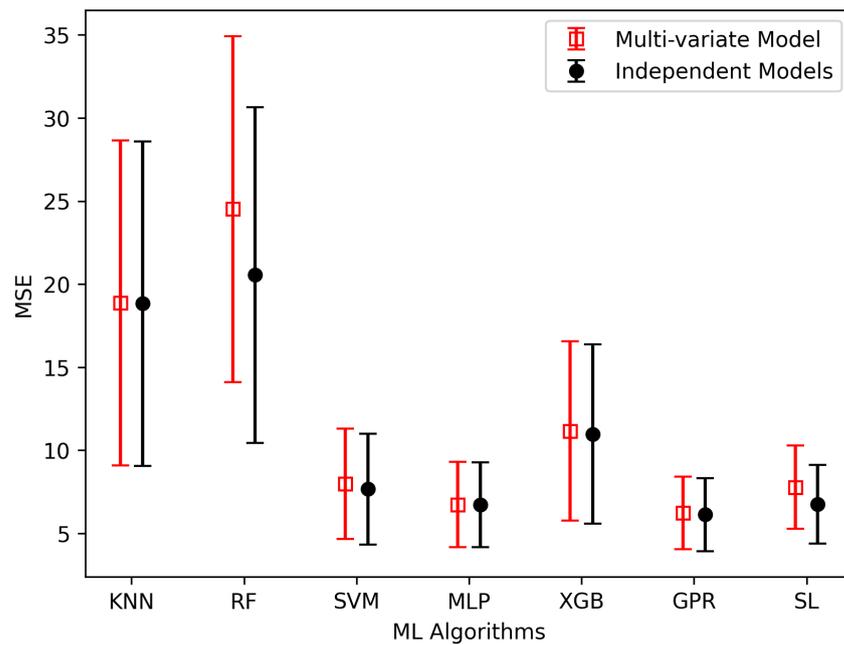


Figure 9. Assessment of Youngs modulus calculated by independent and simultaneous models using various algorithms.

Table 3. Comparison for MSE calculated on test set for multi-variate and independent models using different ML algorithms.

Algorithms	B		G		E	
	Multi-Variate	Independent	Multi-Variate	Independent	Multi-Variate	Independent
KNN	9.09 ± 2.97	9.09 ± 2.97	2.92 ± 1.46	2.90 ± 1.46	18.86 ± 9.81	18.83 ± 9.81
RF	10.20 ± 4.11	8.90 ± 3.21	3.76 ± 1.53	3.21 ± 1.48	24.52 ± 10.45	20.55 ± 10.14
SVM	6.81 ± 4.26	6.80 ± 4.25	1.28 ± 0.52	1.24 ± 0.51	7.96 ± 3.35	7.66 ± 3.33
MLP	8.36 ± 3.21	8.36 ± 3.21	1.08 ± 0.40	1.08 ± 0.40	6.72 ± 2.59	6.71 ± 2.56
XGB	3.99 ± 0.96	3.95 ± 1.09	1.80 ± 0.83	1.80 ± 0.83	11.15 ± 5.42	10.97 ± 5.41
GPR	4.13 ± 1.45	4.13 ± 1.45	1.04 ± 0.35	1.04 ± 0.35	6.32 ± 2.21	6.22 ± 2.20
SL	5.05 ± 1.76	4.00 ± 1.20	1.25 ± 0.40	1.11 ± 0.38	7.77 ± 2.53	6.75 ± 2.38

Now, focusing on the independent model approach for predicting B and G, we proceeded with the predictions and subsequently calculated E using Equation (7). For the ease of comparing different algorithms, the calculated Young’s modulus was compared with its actual counterpart since it will give the aggregate performance of two independent models. Figure 10 provides a visual representation of these predicted versus actual value comparisons, complete with their respective R² scores. Among the seven algorithms used, four algorithms, namely (a) SVM, (b) MLP, (c) GPR, and (d) SL, showed exceptional performance with R² scores of 0.996, 0.997, 0.997, and 0.997, respectively. It is particularly worth highlighting that, once again, GPR and SL demonstrated superior performance, reaffirming their effectiveness in this elastic properties prediction model.

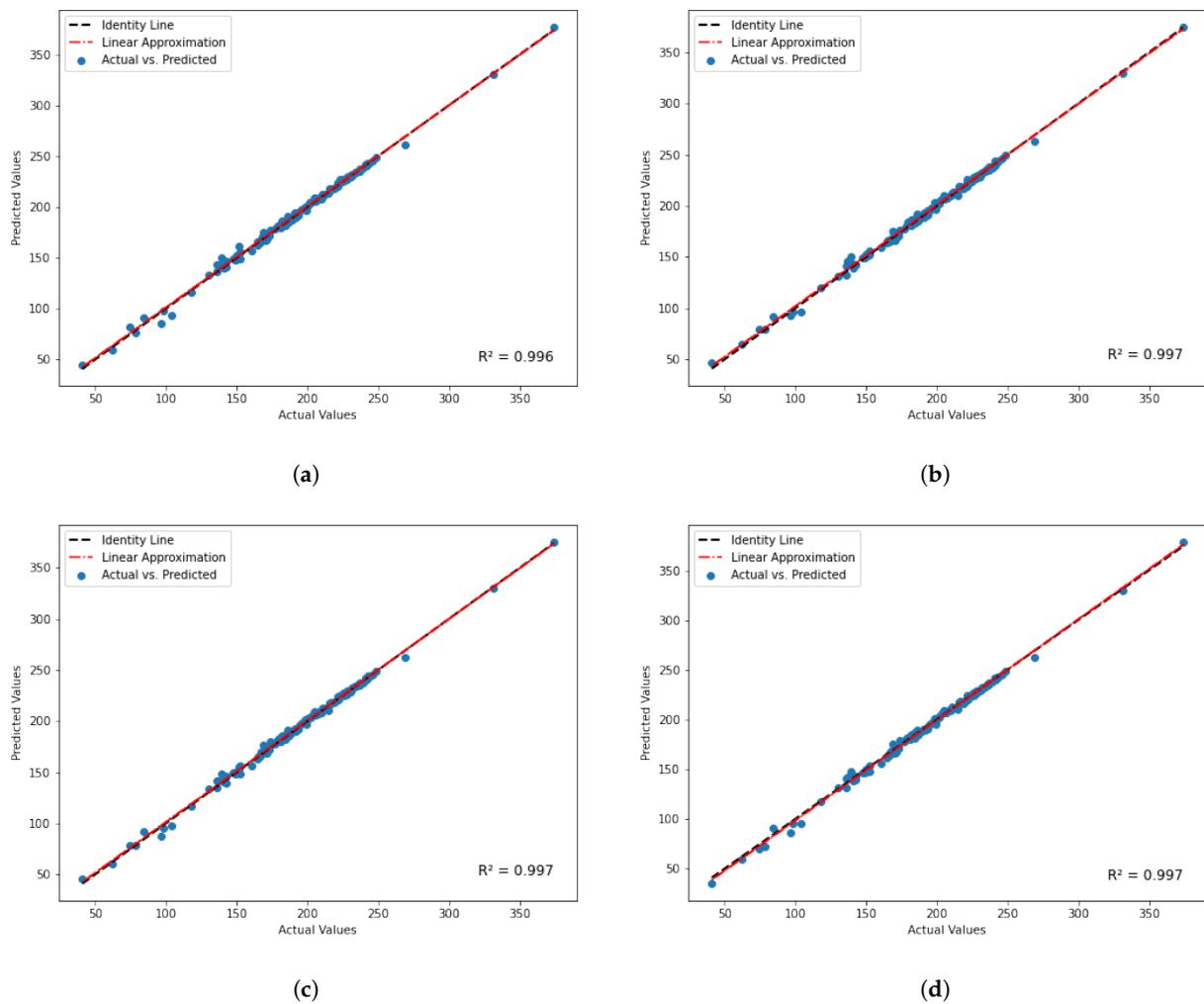


Figure 10. Predicted value versus true value using simultaneous B and G predicting model for top four algorithms: (a) SVM, (b) MLP (c) GPR, and (d) SL.

4. Conclusions

In conclusion, the RF-MEAM potential, developed by fitting DFT-generated forces, energies, and stress tensors, was used to generate thousands of data points for ML, focused on predicting the elastic moduli (i.e., bulk modulus (B), rigidity modulus (G), and Young's modulus (E)) for various Fe-C alloys. We explored a broad spectrum of ordered and disordered alloys, each characterized by varying carbon compositions, and across a range of elevated temperatures. In our quest to identify key input features for our ML models, we conducted meticulous data analysis with an in-depth understanding of the input-output relationships. This process led us to select eight input features: Temp, CC, IntSites, Alloy, Str, L_a , L_b , and L_c . RF regressor was used to assess the importance of these features. Temp was the most influential feature, closely followed by lattice constants and alloy type. Feature importance followed by the CV accuracy evaluation during hyperparameter optimization suggested removing the least important feature, lattice structure type (Str).

Furthermore, we investigated two distinct model training approaches: independent models for B and G prediction and a multi-variate for concurrent B and G prediction. The independent models approach consistently yielded smaller MSEs across all algorithms, compellingly endorsing its effectiveness. The predictive capabilities of the independent model were further affirmed through the accurate prediction of Young's modulus.

A thorough analysis of six individual algorithms and one ensemble algorithm was conducted, with the primary focus on comparing their performance using the R^2 metric. Notably, the independent model approach for B and G prediction showcased remarkable results, particularly for GPR and SL, both achieving remarkable R^2 scores of 0.997 when predicting E. This outcome further solidifies their standing as robust and dependable predictors in the context of our study.

Our study has thus contributed valuable insights into the ML prediction of elastic properties for Fe-C alloys, emphasizing the significance of temperature and a select set of input features. It underscores the effectiveness of the independent model approach and the robust performance of GPR and SL, paving the way for enhanced material design and optimization in materials science.

Author Contributions: Conceptualization, S.R. (Sandesh Risal), N.S. and W.Z.; methodology, S.R. (Sandesh Risal), N.S. and W.Z.; software, S.R. (Sandesh Risal), N.S. and Y.Y.; validation, S.R. (Sandesh Risal); formal analysis, S.R. (Sandesh Risal); investigation, S.R. (Sandesh Risal), N.S. and W.Z.; resources, Y.Y. and W.Z.; data curation, S.R. (Sandesh Risal), S.R. (Samprash Risal) and N.S.; writing—original draft preparation, S.R. (Sandesh Risal), N.S. and W.Z.; writing—review and editing, S.R. (Sandesh Risal), L.S., N.S., S.R. (Samprash Risal) and W.Z.; visualization, S.R. (Sandesh Risal); supervision, N.S., Y.Y., L.S. and W.Z.; project administration, W.Z.; funding acquisition, W.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This paper is partially supported by funding from a computational grant from University of Houston (UH) HPE Data Science Institute, National Science Foundation grants #1855147, #1950036, #2141674, #2220683, National Academy of Sciences, Engineering, and Medicine grant #200011064, U.S. Department of Agriculture grants #13111855, #13424031, and two internal grants from UH Advanced Manufacturing Institute. Their support is much appreciated.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding authors.

Acknowledgments: This work used the Opuntia and the Sabine Cluster Computer of Research Computing Data Core (RCDC) at the University of Houston (UH).

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

References

1. Chen, L.; Chen, L.; Kalinin, S.; Klimeck, G.; Kumar, S.; Neugebauer, J.; Terasaki, I. Design and discovery of materials guided by theory and computation. *NPJ Comput. Mater.* **2015**, *1*, 1–2. [[CrossRef](#)]
2. Singh, V.; Patra, S.; Murugan, N.; Toncu, D.; Tiwari, A. Recent trends in computational tools and data-driven modeling for advanced materials. *Mater. Adv.* **2022**, *3*, 4069–4087. [[CrossRef](#)]
3. Vandenhoute, S.; Rogge, S.; Van Speybroeck, V. Large-scale molecular dynamics simulations reveal new insights into the phase transition mechanisms in MIL-53 (Al). *Front. Chem.* **2021**, *9*, 718920. [[CrossRef](#)]
4. Allen, M.; Tildesley, D. *Computer Simulation of Liquids*; Oxford University Press: Oxford, UK, 2017.
5. Tuckerman, M. *Statistical Mechanics: Theory and Molecular Simulation*; Oxford University Press: Oxford, UK, 2023.
6. Fujisaki, H.; Moritsugu, K.; Matsunaga, Y.; Morishita, T.; Maragliano, L. Extended phase-space methods for enhanced sampling in molecular simulations: A review. *Front. Bioeng. Biotechnol.* **2015**, *3*, 125. [[CrossRef](#)] [[PubMed](#)]
7. Bernardi, R.; Melo, M.; Schulten, K. Enhanced sampling techniques in molecular dynamics simulations of biological systems. *Biochim. Biophys. Acta* **2015**, *1850*, 872. [[CrossRef](#)] [[PubMed](#)]
8. Mueller, T.; Kusne, A.; Ramprasad, R. Machine learning in materials science: Recent progress and emerging applications. *Rev. Comput. Chem.* **2016**, *29*, 186–273.
9. Nosengo, N. Can artificial intelligence create the next wonder material? *Nature* **2016**, *533*, 22–25. [[CrossRef](#)] [[PubMed](#)]
10. Butler, K.; Davies, D.; Cartwright, H.; Isayev, O.; Walsh, A. Machine learning for molecular and materials science. *Nature* **2018**, *559*, 547–555. [[CrossRef](#)] [[PubMed](#)]
11. Schmidt, J.; Marques, M.; Botti, S.; Marques, M. Recent advances and applications of machine learning in solid-state materials science. *Npj Comput. Mater.* **2019**, *5*, 83. [[CrossRef](#)]

12. Risal, S.; Zhu, W.; Guillen, P.; Sun, L. Improving phase prediction accuracy for high entropy alloys with Machine learning. *Comput. Mater. Sci.* **2021**, *192*, 110389. Available online: <https://www.sciencedirect.com/science/article/pii/S0927025621001142> (accessed on 23 October 2023). [[CrossRef](#)]
13. Hu, M.; Tan, Q.; Knibbe, R.; Wang, S.; Li, X.; Wu, T.; Jarin, S.; Zhang, M. Prediction of mechanical properties of wrought aluminium alloys using feature engineering assisted machine learning approach. *Metall. Mater. Trans. A* **2021**, *52*, 2873–2884. [[CrossRef](#)]
14. Deng, Z.; Yin, H.; Jiang, X.; Zhang, C.; Zhang, G.; Xu, B.; Yang, G.; Zhang, T.; Wu, M.; Qu, X. Machine-learning-assisted prediction of the mechanical properties of Cu-Al alloy. *Int. J. Miner. Metall. Mater.* **2020**, *27*, 362–373. [[CrossRef](#)]
15. Devi, M.; Prakash, C.; Chinnannavar, R.; Joshi, V.; Palada, R.; Dixit, R. An informatic approach to predict the mechanical properties of aluminum alloys using machine learning techniques. In Proceedings of the 2020 International Conference on Smart Electronics and Communication (ICOSEC), Trichy, India, 10–12 September 2020; pp. 536–541.
16. Xu, X.; Wang, L.; Zhu, G.; Zeng, X. Predicting tensile properties of AZ31 magnesium alloys by machine learning. *JOM* **2020**, *72*, 3935–3942. [[CrossRef](#)]
17. Choyi, Y. Computational Alloy Design and Discovery Using Machine Learning. 2020. Available online: <https://www.semanticscholar.org/paper/Computational-Alloy-Design-and-Discovery-Using-Choyi/c2648b0cc575cf288e778857f6dbcad357b46e9a> (accessed on 21 December 2023).
18. Ling, J.; Antonio, E.; Bajaj, S.; Paradiso, S.; Hutchinson, M.; Meredig, B.; Gibbons, B. Machine learning for alloy composition and process optimization. *Turbo Expo Power Land Sea Air* **2018**, *51128*, V006T24A005.
19. Lu, J.; Chen, Y.; Xu, M. Prediction of mechanical properties of Mg-rare earth alloys by machine learning. *Mater. Res. Express* **2022**, *9*, 106519. [[CrossRef](#)]
20. Tan, R.; Li, Z.; Zhao, S.; Birbilis, N. A primitive machine learning tool for the mechanical property prediction of multiple principal element alloys. *arXiv* **2023**, arXiv:2308.07649.
21. Zhang, Q.; Mahfouf, M. Fuzzy predictive modelling using hierarchical clustering and multi-objective optimisation for mechanical properties of alloy steels. *IFAC Proc. Vol.* **2007**, *40*, 427–432. [[CrossRef](#)]
22. Lee, J.; Park, C.; Do Lee, B.; Park, J.; Goo, N.; Sohn, K. A machine-learning-based alloy design platform that enables both forward and inverse predictions for thermo-mechanically controlled processed (TMCP) steel alloys. *Sci. Rep.* **2021**, *11*, 11012. [[CrossRef](#)] [[PubMed](#)]
23. Shen, C.; Wang, C.; Wei, X.; Li, Y.; Zwaag, S.; Xu, W. Physical metallurgy-guided machine learning and artificial intelligent design of ultrahigh-strength stainless steel. *Acta Mater.* **2019**, *179*, 201–214. [[CrossRef](#)]
24. Mahfouf, M. Optimal design of alloy steels using genetic algorithms. In *Advances in Computational Intelligence and Learning: Methods and Applications*; Springer Science+Business Media: New York, NY, USA, 2002; pp. 425–436.
25. Gaffour, S.; Mahfouf, M.; Yang, Y. ‘Symbiotic’ data-driven modelling for the accurate prediction of mechanical properties of alloy steels. In Proceedings of the 2010 5th IEEE International Conference Intelligent Systems, London, UK, 7–9 July 2010; pp. 31–36.
26. Dutta, T.; Dey, S.; Datta, S.; Das, D. Designing dual-phase steels with improved performance using ANN and GA in tandem. *Comput. Mater. Sci.* **2019**, *157*, 6–16. Available online: <https://www.sciencedirect.com/science/article/pii/S092702561830692X> (accessed on 25 September 2023). [[CrossRef](#)]
27. Pattanayak, S.; Dey, S.; Chatterjee, S.; Chowdhury, S.; Datta, S. Computational intelligence based designing of microalloyed pipeline steel. *Comput. Mater. Sci.* **2015**, *104*, 60–68. [[CrossRef](#)]
28. Risal, S.; Singh, N.; Duff, A.; Yao, Y.; Sun, L.; Risal, S.; Zhu, W. Development of the RF-MEAM Interatomic Potential for the Fe-C System to Study the Temperature-Dependent Elastic Properties. *Materials* **2023**, *16*, 3779. [[CrossRef](#)]
29. Kresse, G.; Hafner, J. Ab initio molecular dynamics for liquid metals. *Phys. Rev. B* **1993**, *47*, 558–561. [[CrossRef](#)]
30. Kresse, G.; Joubert, D. From ultrasoft pseudopotentials to the projector augmented-wave method. *Phys. Rev. B* **1999**, *59*, 1758–1775. [[CrossRef](#)]
31. Perdew, J.; Ruzsinszky, A.; Csonka, G.; Vydrov, O.; Scuseria, G.; Constantin, L.; Zhou, X.; Burke, K. Restoring the Density-Gradient Expansion for Exchange in Solids and Surfaces. *Phys. Rev. Lett.* **2008**, *100*, 136406. [[CrossRef](#)]
32. Timonova, M.; Thijsse, B. Optimizing the MEAM potential for silicon. *Model. Simul. Mater. Sci. Eng.* **2010**, *19*, 015003. [[CrossRef](#)]
33. Duff, A. MEAMfit: A reference-free modified embedded atom method (RF-MEAM) energy and force-fitting code. *Comput. Phys. Commun.* **2016**, *203*, 354–355. [[CrossRef](#)]
34. Thompson, A.; Aktulga, H.; Berger, R.; Bolintineanu, D.; Brown, W.; Crozier, P.; Veld, P.; Kohlmeyer, A.; Moore, S.; Nguyen, T.; et al. LAMMPS—a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comput. Phys. Commun.* **2022**, *271*, 108171. [[CrossRef](#)]
35. Breiman, L. Random forests. *Mach. Learn.* **2001**, *45*, 5–32. [[CrossRef](#)]
36. Hastie, T.; Tibshirani, R.; Friedman, J.; Hastie, T.; Tibshirani, R.; Friedman, J. Random forests. In *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 587–604.
37. Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J.; Sheridan, R.; Feuston, B. Random forest: A classification and regression tool for compound classification and QSAR modeling. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 1947–1958. [[CrossRef](#)] [[PubMed](#)]
38. Chen, T.; He, T.; Benesty, M.; Khotilovich, V.; Tang, Y.; Cho, H.; Chen, K.; Mitchell, R.; Cano, I.; Zhou, T.; et al. R Package Version 0.4-2; Xgboost: Extreme gradient boosting; 2015; Volumer 1, pp. 1–4.
39. Chen, T.; Guestrin, C. Xgboost: A scalable tree boosting system. In Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, 13–17 August 2016; pp. 785–794.

40. Guillen-Rondon, P.; Robinson, M.; Torres, C.; Pereya, E. Support vector machine application for multiphase flow pattern prediction. *arXiv* **2018**, arXiv:1806.05054.
41. Steinwart, I.; Christmann, A. *Support Vector Machines*; Springer Science & Business Media: New York, NY, USA, 2008.
42. Statnikov, A. *A Gentle Introduction to Support Vector Machines in Biomedicine: Theory and Methods*; World Scientific: London, UK, 2011.
43. Kramer, O. K-nearest neighbors. In *Dimensionality Reduction with Unsupervised Nearest Neighbors*; Springer: Berlin/Heidelberg, Germany, 2013; pp. 13–23.
44. Peterson, L. K-nearest neighbor. *Scholarpedia* **2009**, *4*, 1883. [[CrossRef](#)]
45. Kumar, T. Solution of linear and non linear regression problem by K Nearest Neighbour approach: By using three sigma rule. In Proceedings of the 2015 IEEE International Conference on Computational Intelligence & Communication Technology, Ghaziabad, India, 13–14 February 2015; pp. 197–201.
46. Skryjomski, P.; Krawczyk, B.; Cano, A. Speeding up k-Nearest Neighbors classifier for large-scale multi-label learning on GPUs. *Neurocomputing* **2019**, *354*, 10–19. Available online: <https://www.sciencedirect.com/science/article/pii/S0925231219304588> (accessed on 20 September 2023). [[CrossRef](#)]
47. Wu, Y.; Feng, J. Development and application of artificial neural network. *Wirel. Pers. Commun.* **2018**, *102*, 1645–1656. [[CrossRef](#)]
48. Popescu, M.; Balas, V.; Perescu-Popescu, L.; Mastorakis, N. Multilayer perceptron and neural networks. *WSEAS Trans. Circuits Syst.* **2009**, *8*, 579–588.
49. Sheela, K.; Deepa, S. Review on methods to fix number of hidden neurons in neural networks. *Math. Probl. Eng.* **2013**, *2013*, 425740. [[CrossRef](#)]
50. Wilson, A.; Knowles, D.; Ghahramani, Z. Gaussian process regression networks. *arXiv* **2011**, arXiv:1110.4411.
51. Deringer, V.; Bartók, A.; Bernstein, N.; Wilkins, D.; Ceriotti, M.; Csányi, G. Gaussian process regression for materials and molecules. *Chem. Rev.* **2021**, *121*, 10073–10141. [[CrossRef](#)] [[PubMed](#)]
52. McBride, K.; Sundmacher, K. Overview of surrogate modeling in chemical process engineering. *Chem. Ing. Tech.* **2019**, *91*, 228–239. [[CrossRef](#)]
53. Gelfand, A.; Schliep, E. Spatial statistics and Gaussian processes: A beautiful marriage. *Spat. Stat.* **2016**, *18*, 86–104. [[CrossRef](#)]
54. Gonzalez, J.; Lezmi, E.; Roncalli, T.; Xu, J. Financial applications of Gaussian processes and Bayesian optimization. *arXiv* **2019**, arXiv:1903.04841.
55. Deisenroth, M.; Fox, D.; Rasmussen, C. Gaussian processes for data-efficient learning in robotics and control. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *37*, 408–423. [[CrossRef](#)] [[PubMed](#)]
56. Laan, M.; Polley, E.; Hubbard, A. Super learner. *Stat. Appl. Genet. Mol. Biol.* **2007**, *6*. [[CrossRef](#)]
57. Golmakani, M.; Polley, E. Super learner for survival data prediction. *Int. J. Biostat.* **2020**, *16*, 20190065. [[CrossRef](#)] [[PubMed](#)]
58. Lee, S.; Nguyen, N.; Karamanli, A.; Lee, J.; Vo, T. Super learner machine-learning algorithms for compressive strength prediction of high performance concrete. *Struct. Concr.* **2023**, *24*, 2208–2228. [[CrossRef](#)]
59. Henriksson, K.; Sandberg, N.; Wallenius, J. Carbides in stainless steels: Results from ab initio investigations. *Appl. Phys. Lett.* **2008**, *93*, 191912. [[CrossRef](#)]
60. Chihi, T.; Bouhemadou, A.; Reffas, M.; Khenata, R.; Ghebouli, M.; Ghebouli, B.; Louail, L. Structural, elastic and thermodynamic properties of iron carbide Fe₇C₃ phases: An ab initio study. *Chin. J. Phys.* **2017**, *55*, 977–988. [[CrossRef](#)]
61. Liyanage, L.; Kim, S.; Houze, J.; Kim, S.; Tschopp, M.; Baskes, M.; Horstemeyer, M. Structural, elastic, and thermal properties of cementite (Fe₃C) calculated using a modified embedded atom method. *Phys. Rev. B* **2014**, *89*, 094102. [[CrossRef](#)]
62. EngineeringToolBox Metals and Alloys-Young's Modulus of Elasticity. 2004. Available online: https://www.engineeringtoolbox.com/young-modulus-d_773.html (accessed on 12 October 2023).
63. Tinh, B.; Hoc, N.; Vinh, D.; Cuong, T.; Hien, N. Thermodynamic and Elastic Properties of Interstitial Alloy FeC with BCC Structure at Zero Pressure. *Adv. Mater. Sci. Eng.* **2018**, *2018*, 5251741. [[CrossRef](#)]

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