EAI Endorsed Transactions
on AI and Robotics

Research Article EAI.EU

A Machine Learning Based Investigative Analysis for Predicting the Critical Temperature of Superconductors

Fatin Abrar Shams[1], Rashed Hasan Ratul[1], Ahnaf Islam Naf[1], Syed Shaek Hossain Samir[1], Samiur Rashid Abir[1], Fahim Faisal[1]*, Mirza Muntasir Nishat[1], Md. Ashraful Hoque[1]

[1]Department of Electrical and Electronic Engineering, Islamic University of Technology (IUT), Dhaka, Bangladesh

Abstract

INTRODUCTION: Ever since the initial discovery of superconductivity, the fundamental concept and the complex relationship between critical temperature and superconductive materials have been subject to extensive investigation. However, identifying superconductors at normal temperatures remains a significant challenge, and there are still significant gaps in our understanding of this unique phenomenon, particularly regarding the fundamental criteria used to estimate critical temperature.
OBJECTIVES: To address this knowledge gap, a plethora of machine learning (ML) techniques have been developed in this work to model critical temperatures, given the inherent difficulty in predicting them using traditional methods.
METHODS: Additionally, the limitations of the standard empirical formula in determining the temperature range require the development of more advanced and viable methods. This article presents an investigative analysis of the performance achieved by different supervised machine learning algorithms when used with three different feature selection techniques.
RESULTS: The stacking model used in this work is found to be the best performer among all the algorithms tested, as reflected by the Root Mean Squared Error (RMSE) of 9.68, R2 score of 0.922, Mean Absolute Error (MAE) score of 5.383, and Mean Absolute Percentage Error (MAPE) score of 4.575.
CONCLUSION: Therefore, it is observed that ML algorithms can contribute significantly in the domain of predictive analysis of modeling critical temperatures in superconductors and can assist in developing a robust computer-aided system to aid the education personnel and research scientists to further assess the performance of the ML models.

Received on 25 July 2023; accepted on 27 October 2023; published on 30 October 2023

Keywords: Superconductor, Critical Temperature, Machine Learning, Stacking Ensemble Method.

doi:10.4108/airo.3622

1. Introduction

The importance of superconductivity as a distinct and captivating phenomena is paramount, and it has been the subject of extensive research for almost a century [1]. The critical temperature, denoted as T_c, refers to the temperature at which certain metals, such as indium, mercury, lead, tin, and niobium, cease to exhibit electrical resistance [2]. The current analytical models used to predict the superconducting threshold temperature T_c based on available data are not sufficiently accurate due to their oversimplification of a complex and nonlinear problem. Superconductors possess two distinct physical characteristics: absolute electrical conductivity and full diamagnetism. These properties are of utmost importance in various industrial sectors such as mechanical engineering, wireless communication, underwater sensors, superconducting electric motors, non-destructive testing of components, instrumentation, and power systems [3–5]. Moreover, the exceptional responsiveness of superconductive materials to magnetic fields can be

*Corresponding author. Email: fahimf@iut-dhaka.edu

utilized to accurately ascertain the location of underwater sensor nodes by utilizing optoacoustic signals [6–8]. The traditional method of trial-and-error experimentation for discovering new superconductors is time-consuming and requires the use of very high pressure and low temperatures. Computational approaches based on density functional theory (DFT) are often costly and require a significant amount of effort [9]. This has led researchers to explore other methods, such as machine learning [10–19], to gain a deeper understanding of the relationships between superconductivity and a material's chemistry and structure. Machine learning algorithms [20–29] offer a unique opportunity to extract patterns and relationships from large datasets that traditional methods may miss [30–41]. It is interesting to note that the critical temperatures ($T_c s$) of most recently found superconducting materials, which have a stronger interaction between two-dimensional electrons and phonons, do not align with Allen and Dynes' formulation [42]. Hence, it seems that employing machine-learning methodology is a more effective strategy for highlighting the specification that surpasses the constraints of traditional empirical methodologies in describing the characteristics of superconductors, particularly in relation to critical temperature. This research introduces an integrated machine learning approach to predict the T_c information of superconducting substances, which can accelerate the identification of potential high-T_c superconductors.
Moreover, accurately estimating the critical temperature (T_c) is a crucial aspect of the complex and resource-intensive process involved in synthesizing superconductors [43]. The viability of the synthesis process depends significantly on the accuracy of the T_c estimation. Therefore, the findings of this study could have potential implications for decision-making in the synthesis of superconductors [44]. The proposed approximation technique offers a reliable way for researchers to estimate the T_c of newly discovered superconducting materials. By accurately estimating T_c, researchers can efficiently identify materials with desirable superconducting properties, expediting the development of new superconductors.

In this study, five supervised machine learning models, along with a stacking ensemble model and a voting ensemble model, have been tested. The purpose of this study was to analyze the performance of these models under three different feature selection methods that were tested, i.e., f-classif, f-regression, and mutual info regression. Again, to analyze the difference in performance when these feature selection methods are used, all features were used to analyze the performance of these algorithms. Furthermore, hyper-parameter optimization was done using RandomizedSearchCV. The difference before and after this optimization is compared. The novelty of this study can be considered as follows:

- Firstly, understanding the influence of feature selection methods on the overall performance of the ML methods applied in this work to estimate the critical temperature of the superconductors.
- Secondly, evaluating the performance of the stacking and voting ensemble methods applied in this work to investigate whether it adds any additional benefits for using these models over the previously used models.
- Finally, inspecting the performance gain achieved after optimizing the hyper-parameters.

This paper is organized in the following manner. The literature and works related to this work are described in section 2. Section 3 contains a detailed description of the methodology including the data description, preprocessing, feature scaling, Hyper-parameter tuning, the overall workflow along with the approaches adopted in this work and the model descriptions. In section 4 the evaluation metrics used in this work are described. Finally, in Sections 5 and 6, the results and the conclusion have been included.

2. Literature Review

Machine learning has been a crucial aspect in numerous significant domains for researchers [45–64]. In recent years, there has been a surge of interest in using machine learning algorithms to predict the critical temperature of superconductors. To predict T_c from diverse sets of input characteristics, several models such as Random Forest, Support Vector Machines, and Artificial Neural Networks have been used [65]. Some research has also concentrated on finding important factors that contribute the most to T_c prediction. Despite substantial advances in this field, precisely predicting T_c for complex superconductors with multiple elements and disorder in their crystal structure remains a difficulty. The objective of this literature review is to provide a comprehensive summary of the current advancements in utilizing machine learning techniques to estimate the critical temperature (T_c) of superconductors. The review will examine the various machine learning models that have been developed for this objective and their respective merits and limitations. Hamidieh created a model in [66], that employs a combination of linear regression, gradient boosting, and neural networks to make predictions. They also used feature engineering to extract relevant information from the chemical formulas and improve the accuracy of the predictions Their statistical model performs fairly well with an RMSE of 9.5K. (Despite achieving superior RMSE and

R2 scores, their approach cannot identify the features that are more crucial for predicting T_c, whereas this paper's approach demonstrates the ability to do so. The study in [67] suggests a technique that describes materials using atomic vectors and predicts T_c using a hybrid neural network (HNN) model that combines a convolutional neural network (CNN) and a long short-term memory neural network (LSTM). The LSTM recovers the long-dependence feature interactions between atoms, while the HNN model employs CNN to extract the short-dependence feature relationships. This deep learning-based approach performs pretty well with an R2 score of 0.899 and MAE 5.023K; However, their approach only yields a substandard RMSE of 83.565. Consequently, the current study achieves superior results in terms of both RMSE and R2 compared to their findings. A novel method was developed by Paulino et al. [68] by fusing the MARS approximation and the whale optimization algorithm (WOA). This may be an appealing methodology that had not previously been explored. In addition to that Ridge, Lasso, and Elastic-net regression was used for comparison purposes. The results show that all four machine learning techniques are capable of predicting T_c with reasonable accuracy but this hybrid WOA/MARS-based model outperforms the rest three models with an RMSE value of 15.14, R2 score of 0.80, and MAE of 10.75. However, in the aspects of RMSE, R2, and MAE, the results of the present study outperform their findings. Two popular regression methods, linear and simple linear regression models, were utilized in the research of Babu et al. [69] to compare various performance metrics. Their linear regression model yields improved results, featuring an RMSE of 17.68, MAE of 13.42, and R2 of 0.7396. Once again, the results from the present study stand superior, highlighting better RMSE, R2, and MAE metrics. In the study of Mohammad N. Haque et al. [70], they introduced a new model for multivariate regression that involves the iterative fitting of a continued fraction alongside additive spline models. To assess its efficacy, they compared it with different established techniques, including AdaBoost, Kernel Ridge, Linear Regression, etc. They evaluated the performance of these methods in predicting the critical temperature of superconductors based on their physical-chemical properties, which is a crucial problem in the field. They achieved an RMSE of 10.989, which the current research managed to outperform. The numerical characterization of the material is a necessary first step in such methods, after which various machine learning algorithms are employed to test and compare the predictive model. The use of machine learning has the potential to revolutionize the field of superconductivity, providing a more efficient approach to understanding the relationships between a material's chemistry and structure and its superconducting properties.

Nonetheless, these investigations provide valuable knowledge in the realm of materials science and lay the groundwork for future progress in superconductivity research employing machine learning methods. In this paper, we introduce a stacked integrated machine learning approach that merges various streamlined machine learning algorithms while enhancing the meta-model's hyperparameters. Consequently, the proposed approach provides a more comprehensive and precise depiction of machine learning model performance in material discovery applications with a simplified approach. The study exhibited a high coefficient of determination (R2 score = 0.922) in the predictive framework, which is on par with or superior to certain prior artificial intelligence techniques[1].

3. Methodology

3.1. Data Description

This dataset emphasizes a cutting-edge machine learning technique to extract complex superconductive material properties for critical temperature (T_c) prediction. The information regarding superconductors was compiled from the Superconducting Material Database (SuperCon), which is preserved by the Japanese National Institute of Materials Science (NIMS) [72]. 21,263 superconductors are employed after some data preprocessing. However, this database was not open-accessed at the time of this research. That is why, the same dataset from the UCI Machine Learning Repository [73] was used in this paper. It includes the critical temperature and 81 features derived from 21,263 superconductors.

3.2. Feature Scaling

Feature scaling refers to the process of normalizing or standardizing the data. This is done to ensure that all features contribute equally to the model and to prevent features with larger values from dominating the model, as machine learning models can make the underlying assumption that larger values have greater importance compared to lower values [74]. Furthermore, in this study min-max scaling was implemented to normalize the data, which can be expressed in the following mathematical expression:

$$\text{Min-Max Scaling} = \frac{X - X_{\min}}{X_{\max} - X_{\min}} \tag{1}$$

[1]Part of this research was presented at the 7th IEEE International Conference on Sustainable Innovation (ICOSI 2023). This paper first appeared on ArXiv in August 2023 [71].

Where:

X : The original value of the feature,
$X_{\min}$: The minimum value of the feature,
$X_{\max}$: The maximum value of the feature.

3.3. Data Preprocessing

To compare the machine learning models used in this work, three feature selection methods (f-regression, f-classif, mutual info regression) were used. The top 50 features were chosen out of the original 81 attributes. This enhances performance by focusing on the most relevant attributes and reducing dimensionality. The selection of 50 features was made to strike a balance between retaining enough information for accurate modeling and reducing the dimensionality of the dataset. Choosing a smaller subset of features helps to eliminate noise and irrelevant information that could potentially hinder the model's performance [75]. Table 1 shows the overall progression of dataset preprocessing.

3.4. Hyperparameter Tuning

In machine learning, the task of determining the ideal collection of hyperparameters for a learning algorithm is referred to as hyperparameter optimization or tuning. A parameter whose value is utilized to guide the learning process is termed a hyperparameter. For this process, RandomizedSearchCV was used to find out the best hyperparameter. Cross-validated search across parameter settings is used to optimize the estimator's parameters, which are then used to implement these methods.

3.5. Workflow

In this study, the effectiveness of three feature selection techniques, namely f-regression, f-classif, and mutual info regression, was evaluated both individually and in comparison to a baseline where all the features were used. These specific feature selection methods have not been previously utilized in the preprocessing of this dataset.

The study is divided into four stages. The first stage involves obtaining baseline results without applying any feature selection techniques. This step provides a reference point for evaluating the performance of the subsequent feature selection methods. Before proceeding with the feature selection stages, the data is preprocessed. Min-max scaling is employed as the chosen data preprocessing technique. This method scales the data to a fixed range, typically between 0 and 1. By applying min-max scaling, the data is normalized and brought within a standardized range, facilitating ease of further analysis.

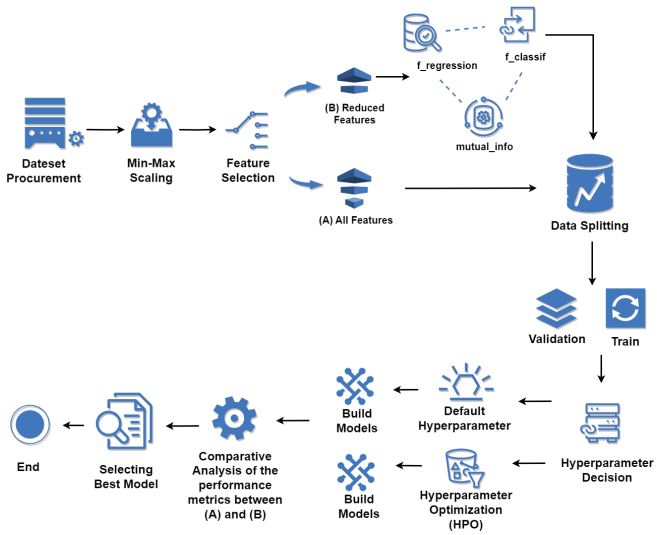

Figure 1. Overall Workflow Diagram.

In the next three stages, each of the feature selection methods is applied separately. Feature selection is a process of reducing the number of input variables to include only those that are most beneficial to the model [76]. It can enhance model efficiency and reduce computational costs. By selecting the most relevant features, the model can focus on the attributes that have the greatest impact on accurate predictions. The f-regression and f-classif methods are used to calculate the correlation between each predictor and the target variable. These methods capture linear interactions between predictors and the target. On the other hand, mutual info regression is capable of capturing various types of relationships, including linear, quadratic, and exponential. Compared to f-regression and f-classif, mutual info regression is generally considered more reliable and adaptable, particularly when the relationships between predictors and the target are unclear. Finally, to evaluate the models, four performance metrics (RMSE, R2 score, MAE, and MAPE) were employed. The objective was to assess the impact of hyperparameter optimization on model performance when utilizing reduced feature sets. Fig. 1 represents the complete workflow diagram.

Overall, the workflow of this research article involves preprocessing the data using min-max scaling, followed by evaluating the performance of three feature selection techniques individually and in comparison to the baseline. This approach showcases the potential of these feature selection methods and their impact on the predictive accuracy of the model.

3.6. Approach I

To compare model performance with fewer features, a baseline was established using all 81 features. The

Table 1. Dataset Preprocessing

Feature Selection Method	Scaling Method	Feature Count	Status of Hyperparameter
None	Min-Max Scaler	81	Not optimized
None	Min-Max Scaler	81	Optimized
f-regression	Min-Max Scaler	50	Not optimized
f-regression	Min-Max Scaler	50	Optimized
mutual-info-regression	Min-Max Scaler	50	Not optimized
mutual-info-regression	Min-Max Scaler	50	Optimized
f-classif	Min-Max Scaler	50	Not optimized
f-classif	Min-Max Scaler	50	Optimized

study employed five standalone machine learning models, two stacked regression models, and one voting regression model. Initially, default hyperparameters were used for training the models. Subsequently, hyperparameter optimization was conducted using the Random-Search CV algorithm to find the best combinations [77]. The resulting optimal hyperparameter configurations for each model can be found in Table 2.

3.7. Approach II

The f-classif feature selection method employed in this study is a univariate technique. It utilizes univariate statistical tests to select the most relevant features, making it a preprocessing step for estimators [78]. From this method, the top 50 features were chosen. The same set of five standalone models from the baseline section were utilized, along with stacking and voting models. Hyperparameter optimization was performed using RandomSearch CV, resulting in the best hyperparameter combinations presented in Table 3.

3.8. Approach III

The f-regression method is a recommended feature selection criterion for identifying potentially predictive features, regardless of their association's sign with the target variable. This method provides p-values as a measure of feature significance [79]. In this section, the top 50 features were selected using the f-regression method. After this, five standalone models and the stacking and voting models were employed. The hyperparameters of these algorithms were optimized using RandomSearch CV. The optimized hyperparameters were also applied to build the stacking and voting models. Table 4 summarizes the best hyperparameters for each model while implementing this feature selection technique.

3.9. Approach IV

RandomSearchal information is a non-negative measure of the interdependence between two random variables. RandomSearchal information quantifies variable dependence, with zero indicating independence and higher values indicating stronger dependence. This section utilized RandomSearchal information regression. This method utilizes non-parametric algorithms based on k-nearest neighbor distances to estimate entropy [80]. Optimal hyperparameters were determined through the RandomizedSearchCV method after initially using default hyperparameters. Table 5 summarizes the best hyperparameters for each model.

3.10. Model Description

Stacking Model. Stacking is a method that can be used to ensemble several different classification or regression models [81]. Ensemble models can be created in a variety of ways; however, bagging and boosting are the most common approaches. The variance can be reduced using the bagging technique by averaging the results of numerous similar models with high volatility. Boosting is the process of building numerous incremental models to reduce bias while maintaining a low variance. When used for a problem involving classification or regression, stacking has the advantage of combining the most successful features of multiple different efficient models. This, in turn, produces predictions that are superior to those produced by any one individual model in the ensemble. A random division into J sections of the same size is performed using this method on the dataset. One set is utilized for the testing phase of the j-fold cross-validation, while the remaining sets are put to use in the training phase. Because of these training testing pair subsets, it can obtain the predictions of several learning models, which are subsequently utilized as the meta-data to construct the meta-model. The ultimate forecast is determined by the meta-model, which is also referred to as the winner-takes-all technique [82]. In this suggested model, five algorithms as estimators were used, with the Random Forest regressor with the default hyper-parameters, serving as the final estimator. Five algorithms were used as the base estimators, which were Random Forest, KNN, SVR, Ridge, and Lasso as shown in Fig. 2. All these base estimators were once used with the default hyper-parameters and once with the hyper-parameters found

Table 2. Hyperparameter Optimization On All The 81 Features

Algorithm	Hyperparameters	Test Values	Best Vales
Ridge Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.001, 0.0001, 0.01, 0.1, 0.00001	0.00001
	solver	auto, svd, cholesky, lsqr, sparse_cg, sag, saga	sparse_cg
Lasso Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.001, 0.0001, 0.01, 0.1, 0.00001, 0.000001, 0.0000001	0.0000001
KNN	n_neighbors	2, 5, 10, 25, 50	2
	leaf_size	10, 20, 30, 60, 90, 105, 120, 150	20
	algorithm	auto, ball_tree, kd_tree, brute	brute
	p	1, 2, 3, 5, 10, 20, 40, 80, 100, 200	2
SVR	epsilon	0.01, 0.1, 1, 10, 100	1
	C	0.5, 1, 5, 10, 100, 0.05	100
	cache_size	0.2, 2, 20, 200, 2000	2000
	coef0	0.01, 0.1, 0, 1, 10	10
	degree	1, 2, 3, 4, 5	1
MLP	activation	logistic, relu	relu
	learning_rate_init	0.01, 0.1, 0.001	0.1
	hidden_layer_sizes	(55, 52, 78, 30), (56, 32, 25), (57, 40, 52, 75, 60)	(56, 32, 25)
RFR	n_estimators	20, 40, 60, 80, 100, 120	120
	min_samples_split	2, 4, 8, 10	2
	max_depth	5, 10, 15, 20	20

Table 3. Hyperparameter optimization on 50 features selected by f-classif

Algorithm	Hyperparameters	Test Values	Best Values
Ridge Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.001, 0.0001, 0.01, 0.1, 0.00001	0.0001
	solver	'auto', 'svd', 'cholesky', 'lsqr', 'sparse_cg', 'sag', 'saga'	svd
Lasso Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.001, 0.0001, 0.01, 0.1, 0.00001, 0.000001, 0.0000001	0.0000001
KNN	n_neighbors	2, 5, 10, 25, 50	5
	leaf_size	10, 20, 30, 60, 90, 105, 120, 150	150
	algorithm	auto, ball_tree, kd_tree, brute	kd_tree
	p	1, 2, 3, 5, 10, 20, 40, 80, 100, 200	1
SVR	epsilon	0.01, 0.1, 1, 10, 100	0.1
	C	0.5, 1, 5, 10, 100, 0.05	100
	cache_size	0.2, 2, 20, 200, 2000	20
	coef0	0.01, 0.1, 0, 1, 10	0.1
	degree	1, 2, 3, 4, 5	3
MLP	activation	logistic, relu	relu
	learning_rate_init	0.01, 0.1, 0.001	0.001
	hidden_layer_sizes	(55, 52, 78, 30), (56, 32, 25), (57, 40, 52, 75, 60)	(57, 40, 52, 75, 60)
RFR	n_estimators	20, 40, 60, 80, 100, 120	100
	min_samples_split	2, 4, 8, 10	2
	max_depth	5, 10, 15, 20	20

Table 4. Hyperparameter optimization On 50 Features f-regression

Algorithm	Hyperparameters	Test Values	Best Vales
Ridge Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.001, 0.0001, 0.01, 0.1, 0.00001	0.1
	solver	auto, svd, cholesky, lsqr, sparse_cg, sag, saga	svd
Lasso Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.001, 0.0001, 0.01, 0.1, 0.00001, 0.000001, 0.0000001	0.000001
KNN	n_neighbors	2, 5, 10, 25, 50	2
	leaf_size	10, 20, 30, 60, 90, 105, 120, 150	20
	algorithm	auto, ball_tree, kd_tree, brute	ball_tree
	p	1, 2, 3, 5, 10, 20, 40, 80, 100, 200	1
SVR	epsilon	0.01, 0.1, 1, 10, 100	1
	C	0.5, 1, 5, 10, 100, 0.05	100
	cache_size	0.2, 2, 20, 200, 2000	20
	coef0	0.01, 0.1, 0, 1, 10	0.1
	degree	1, 2, 3, 4, 5	1
MLP	activation	logistic, relu	relu
	learning_rate_init	0.01, 0.1, 0.001	0.001
	hidden_layer_sizes	(55, 52, 78, 30), (56, 32, 25), (57, 40, 52, 75, 60)	(55, 52, 78, 30)
RFR	n_estimators	20, 40, 60, 80, 100, 120	120
	min_samples_split	2, 4, 8, 10	4
	max_depth	5, 10, 15, 20	20

Table 5. Hyperparameter optimization on 50 features selected by mutual-info

Algorithm	Hyperparameters	Test Values	Best Values
Ridge Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.001, 0.0001, 0.01, 0.1, 0.00001	0.0001
	solver	'auto', 'svd', 'cholesky', 'lsqr', 'sparse_cg', 'sag', 'saga'	auto
Lasso Regressor	alpha	0.1, 1, 10, 0.001, 100	0.001
	tol	0.0010, 0.0001, 0.01, 0.1, 0.00001, 0.000001, 0.0000001	0.01
KNN	n_neighbors	2, 5, 10, 25, 50	2
	leaf_size	10, 20, 30, 60, 90, 105, 120, 150	105
	algorithm	'auto', 'ball_tree', 'kd_tree', 'brute'	brute
	p	1, 2, 3, 5, 10, 20, 40, 80, 100, 200	20
SVR	epsilon	0.01, 0.1, 1, 10, 100	1
	C	0.5, 1, 5, 10, 100, 0.05	100
	cache_size	0.2, 2, 20, 200, 2000	200
	coef0	0.01, 0.1, 0, 1, 10	0.01
	degree	1, 2, 3, 4, 5	2
MLP	activation	logistic, relu	relu
	learning_rate_init	0.01, 0.1, 0.001	0.01
	hidden_layer_sizes	(55,52,78,30), (56,32,25), (57,40,52,75,60)	(56, 32, 25)
RFR	n_estimators	20, 40, 60, 80, 100, 120	100
	min_samples_split	2, 4, 8, 10	4
	max_depth	5, 10, 15, 20	20

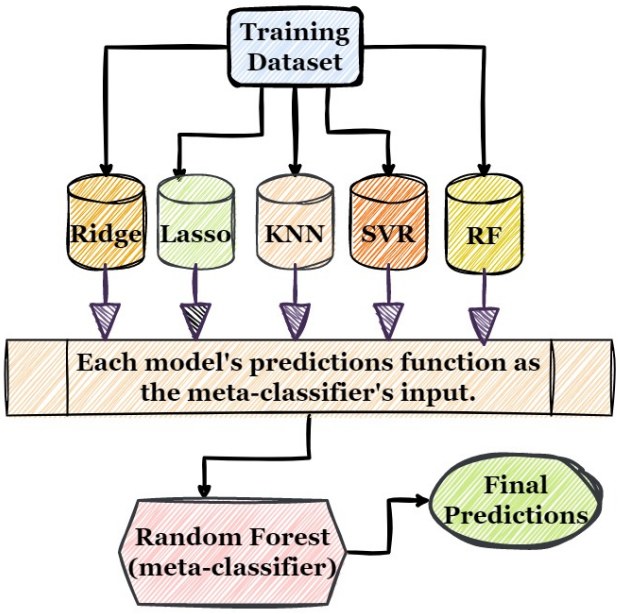

Figure 2. Stacking model.

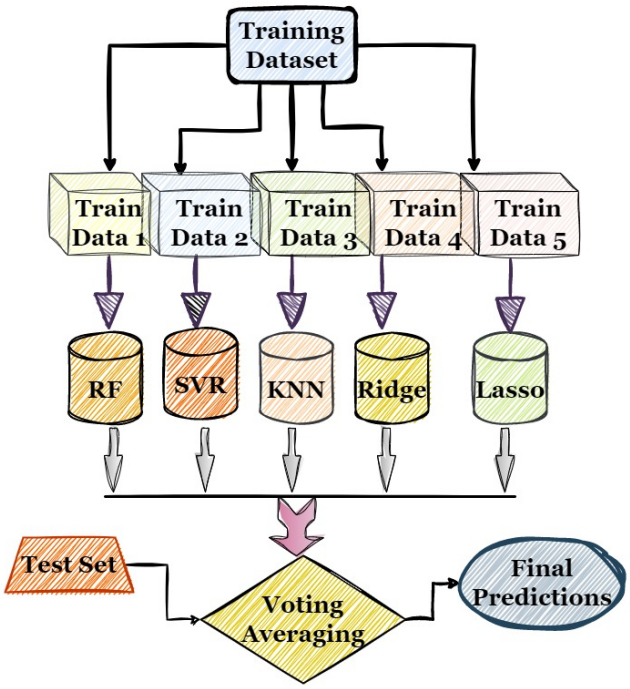

Figure 3. Voting model.

while optimizing each of those standalone models with RandomSearch CV.

Voting Model. In Voting ensemble models, there are several models of the various machine learning algorithms that are present. These models are fed the entire dataset, and after being trained on the data, each algorithm will make a prediction [83]. After all of the models have made their predictions for the sample data, the average of the predictions of all the models will be taken which will reflect the final prediction of the voting ensemble model. The voting ensemble method used in this work consists of five standalone models. These models are the Random Forest, KNN, SVR, Ridge, and Lasso as shown in Fig. 3. All of the models were first used with the default hyperparameters. To optimize the voting ensemble model further, the best hyperparameters that were found using RandomizedSearchCV were used for each of the standalone models while using them as the estimators. Along with this another voting ensemble model was created where all the default hyper-parameters of the models were used.

4. Evaluation Metrics

4.1. RMSE

The standard deviation of the error in the prediction can be found from the Root Mean Square Error (RMSE). Residuals of these prediction errors are the measure of how far from the regression line data points are. The RMSE measures how spread these residual values are. The underlying assumption when presenting the RMSE is that the errors are unbiased and follow a normal distribution. The RMSE can be defined by the following equation [84]:

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

Where:

N : The number of data points,
y_i : The actual values,
$\hat{y}_i$: The predicted values.

4.2. MAE

Mean Absolute Error (MAE) is a measure of the average magnitude of the errors in a set of predictions, without taking into account their direction. It is the average absolute difference between the predicted and actual values and is used to evaluate the performance of a regression model. MAE can be represented by the following equation [85]:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \tag{3}$$

Where:

N : The number of data points,
y_i : The actual values,
$\hat{y}_i$: The predicted values.

MAE presents itself to be the most natural measure of average error magnitude, and (unlike RMSE) it is an unambiguous measure of average error magnitude.

4.3. MAPE

Mean Absolute Percentage Error (MAPE) is the most common error analysis technique used for forecasting. It measures accuracy as a percentage. MAPE can be represented by the following mathematical equation [86]:

$$\text{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \times 100\% \tag{4}$$

Where:

N : The number of data points,
y_i : The actual values,
$\hat{y}_i$: The predicted values.

MAPE is generally used when the quantity to be predicted remains much higher than zero.

4.4. R2 Score

A measure of how well a linear regression model fits the data is called the R-squared. This statistic expresses, as a percentage, the percentage of the variation in the dependent variable that can be attributed, as a whole, to the effects of the independent variables [87]. On a scale that ranges from 0 to 100 percent, the coefficient of determination, or R-squared, provides an important measurement of the strength of the relationship between the model and the dependent accurately by the various algorithms will be utilized as the model's final prediction.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \tag{5}$$

Where:

N : The number of data points,
y_i : The actual values,
$\hat{y}_i$: The predicted values,
$\bar{y}$: The mean of the actual values.

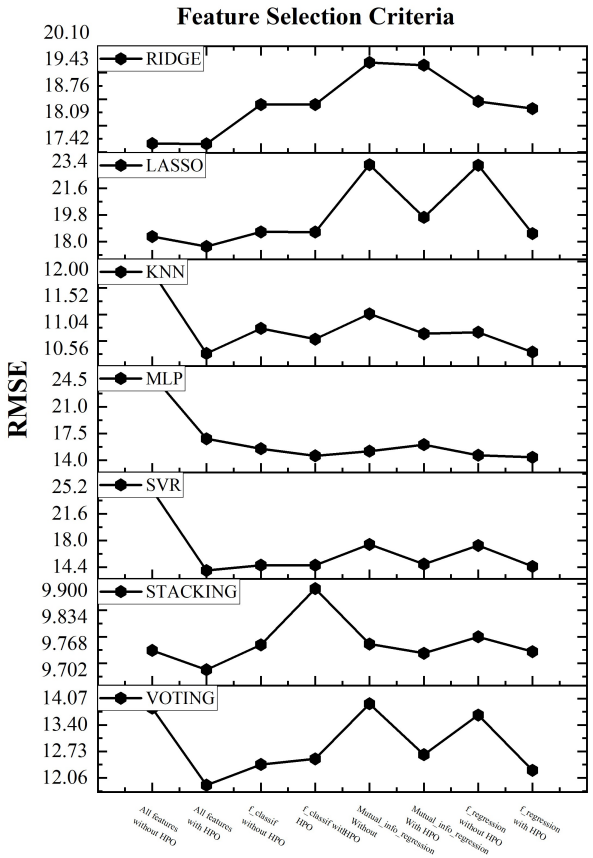

Figure 4. RMSE for different ML models in terms of feature selection methods.

5. Results

In this section, the testing set created in each of the approaches mentioned above is tested with the evaluation metrics that were discussed in section 4. In terms of RMSE scores, the stacking model performed better than other regressor models. Consequently, the stacking model predicted the critical temperature of superconductors with fewer variations in results when tested with multiple feature selection strategies, but all other pertinent ML models displayed fluctuations when various feature selection methods were chosen for RMSE as seen in Fig. 4. Table 6 and Table 7 consequently show the average RMSE for all machine learning models tested for the different feature selection methods used. The prior one depicts the results when the default hyperparameters were used and the latter does the same for the case when the hyperparameters were optimized. From these two tables, it can be seen, that the stacking model achieves the best result of RMSE 9.686 after hyperparameter optimization with all 81 characteristics taken into account.

The stacking model performed the best among all models tested with an average R2 Score that was close to 0.919 across eight different situations considering the result from Table 8 and Table 9, whereas the KNN regressor being the second-best performer had the highest R2 score of 0.9089 when all the features were used with HPO and the lowest R2 score of 0.8808 when all features were used without HPO. This proves that the stacking model has performed similarly in all cases with fewer variations.

The average value of MAE for all the algorithms for each of the feature selection techniques used, with HPO and without HPO are shown in Table 10 and Table 11. The stacking model once again has the lowest MAE score, coming in at 5.383, with scores hovering around 5.4 throughout all eight cases, shown in Tables 10 and 11. The second lowest MAE score of 5.448 was achieved by the KNN model when all features were used with HPO.

The stacking model once again receives the lowest MAPE score, which is 4.575 as shown in Table 12, which is the best result among all 8 cases. It is evident that using the default hyper-parameters for all of the stacking model's algorithms gives the best results for MAPE. The width of each of the separate portions in Fig. 5 indicates the MAPE values. The image shows that, among all the feature selection methods employed here, the MAPE stacking model's breadth is the smallest in comparison to other models.

6. Conclusion

This study attempts to analyze the performance of some supervised machine learning models when utilized with different feature selection methods. Utilizing a stacking ensemble method with hyperparameter optimization outperforms previous research in terms of performance. Evaluation of the model's efficacy using a variety of metrics, such as RMSE, R2 score, MAE, and MAPE, yields insightful information. The results indicate that the incorporation of hyperparameter optimization improves the estimation of critical temperature's precision and dependability. The average RMSE, R2 score, MAE, and MAPE values obtained from models with hyperparameter optimization consistently outperform those without hyperparameter optimization, demonstrating the significance of optimizing model parameters. The study also investigates the effect of feature reduction on model performance. Even after employing feature reduction techniques, the stacking method performs consistently in performance metrics, for all feature selection methods. The results demonstrate the capability of the stacking ensemble method with hyperparameter optimization to improve the dependability of critical temperature estimation under the feature selection methods tested. A deep learning model to predict the critical temperature of superconductors is under development. In the future, we would also like to incorporate the readings of the properties of newly found superconductors in this study to further assess the performance of the ML models.

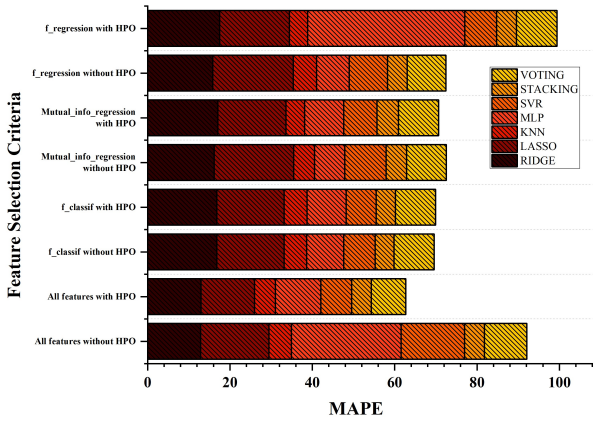

Figure 5. Feature selection criteria vs MAPE.

References

[1] N. Lazarides and G. Tsironis, "Superconducting metamaterials," *Physics Reports*, vol. 752, pp. 1–67, 2018.

[2] P. Richards and M. Tinkham, "Far-infrared energy gap measurements in bulk superconducting in, sn, hg, ta, v, pb, and nb," *Physical Review*, vol. 119, no. 2, p. 575, 1960.

[3] A. Costa, *Theoretical investigations of charge and spin transport through superconducting tunnel junctions*. PhD thesis, 2021.

[4] R. H. Ratul, M. Iqbal, J.-Y. Pan, M. M. Al Deen, M. T. Kawser, and M. M. Billah, "Performance comparison between volte and non-volte voice calls during mobility in commercial deployment: A drive test-based analysis," in *2023 10th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*, pp. 446–452, Sep. 2023.

[5] R. H. Ratul and H.-C. Wang, "Cellular communication network evolution and the reliability of system design from 1g to 6g," in *International Conference on Wireless Intelligent and Distributed Environment for Communication*, pp. 165–187, Springer, 2023.

[6] M. Tasnim and R. H. Ratul, "Optoacoustic signal-based underwater node localization technique: Overcoming gps limitations without auv requirements," in *2023 IEEE Symposium on Wireless Technology & Applications (ISWTA)*, pp. 7–11, 2023.

[7] M. Mahmud, M. Younis, G. Carter, and F.-S. Choa, "Underwater node localization using optoacoustic signals," in *ICC 2022-IEEE International Conference on Communications*, pp. 4444–4449, IEEE, 2022.

[8] M. S. Islam, M. Younis, M. Mahmud, G. Carter, and F.-S. Choa, "A peak detection based ook photoacoustic modulation scheme for air to underwater communication," *Optics Communications*, vol. 529, p. 129078, 2023.

[9] D. Wines, K. Choudhary, A. J. Biacchi, K. F. Garrity, and F. Tavazza, "High-throughput dft-based discovery of next generation two-dimensional (2d) superconductors," *Nano letters*, vol. 23, no. 3, pp. 969–978, 2023.

[10] F. Faisal and M. M. Nishat, "An investigation for enhancing registration performance with brain atlas by novel image inpainting technique using dice and jaccard score on multiple sclerosis (ms) tissue," *Biomedical and Pharmacology Journal*, vol. 12, no. 3, pp. 1249–1262, 2019.

[11] M. M. Nishat, F. Faisal, T. Hasan, M. F. B. Karim, Z. Islam, and M. R. K. Shagor, "An investigative approach to employ support vector classifier as a potential detector of brain cancer from mri dataset," in *2021 International Conference on Electronics, Communications and Information Technology (ICECIT)*, pp. 1–4, IEEE, 2021.

[12] M. A.-A.-R. Asif, M. M. Nishat, F. Faisal, R. R. Dip, M. H. Udoy, M. F. Shikder, and R. Ahsan, "Performance evaluation and comparative analysis of different machine learning algorithms in predicting cardiovascular disease.," *Engineering Letters*, vol. 29, no. 2, 2021.

[13] M. M. Nishat, F. Faisal, R. R. Dip, S. M. Nasrullah, R. Ahsan, F. Shikder, M. A.-A.-R. Asif, and M. A. Hoque, "A comprehensive analysis on detecting chronic kidney disease by employing machine learning algorithms," *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. 7, no. 29, pp. e1–e1, 2021.

[14] M. R. Farazi, F. Faisal, Z. Zaman, and S. Farhan, "Inpainting multiple sclerosis lesions for improving registration performance with brain atlas," in *2016 International Conference on Medical Engineering, Health Informatics and Technology (MediTec)*, pp. 1–6, IEEE, 2016.

[15] M. A.-A.-R. Asif, M. M. Nishat, F. Faisal, M. F. Shikder, M. H. Udoy, R. R. Dip, and R. Ahsan, "Computer aided diagnosis of thyroid disease using machine learning algorithms," in *2020 11th International Conference on Electrical and Computer Engineering (ICECE)*, pp. 222–225, IEEE, 2020.

[16] M. M. Nishat and F. Faisal, "An investigation of spectroscopic characterization on biological tissue," in *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)*, pp. 290–295, IEEE, 2018.

[17] F. Faisal, M. M. Nishat, M. A. Mahbub, M. M. I. Shawon, and M. M.-U.-H. Alvi, "Covid-19 and its impact on school closures: a predictive analysis using machine learning algorithms," in *2021 International Conference on Science & Contemporary Technologies (ICSCT)*, pp. 1–6, IEEE, 2021.

[18] A. A. Rahman, F. Faisal, M. M. Nishat, M. I. Siraji, L. I. Khalid, M. R. H. Khan, and M. T. Reza, "Detection of epileptic seizure from eeg signal data by employing machine learning algorithms with hyperparameter optimization," in *2021 4th International Conference on Bio-Engineering for Smart Technologies (BioSMART)*, pp. 1–4, IEEE, 2021.

[19] M. M. Nishat, T. Hasan, S. M. Nasrullah, F. Faisal, M. A.-A.-R. Asif, and M. A. Hoque, "Detection of parkinson's disease by employing boosting algorithms," in *2021 Joint 10th International Conference on Informatics, Electronics & Vision (ICIEV) and 2021 5th International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*, pp. 1–7, IEEE, 2021.

[20] M. M. Nishat, F. Faisal, T. Hasan, S. M. Nasrullah, A. H. Bristy, M. Minhajul Islam Shawon, and M. Ashraful Hoque, "Detection of autism spectrum disorder by discriminant analysis algorithm," in *Proceedings of the International Conference on Big Data, IoT, and Machine Learning: BIM 2021*, pp. 473–482, Springer, 2022.

[21] A. A. Rahman, M. I. Siraji, L. I. Khalid, F. Faisal, M. M. Nishat, M. R. Islam, *et al.*, "Detection of mental state from eeg signal data: an investigation with machine learning classifiers," in *2022 14th International Conference on Knowledge and Smart Technology (KST)*, pp. 152–156, IEEE, 2022.

[22] M. R. Rahman, S. Tabassum, E. Haque, M. M. Nishat, F. Faisal, and E. Hossain, "Cnn-based deep learning approach for micro-crack detection of solar panels," in *2021 3rd International Conference on Sustainable Technologies for Industry 4.0 (STI)*, pp. 1–6, IEEE, 2021.

[23] M. Muntasir Nishat, F. Faisal, I. Jahan Ratul, A. Al-Monsur, A. M. Ar-Rafi, S. M. Nasrullah, M. T. Reza, and M. R. H. Khan, "A comprehensive investigation of the performances of different machine learning classifiers with smote-enn oversampling technique and hyperparameter optimization for imbalanced heart failure dataset," *Scientific Programming*, vol. 2022, pp. 1–17, 2022.

[24] M. M. Nishat, F. Faisal, M. A. Mahbub, M. H. Mahbub, S. Islam, and M. A. Hoque, "Performance assessment of different machine learning algorithms in predicting diabetes mellitus," *Biosc. Biotech. Res. Comm*, vol. 14, no. 1, pp. 74–82, 2021.

[25] F. Faisal, *Image Inpainting to Improve the Registration Performance of Multiple Sclerosis (MS) Patient Brain with Brain Atlas*. PhD thesis, Department of Electrical and Electronic Engineering, Islamic University of . . . , 2016.

[26] M. M. Nishat, F. Faisal, R. R. Dip, M. F. Shikder, R. Ahsan, M. A.-A.-R. Asif, and M. H. Udoy, "Performance investigation of different boosting algorithms in predicting chronic kidney disease," in *2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI)*, pp. 1–5, IEEE, 2020.

[27] M. Islam, M. Tabassum, M. M. Nishat, F. Faisal, and M. S. Hasan, "Real-time clinical gait analysis and foot anomalies detection using pressure sensors and convolutional neural network," in *2022 7th International Conference on Business and Industrial Research (ICBIR)*, pp. 717–722, IEEE, 2022.

[28] I. J. Ratul, A. Al-Monsur, B. Tabassum, A. M. Ar-Rafi, M. M. Nishat, and F. Faisal, "Early risk prediction of cervical cancer: A machine learning approach," in *2022 19th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 1–4, IEEE,

2022.

[29] A. Al-Monsur, M. R. Kabir, A. M. Ar-Rafi, M. M. Nishat, and F. Faisal, "Covid-ensemblenet: an ensemble based approach for detecting covid-19 by utilising chest x-ray images," in *2022 IEEE World AI IoT Congress (AIIoT)*, pp. 351–356, IEEE, 2022.

[30] I. A. Jahan, I. A. Jamil, M. S. H. Fahim, A. S. Huq, F. Faisal, and M. M. Nishat, "Accident detection and road condition monitoring using blackbox module," in *2022 Thirteenth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 247–252, IEEE, 2022.

[31] A. A. Rahman, M. I. Siraji, L. I. Khalid, F. Faisal, M. M. Nishat, A. Ahmed, and M. A. Al Mamun, "Perceived stress analysis of undergraduate students during covid-19: a machine learning approach," in *2022 IEEE 21st Mediterranean Electrotechnical Conference (MELECON)*, pp. 1129–1134, IEEE, 2022.

[32] F. Faisal, M. M. Nishat, and M. A. M. Oninda, "Spectroscopic characterization of biological tissue using quantitative acoustics technique," in *2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)*, pp. 38–43, IEEE, 2018.

[33] Z. K. Eisham, M. M. Haque, M. S. Rahman, M. M. Nishat, F. Faisal, and M. R. Islam, "Chimp optimization algorithm in multilevel image thresholding and image clustering," *Evolving Systems*, vol. 14, no. 4, pp. 605–648, 2023.

[34] F. Faisal, M. A. Salam, M. B. Habib, M. S. Islam, and M. M. Nishat, "Depth estimation from video using computer vision and machine learning with hyperparameter optimization," in *2022 4th International Conference on Smart Sensors and Application (ICSSA)*, pp. 39–44, IEEE, 2022.

[35] M. M. Nishat, M. A. M. Oninda, F. Faisal, and M. A. Hoque, "Modeling, simulation and performance analysis of sepic converter using hysteresis current control and pi control method," in *2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, pp. 7–12, IEEE, 2018.

[36] M. M. Nishat, F. Faisal, M. Rahman, and M. A. Hoque, "Modeling and design of a fuzzy logic based pid controller for dc motor speed control in different loading condition for enhanced performance," in *2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT)*, pp. 1–6, IEEE, 2019.

[37] A. Al Mehadi, M. M. Nishat, F. Faisal, A. R. H. Bhuiyan, M. Hussain, and M. A. Hoque, "Design, simulation and feasibility analysis of bifacial solar pv system in marine drive road, cox's bazar," in *2021 International Conference on Science & Contemporary Technologies (ICSCT)*, pp. 1–6, IEEE, 2021.

[38] M. M. Nishat, F. Faisal, A. J. Evan, M. M. Rahaman, M. S. Sifat, H. F. Rabbi, *et al.*, "Development of genetic algorithm (ga) based optimized pid controller for stability analysis of dc-dc buck converter," *Journal of Power and Energy Engineering*, vol. 8, no. 09, p. 8, 2020.

[39] A. J. Moshayedi, A. Kolahdooz, A. S. Roy, S. A. L. Rostami, and X. Xie, "Design and promotion of cost-effective iot-based heart rate monitoring," in *International Conference on Cloud Computing, Internet of Things, and Computer Applications (CICA 2022)*, vol. 12303, pp. 405–410, SPIE, 2022.

[40] A. J. Moshayedi, A. S. Roy, L. Liao, and S. Li, "Raspberry pi scada zonal based system for agricultural plant monitoring," in *2019 6th International Conference on Information Science and Control Engineering (ICISCE)*, pp. 427–433, IEEE, 2019.

[41] A. J. Moshayedi, A. S. Roy, S. K. Sambo, Y. Zhong, and L. Liao, "Review on: The service robot mathematical model," *EAI Endorsed Transactions on AI and Robotics*, vol. 1, no. 1, 2022.

[42] X. Gui, *Design, Synthesis and Characterization of New Superconductors*. Louisiana State University and Agricultural & Mechanical College, 2020.

[43] B. Lilia, R. Hennig, P. Hirschfeld, G. Profeta, A. Sanna, E. Zurek, W. E. Pickett, M. Amsler, R. Dias, M. I. Eremets, *et al.*, "The 2021 room-temperature superconductivity roadmap," *Journal of Physics: Condensed Matter*, vol. 34, no. 18, p. 183002, 2022.

[44] M. Yazdani-Asrami, W. Song, A. Morandi, G. De Carne, J. Murta-Pina, A. Pronto, R. Oliveira, F. Grilli, E. Pardo, M. Parizh, *et al.*, "Roadmap on artificial intelligence and big data techniques for superconductivity," *Superconductor Science and Technology*, vol. 36, no. 4, p. 043501, 2023.

[45] F. Faisal, M. M. Nishat, and M. R. Mia, "An investigation on dc motor braking system by implementing electromagnetic relay and timer," in *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pp. 1–6, IEEE, 2019.

[46] M. M. Nishat, M. R. K. Shagor, H. Akter, S. A. Mim, and F. Faisal, "An optimal design of pid controller for dc-dc zeta converter using particle swarm optimization," in *2020 23rd International Conference on Computer and Information Technology (ICCIT)*, pp. 1–6, IEEE, 2020.

[47] M. D. Rahman, F. Faisal, M. M. Nishat, and M. R. K. Shagor, "Design and analysis of passive lc 3 component boost converter," in *2021 IEEE Madras Section Conference (MASCON)*, pp. 1–6, IEEE, 2021.

[48] M. R. K. Shagor, M. M. Nishat, F. Faisal, M. H. Mithun, M. A. Khan, *et al.*, "Firefly algorithm based optimized pid controller for stability analysis of dc-dc sepic converter," in *2021 IEEE 12th Annual Ubiquitous Computing, Electronics & Mobile Communication Conference (UEMCON)*, pp. 0957–0963, IEEE, 2021.

[49] A. Al Mehadi, M. A. Chowdhury, M. M. Nishat, F. Faisal, and M. M. Islam, "Design, simulation and analysis of monofacial solar pv panel based energy system for university residence: a case study," in *IOP Conference Series: Materials Science and Engineering*, vol. 1045, p. 012011, IOP Publishing, 2021.

[50] F. Faisal, M. Rahman, and F. B. Hashem, *Speed control of dc motor using Self-tuned fuzzy PID controller*. PhD thesis, Department of Electrical and Electronic Engineering, Islamic University of . . . , 2015.

[51] S. Hassan, A. A. Hassan, I. Marshad, M. A. Al Hosain, M. Amin, F. Faisal, and M. M. Nishat, "Comparative analysis of machine learning algorithms in detection of brain tumor," in *2022 3rd International Conference on Big Data Analytics and Practices (IBDAP)*, pp. 31–36, IEEE, 2022.

[52] I. J. Ratul, U. H. Wani, M. M. Nishat, A. Al-Monsur, A. M. Ar-Rafi, F. Faisal, M. R. Kabir, *et al.*, "Survival prediction of children undergoing hematopoietic stem cell transplantation using different machine learning classifiers by performing chi-square test and hyperparameter optimization: a retrospective analysis," *Computational and Mathematical Methods in Medicine*, vol. 2022, 2022.

[53] T. Hasan, M. F. Bin Karim, M. K. Mahadi, M. M. Nishat, F. Faisal, *et al.*, "Employment of ensemble machine learning methods for human activity recognition," *Journal of Healthcare Engineering*, vol. 2022, 2022.

[54] M. R. Kabir, M. M. Muhaimin, M. A. Mahir, M. M. Nishat, F. Faisal, and N. N. I. Moubarak, "Procuring mfccs from crema-d dataset for sentiment analysis using deep learning models with hyperparameter tuning," in *2021 IEEE International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things (RAAICON)*, pp. 50–55, IEEE, 2021.

[55] M. K. Mahadi, S. R. Abir, A.-M. Moon, M. Adnan, M. A. N. I. Khan, M. M. Nishat, F. Faisal, and M. T. Reza, "Machine learning assisted decision support system for prediction of prostrate cancer," in *2023 20th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, pp. 1–5, IEEE, 2023.

[56] A. A. Rahman, M. D. R. Kabir, R. Ratul, F. A. Shams, M. M. Nishat, and F. Faisal, "An efficient analysis of eeg signals to perform emotion analysis," in *4th International Conference on Artificial Intelligence, Robotics and Control (AIRC)*, 2023. In Press.

[57] F. Faisal, M. M. Nishat, K. R. Raihan, A. Shafiullah, and S. Ali, "A machine learning approach for analyzing and predicting suicidal thoughts and behaviors," in *2023 Fourteenth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 43–48, IEEE, 2023.

[58] I. J. Ratul, M. M. Nishat, F. Faisal, S. Sultana, A. Ahmed, and M. A. Al Mamun, "Analyzing perceived psychological and social stress of university students: A machine learning approach," *Heliyon*, 2023.

[59] M. N. S. Shahi, S. S. Muhaimin, M. M. Nishat, F. Faisal, and M. S. B. Motahar, "Automated vehicle access and exit control: A smart parking management system using finite state machine in vhdl," in *2022 Research, Invention, and Innovation Congress: Innovative Electricals and Electronics (RI2C)*, pp. 170–177, IEEE, 2022.

[60] F. Faisal, M. M. Nishat, N. Tazreen, M. F. B. Karim, A. H. Bristy, and S. Haque, "Ex-vivo design of ph meter to analyze nutrients feature assessment for economic domestic purpose," in *2022 37th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, pp. 637–640, IEEE, 2022.

[61] N. Sakib, Z. B. Saif, M. T. Hasan, M. Adnan, F. Faisal, M. M. Nishat, and et al., "Eeg-driven age prediction: Advancements in machine learning models," in *Proceedings of the ICE3IS, IEEE 2023*. Accepted, in press.

[62] A. Newaz, M. S. Mohosheu, and M. A. Al Noman, "Predicting complications of myocardial infarction within several hours of hospitalization using data mining techniques," *Informatics in Medicine Unlocked*, vol. 42, p. 101361, 2023.

[63] H.-C. Wang and J.-J. Zhuang, "Effective fatigue driving detection by machine learning," in *International Conference on Wireless Intelligent and Distributed Environment for Communication*, pp. 59–75, Springer, 2023.

[64] F. A. Shams, M. S. Mohosheu, M. A. al Noman, S. R. Abir, Al-Amin, M. M. Nishat, and F. Faisal, "Roc based performance evaluation of machine learning classifiers for multiclass imbalanced intrusion detection dataset," in *8th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE 2023)*, in press.

[65] V. Stanev, C. Oses, A. G. Kusne, E. Rodriguez, J. Paglione, S. Curtarolo, and I. Takeuchi, "Machine learning modeling of superconducting critical temperature," *npj Computational Materials*, vol. 4, no. 1, p. 29, 2018.

[66] K. Hamidieh, "A data-driven statistical model for predicting the critical temperature of a superconductor," *Computational Materials Science*, vol. 154, pp. 346–354, 2018.

[67] S. Li, Y. Dan, X. Li, T. Hu, R. Dong, Z. Cao, and J. Hu, "Critical temperature prediction of superconductors based on atomic vectors and deep learning," *Symmetry*, vol. 12, no. 2, p. 262, 2020.

[68] P. J. García-Nieto, E. García-Gonzalo, and J. P. Paredes-Sánchez, "Prediction of the critical temperature of a superconductor by using the woa/mars, ridge, lasso and elastic-net machine learning techniques," *Neural Computing and Applications*, vol. 33, pp. 17131–17145, 2021.

[69] R. V. Babu, G. Ayyappan, and A. Kumaravel, "Comparison of linear regression and simple linear regression for critical temperature of semiconductor," *Indian J. Comput. Sci. Eng.*, vol. 10, pp. 177–183, 2020.

[70] P. Moscato, M. N. Haque, K. Huang, J. Sloan, and J. Corrales de Oliveira, "Learning to extrapolate using continued fractions: Predicting the critical temperature of superconductor materials," *Algorithms*, vol. 16, no. 8, p. 382, 2023.

[71] F. A. Shams, R. H. Ratul, A. I. Naf, S. S. H. Samir, M. M. Nishat, F. Faisal, and M. A. Hoque, "Investigation on machine learning based approaches for estimating the critical temperature of superconductors," *arXiv preprint arXiv:2308.01932*, 2023.

[72] National Institute of Materials Science, "Materials information station, supercon." http://supercon.nims.go.jp/index_en.html, 2022. Accessed: July 16, 2023.

[73] A. Frank, "Uci machine learning repository," *http://archive. ics. uci. edu/ml*, 2010.

[74] M. Muntasir Nishat, F. Faisal, I. Jahan Ratul, A. Al-Monsur, A. M. Ar-Rafi, S. M. Nasrullah, M. T. Reza, and M. R. H. Khan, "A comprehensive investigation of the performances of different machine learning classifiers with smote-enn oversampling technique and hyperparameter optimization for imbalanced heart failure dataset," *Scientific Programming*, vol. 2022, pp. 1–17, 2022.

[75] R. Zebari, A. Abdulazeez, D. Zeebaree, D. Zebari, and J. Saeed, "A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction," *Journal of Applied Science and Technology*

Trends, vol. 1, no. 2, pp. 56–70, 2020.
[76] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.
[77] E. Elgeldawi, A. Sayed, A. R. Galal, and A. M. Zaki, "Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis," in *Informatics*, vol. 8, p. 79, MDPI, 2021.
[78] C. Yan, J. Zhang, X. Kang, Z. Gong, J. Wang, and G. Zhang, "Comparison and evaluation of the combinations of feature selection and classifier on microarray data," in *2021 IEEE 6th International Conference on Big Data Analytics (ICBDA)*, pp. 133–137, IEEE, 2021.
[79] M. Schleich, D. Olteanu, and R. Ciucanu, "Learning linear regression models over factorized joins," in *Proceedings of the 2016 International Conference on Management of Data*, pp. 3–18, 2016.
[80] G. Van Dijck and M. M. Van Hulle, "Speeding up the wrapper feature subset selection in regression by mutual information relevance and redundancy analysis," in *International Conference on Artificial Neural Networks*, pp. 31–40, Springer, 2006.
[81] C. Shu and D. H. Burn, "Artificial neural network ensembles and their application in pooled flood frequency analysis," *Water Resources Research*, vol. 40, no. 9, 2004.
[82] R. Xia, Y. Gao, Y. Zhu, G. Dexi, and C. Wu, "A fast and efficient method combined data-driven for detecting electricity theft to secure the smart grid with stacking structure," *Available at SSRN 4019865*.
[83] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1249, 2018.
[84] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1249, 2018.
[85] B. Choudhury, "Evaluation of an empirical equation for annual evaporation using field observations and results from a biophysical model," *Journal of Hydrology*, vol. 216, no. 1-2, pp. 99–110, 1999.
[86] J. Tayman and D. A. Swanson, "On the validity of mape as a measure of population forecast accuracy," *Population Research and Policy Review*, vol. 18, pp. 299–322, 1999.
[87] J. Lupón, H. K. Gaggin, M. De Antonio, M. Domingo, A. Galán, E. Zamora, J. Vila, J. Peñafiel, A. Urrutia, E. Ferrer, *et al.*, "Biomarker-assist score for reverse remodeling prediction in heart failure: The st2-r2 score," *International journal of cardiology*, vol. 184, pp. 337–343, 2015.

Appendix

Table 6. Average RMSE under all conditions without hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	17.636	18.345	11.821	25.158	24.812	9.734	13.826
f-classif	18.624	18.655	10.788	15.513	14.649	9.748	12.400
RandomSearchal-info-regression	19.689	23.204	11.052	15.193	17.477	9.750	13.938
f-regression	18.705	23.159	10.718	14.657	17.315	9.768	13.652

Table 7. Average RMSE under all conditions with hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	17.627	17.677	10.332	16.826	13.938	9.686	11.875
f-classif	18.624	18.651	10.591	14.595	14.661	9.888	12.540
RandomSearchal-info-regression	19.622	19.639	10.691	16.05	14.803	9.727	12.648
f-regression	18.521	18.548	10.356	14.388	14.487	9.731	12.252

Table 8. Average R2 under all conditions without hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	0.7348	0.7131	0.8808	0.4041	0.4751	0.9191	0.8370
f-classif	0.7043	0.7033	0.9006	0.7947	0.8170	0.9189	0.8689
mutual-info-regression	0.6695	0.5410	0.8958	0.8032	0.7396	0.9189	0.8344
f-regression	0.7017	0.5428	0.9020	0.8167	0.7444	0.9186	0.8411

Table 9. Average R2 under all conditions with hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	0.7351	0.7336	0.9089	0.7571	0.8343	0.919958	0.8797
f-classif	0.7043	0.7034	0.9043	0.8166	0.81674	0.916636	0.8659
mutual-info-regression	0.671792	0.6712	0.9025	0.7774	0.81316	0.919301	0.8636
f-regression	0.707593	0.7067	0.9085	0.2169	0.8210	0.91922	0.8720

Table 10. Average MAE under all conditions without hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	13.3535	13.981	6.6822	20.406	18.524	5.475659	10.130
f-classif	14.3864	14.408	5.7242	10.375	8.9589	5.454676	8.7256
mutual-info-regression	14.9859	18.142	6.1600	10.349	12.065	5.405764	10.035
f-regression	14.377	18.110	6.0186	9.9280	11.841	5.465831	9.8239

Table 11. Average MAE under all conditions with hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	13.34874	13.3659	5.44815	11.49106	8.31017	5.383053	8.1803
f-classif	14.38648	14.4049	5.89576	9.47307	8.92848	5.557045	8.8195
mutual-info-regression	14.90684	14.9273	5.64775	10.38217	9.04408	5.438871	8.8572
f-regression	14.22273	14.2468	5.46085	24.37746	8.85529	5.440037	8.5893

Table 12. Average MAPE under all conditions without hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	12.86335	16.6023	5.464846	26.62253	15.40685	4.815816	10.3292
f-classif	16.81786	16.3090	5.488384	9.007473	7.637216	4.575237	9.732087
mutual-info-regression	16.17585	19.2875	5.071434	7.340627	10.02875	4.984084	9.649217
f-regression	15.81139	19.5317	5.687521	7.916581	9.335178	4.747403	9.386212

Table 13. Average MAPE under all conditions with hyperparameter optimization

Criteria	Ridge	Lasso	KNN	MLP	SVR	Stacking	Voting
All features	12.95154	13.003	5.05690	11.038	7.4839	4.80839	8.3418
f-classif	16.81786	16.351	5.5626	9.4737	7.2858	4.7112	9.7490
mutual-info-regression	17.03013	16.545	4.5454	9.5143	8.0557	5.250034	9.7261
f-regression	17.49597	16.945	4.3979	38.199	7.7147	4.818958	9.8588