EAI Endorsed Transactions
On AI and Robotics
Research Article EAI.EU

A Machine Learning based Investigative analysis for predicting critical temperature of superconductors

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

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 subjected to extensive investigation. However, identifying superconductors that exhibit such behavior 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 techniques have been developed 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 on 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 personals and research scientists to further assess the performance of the ML models.

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

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

doi: 10.4108/airo.3622

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

1. Introduction

Superconductivity has been a topic of great interest in condensed matter research for many years [1]. The phenomenon of zero electrical resistance that is exhibited by some metals below a critical temperature, Tc, has been observed in several metals such as indium, mercury, lead, tin, and niobium. The high-temperature superconductors, which possess the ability to conduct electricity without any resistance, have the potential for vast technological applications, including efficient energy distribution, transportation, and magnetic confinement of particles in nuclear fusion plants [2]. Despite over a century of research, determining how a material's composition and structure correlates with its superconducting properties remains a challenging task. The complexity of high-temperature superconductors arises from the fact that they cannot be fully characterized by the existing theories of superconductivity, which can only describe a small subset of actual superconductors [3]. This has led researchers to explore other methods, such as machine learning [4-13], to gain a deeper understanding of the relationships between superconductivity and a material's chemistry and structure. Machine learning algorithms [14-23] offer a unique opportunity to extract patterns and relationships from large datasets that traditional

methods may miss [24-36]. Additionally, for electron-phonon paired superconductors, for which the theory is rather well established, predicting the critical temperatures Tc of emerging superconductors is a notoriously tough task [37]. Previous attempts to develop a Tc formula compatible with strong coupling theory by McMillan, Allen, and Dynes resulted in closed-form approximations of relations between Tc and several metrics of the electron-phonon interaction that were identified in Eliashberg theory [38]. However, recent research has shown that the Tcs for more newly discovered superconducting materials, which have a higher two-dimensional electron-phonon interaction, do not match Allen and Dynes' formulation.

Accurately estimating the critical temperature (Tc) is a crucial aspect of the complex and resource-intensive process involved in synthesizing superconductors [39]. The viability of the synthesis process depends significantly on the accuracy of the Tc estimation. Therefore, the findings of this study could have potential implications for decision-making in the synthesis of superconductors [40]. The proposed approximation technique offers a reliable way for researchers to estimate the Tc of newly discovered superconducting materials. By accurately estimating Tc, researchers can efficiently identify materials with desirable superconducting properties, expediting the development of new superconductors. In this study 5 super-vised 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 features selection methods which were tested, i.e f_classif, f_regression and mutual info regression. Again, in order to analyze the difference of 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 randomized search cv. And the difference before and after this optimization is compared.

2. Literature Review

In recent years, there has been a surge of interest in using machine learning algorithms to predict the critical temperature of superconductors. To predict Tc from diverse sets of input characteristics, several models such as random forest, support vector machines, and artificial neural networks have been used [41]. Some research has also concentrated on finding important factors that contribute the most to Tc prediction. Despite substantial advances in this field, precisely predicting Tc 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 for the purpose of estimating the critical temperature (Tc) of superconductors. The review will examine the various machine learning models [42-60] that have been developed for this objective and their respective merits and limitations.

Hamidieh created a model in [61], 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 having better RMSE and R2 scores than us, our approach enables us to identify the features that are more crucial for predicting the Tc, but their approach is unable to do so. The study in [62] suggests a technique that describes materials using atomic vectors [38,39] and predicts Tc using a hybrid neural network 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 R2 score of 0.899 and MAE 5.023K; however, they only manage a poor RMSE of 83.565. Consequently, we achieve results that are better than this paper in terms of RMSE and R2.

A novel method was developed by Paulino et al. [63] 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 purpose. The results show that all four machine learning techniques are capable of predicting Tc with reasonable accuracy, but this hybrid WOA/MARS based model outperforms the rest three model with an RMSE value of 15.14, R2 score of 0.80 and MAE of 10.75. However, compared to this, in the context of RMSE, R2, and MAE, our work performs. Two popular regression methods, linear and simple linear regression models, were utilized in the research of Babu et al. [64] to compare various performance metrics. Better results are obtained by their linear regression model, which has an RMSE of 17.68, MAE of 13.42, and R2 of 0.7396. Once more, the RMSE, R2, and MAE results for our work are better.

In the study of Mohammad N. Haque et al. [65], 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 obtained RMSE of 10.989, which our work managed to out-perform of superconductors in terms of critical temperature. 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.

3. Methodology

3.1. Data Preprocessing

This dataset emphasizes a cutting-edge machine learning technique to extract complex superconductive material properties for critical temperature (Tc) 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) [66]. 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 exact same dataset from the UCI Machine Learning Repository [67] 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 underlining assumption that larger values have greater importance compared to lower values. Furthermore, in this study min max scaling was implemented to normalize the data, which can be expressed in the following mathematical expression.

$$Min-maxscaling, x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

3.3. Data Preprocessing

Table 1 Dataset Preprocessing

Feature Selection Method	Scaling Method	Feature Count	Status of hyper parameter
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

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 with the intention of striking 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 [68]. Table I shows the overall progression of dataset pre-processing.

3.4. Hyperparameter Tuning

In machine learning, the task of determining the ideal collection of hyperparameters for a learning algorithm is referred to as hyper-parameter optimization, or tuning. A parameter whosevalue is utilized to guide the learning process is termed as hyperparameter. For this process, we used a random search CV to find out the best hyper-parameter. RandomizedSearchCV appliesa "score" and a "fit" method. 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 comparison to a baseline where all the features were used. These specific feature selection methodshave not been previously utilized in the preprocessing of this dataset. The study is divided into four stages. The first stage involves obtaining baseline results withoutapplying 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.

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 [69]. 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.

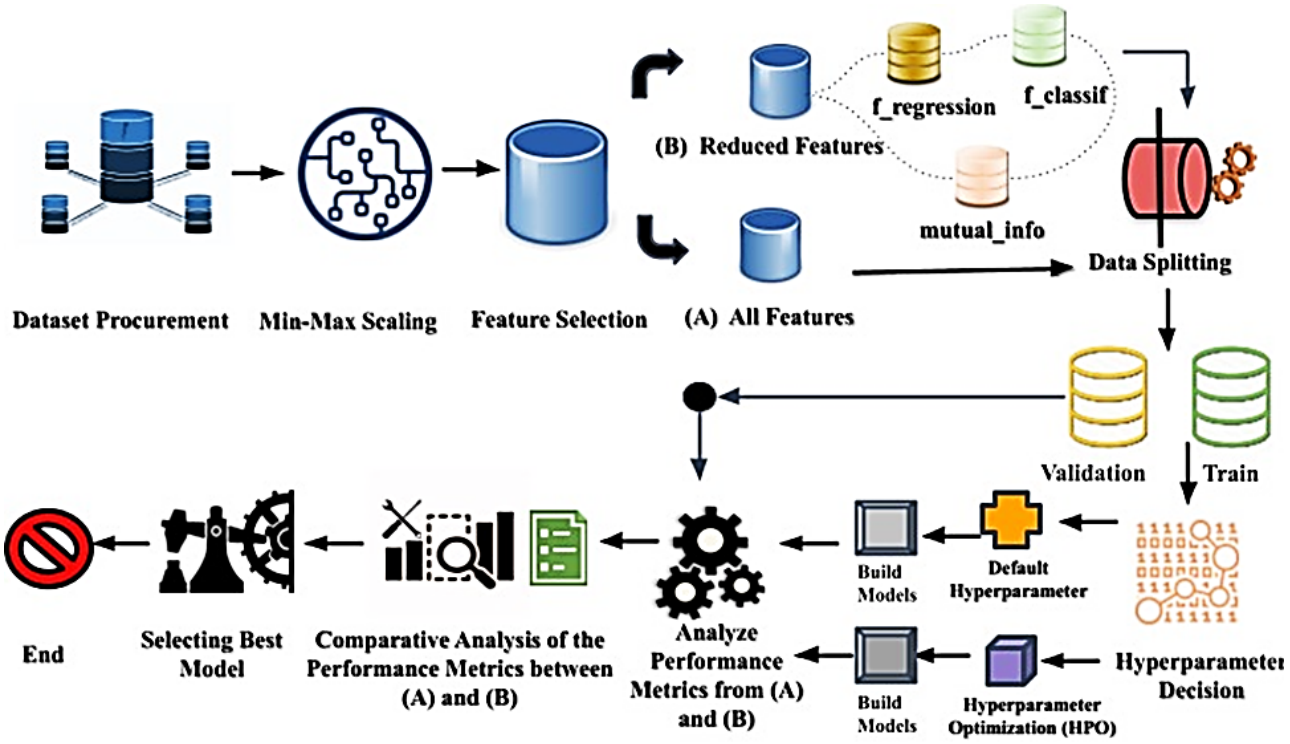

Figure 1. Overall Workflow Diagram

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. Overall, the workflow of this research article involves pre-processing the data using min-max scaling, followed by evaluating the performance of three feature selection techniques individually and 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

In order to compare model performance with fewer features, a baseline was established using all 81 features. The 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 [70]. The resulting optimal hyperparameter configurations for each model can be found in Table 2.

3.6. Approach II

In order to compare model performance with fewer features, a baseline was established using all 81 features. The 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 [70]. The resulting optimal hyperparameter configurations for each model can be found in Table 3.

3.6. Approach III

In order to compare model performance with fewer features, a baseline was established using all 81 features. The 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 [70]. The resulting optimal hyperparameter configurations for each model can be found in Table 4.

3.7. Approach IV

Mutual information is a nonnegative measure of the interdependence between two random variables. Mutual information quantifies variable dependence, with zero indicating independence and higher values indicating stronger dependence.

This section utilized mutual information regression. This method utilizes nonparametric algorithms based on k-nearest neighbor distances to estimate entropy [71]. Optimal hyperparameters were determined through the RandomSearch CV method after initially using default hyperparameters. Table 5 summarizes the best hyperparameters for each model.

Table 2 Hyperparameter optimization on all the 81 features

Algorithms	Hyperparameters passed		Best Hyperparameters
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
MLP	degree	1, 2, 3, 4, 5,	1
	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
	max_depth	5,10,15,20	20
	min_samples_split	2,4,8,10	2

Table 3 Hyperparameter optimization on 50 features selected by f_classif

Algorithms	Hyperparameters passed		Best Hyperparameters
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	0.0001
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	2
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
MLP	degree	1, 2, 3, 4, 5	3
	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)	(56, 32, 25)
RFR	n_estimators	20,40,60,80,100,120	120
	max_depth	5,10,15,20	20
	min_samples_split	2,4,8,10	2

Table 4 Hyperparameter optimization on 50 features selected by f_regression

Algorithms	Hyperparameters passed		Best Hyperparameters
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	0.01
MLP	degree	1, 2, 3, 4, 5	1
	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)	(56, 52, 78, 30)
RFR	n_estimators	20,40,60,80,100,120	120
	max_depth	5,10,15,20	20
	min_samples_split	2,4,8,10	4

Stacking is a method that can be used to ensemble a number of different classification or regression models [72].

Table 5 Hyperparameter optimization on 50 features selected by mutual_info

Algorithms	Hyperparameters passed		Best Hyperparameters
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	0.001, 0.0001, 0.01, 0.1, 0.00001, 0.000001, 0.0000001	0.01
KNNv	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
MLP	degree	1, 2, 3, 4, 5	2
	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
	max_depth	5,10,15,20	20
	min_samples_split	2,4,8,10	4

3.8. Model Description

3.8.1. Stacking Model

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 a high volatility. Boosting is the process of

building numerous incremental models in order to reduce bias while maintaining a low variance.

When used to 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 is able to obtain the predictions of several learning models, which are subsequently utilized as the meta-data in order 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 [73]. In our approach, the hyper-parameters of the final estimator were optimized while evaluating the performance of the proposed algorithm. In our suggested model, five algorithms were used as estimators, 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. All these base estimators were once used with the default hyper-parameters and once with the hyper-parameters found while optimizing each of those standalone models with RandomSearch CV.

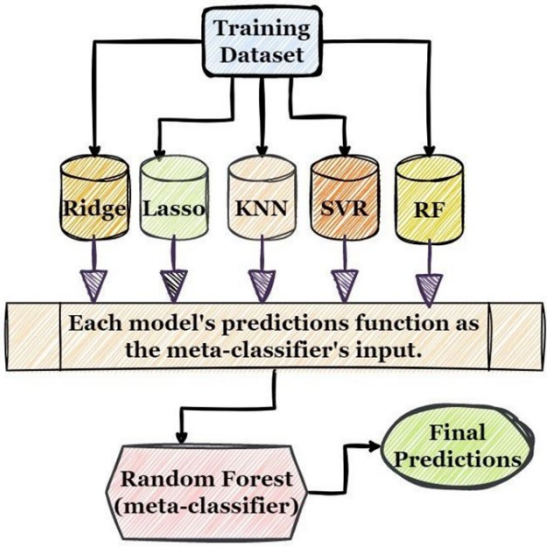

Figure 2. Stacking Model

3.8.2. 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 [74]. After all of the models have made their predictions for the sample data, 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. All of the models were first used with the default hyper-parameters. In order to optimize the voting ensemble model further, the best hyper- parameters which were found using RandomSearch CV 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.

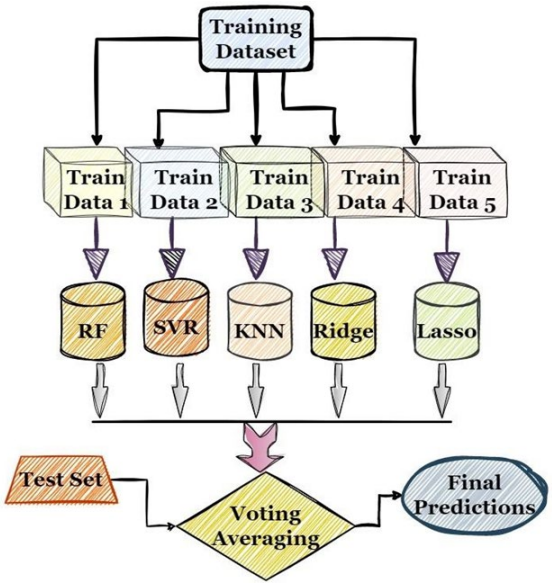

Figure 3. Voting Model

4. Evaluation Metrics

4.1. RMSE

From the RMSE, the standard deviation of the error can be found in predictions. Residuals or 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 [75-76]:

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{e_i^2}{n}} \quad (2)$$

where n is the number of samples and e is the error term.

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 [77]:

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | e_i | \tag{3}$$

where e is the error term and n is the number of samples. MAE presents itself to be the most natural measure of average error magnitude, and that (unlike RMSE) it is an unambiguous measure of average error magnitude.

4.3. MAPE

Mean absolute percentage error is the most common error analysis technique used for forecasting. It measures accuracy as percentage. MAPE can be represented by the following mathematical equation [78]:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left| \frac{A_t - F_t}{A_t} \right| \tag{4}$$

where n is the number of fitted points, A_t is the actual value, F_t is the forecast value. 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 [79]. 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.

$$R^2 = \frac{Varience\_Explained\_by\_Model}{Total\_Varience} \tag{5}$$

where n is the number of fitted points, A_t is the actual value, F_t is the forecast value. MAPE is generally used when the quantity to be predicted remains much higher than zero.

5. Results

In this section, the testing set created in each of the approaches mentioned above is tested with the evaluation metrics which 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 Figure 4. Table 6 and Table 7 consequently shows 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 hyper-parameters were used and the latter does the same for the case when the hyper-parameters were optimized. From these two table the stacking model achieves the best result of RMSE 9.686 after hyperparameter optimization with all 81 characteristics taken into account.

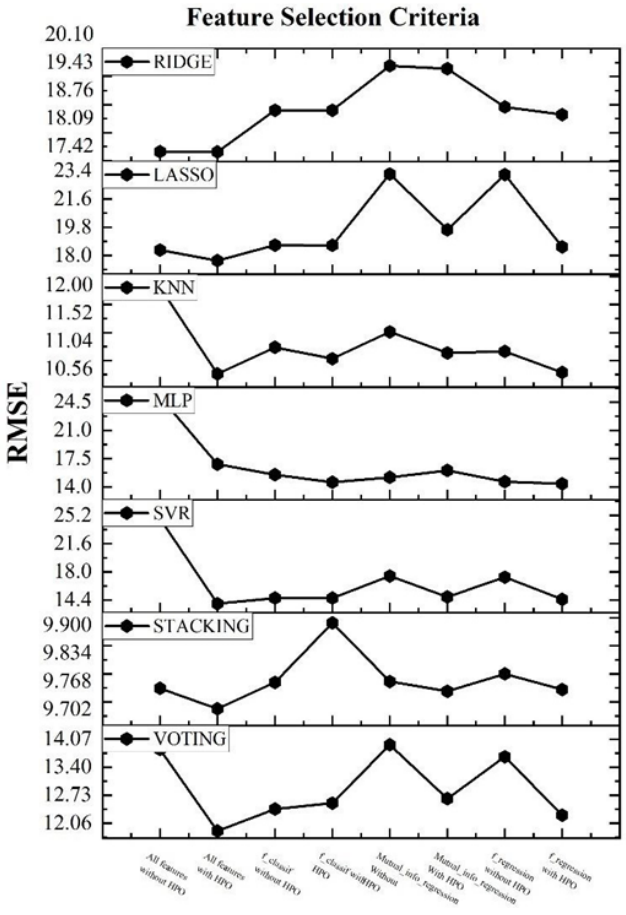

Figure 4. RMSE for different ML models in terms of feature selection methods

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
mutual_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
mutual_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 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 11 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

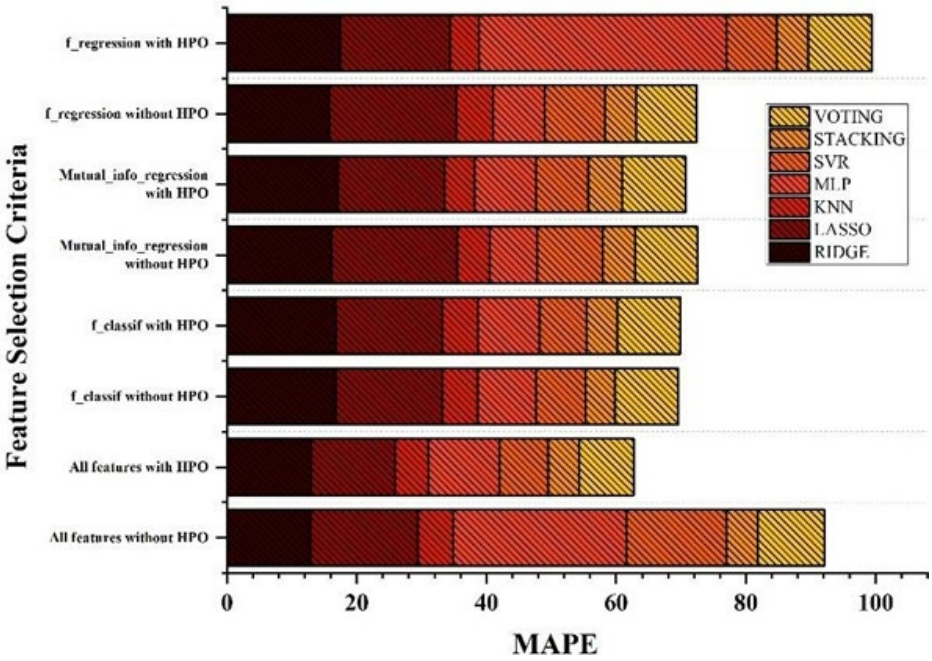

Figure 5. Feature Selection Criteria vs MAPE

The stacking model has 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 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 technique 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 cased, shown in table X and XI. The second lowest MAE score of 5.448 was achieved by KNN model when all features were used with HPO.

6. Conclusion

This study takes an attempt to analyse 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 dependability of critical temperature estimation under the feature selection methods tested. A deep learning model to predict the critical temperature of super-conductors is under development. In future we would also like to incorporate the readings of the properties of newly found superconductors in our study to further assess the performance of the ML models.

References

[1] Fradkin, Eduardo. *Field theories of condensed matter physics*. Cambridge University Press, 2013.

[2] Zhang, Hongye, Zezhao Wen, Francesco Grilli, Konstantinos Gyftakis, and Markus Mueller. "Alternating current loss of superconductors applied to superconducting electrical machines." *Energies* 14, no. 8 (2021): 2234.

[3] Flores-Livas, José A., Lilia Boeri, Antonio Sanna, Gianni Profeta, Ryotaro Arita, and Mikhail Eremets. "A perspective on conventional high-temperature superconductors at high pressure: Methods and materials." *Physics Reports* 856 (2020): 1-78.

[4] 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," *Biomed. and Pharm. J.,* vol. 12, no. 3, pp. 1249-1262, 2019

[5] M. M. Nishat, et al., "An Investigative Approach to Employ Support Vector Classifier as a Potential Detector of Brain Cancer from MRI Dataset," *ICECIT*, pp. 1-4, doi: 10.1109/ICECIT54077.2021.9641168

[6] M. A. A. R. Asif *et al*., "Performance Evaluation and Comparative Analysis of Different ML Algorithms in Predicting Cardiovascular Disease," *Engineering Letters,* vol. 29, no. 2, pp. 731-741, 2021

[7] M. M. Nishat *et al*., "A Comprehensive Analysis on Detecting Chronic Kidney Disease by Employing Machine Learning Algorithms," *EAI Endorsed Transactions on Pervasive Health and Technology*, vol. 18, no. e6, 2021, doi: 10.4108/eai.13-8-2021.170671.

[8] M. R. Farazi, F. Faisal, Z. Zaman and S. Farhan, "Inpainting multiple sclerosis lesions for improving registration performance with brain atlas," *MediTec*, pp 1-6, doi: 10.1109/MEDITEC.2016.7835363.

[9] M. A. A. R. Asif *et al*., "Computer Aided Diagnosis of Thyroid Disease Using Machine Learning Algorithms," *ICECE*, 2020, pp. 222-225, doi: 10.1109/ICECE51571.2020.9393054

[10] M. M. Nishat and F. Faisal, "An Investigation of Spectroscopic Characterization on Biological Tissue," *iCEEiCT*, 2018, pp. 290-295, doi: 10.1109/CEEICT.2018.8628081

[11] F. Faisal, et al., "Covid-19 and its impact on school closures: a predictive analysis using machine learning algorithms," *ICSCT*, 2021, pp. 1-6, doi: 10.1109/ICSCT53883.2021.9642617

[12] A. A. Rahman *et al*., "Detection of Epileptic Seizure from EEG Signal Data by Employing Machine Learning Algorithms with Hyperparameter Optimization," *BioSMART*, 2021, pp. 1-4, doi: 10.1109/BioSMART54244.2021.9677770.

[13] M. M. Nishat, et al., "Detection of Parkinson's Disease by Employing Boosting Algorithms," *ICIEV and icIVPR*, pp. 1-7, 2021, doi: 10.1109/ICIEVicIVPR52578.2021.956410.

[14] Nishat, M. M., et al. "Detection of Autism Spectrum Disorder by Discriminant Analysis Algorithm," *BIM,* pp. 473-482, Springer, Singapore, 2022 doi: 10.1007/978-981-16-6636-0_36

[15] A. A. Rahman, et al., "Detection of Mental State from EEG Signal Data: An Investigation with Machine Learning Classifiers," *KST*, pp. 152-156, 2022, doi: 10.1109/KST53302.2022.9729084.

[16] 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," *2021 3rdInt. Conf. on STI 4.0*, 2021, pp. 1-6, doi: 10.1109/STI53101.2021.9732592.

[17] Nishat, M. M., et al. "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*, https://doi.org/10.1155/2022/3649406.

[18] Nishat, M. M, Faisal, F., Mahbub, M. A., Mahbub, M. H., et al., "Performance Assessment of Different Machine Learning Algorithms in Predicting Diabetes Mellitus". *Biosc.Biotech.Res.Comm.*,14(1), 2021 doi: http://dx.doi.org/10.21786/bbrc/14.1/10.

[19] F. Faisal, "Image Inpainting to Improve the Registration Performance of Multiple Sclerosis (MS) Patient Brain with Brain Atlas," Diss. Department of EEE, IUT, OIC, Gazipur-1704, Bangladesh, 2016.

[20] M. M. Nishat *et al*., "Performance Investigation of Different Boosting Algorithms in Predicting Chronic Kidney Disease," *2020 2nd Int. Conf. on STI 4.0*, pp. 1-5, 2020,doi: 10.1109/STI50764.2020.9350440

[21] 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," *ICBIR*, 2022, pp. 717-722, doi: 10.1109/ICBIR54589.2022.9786472

[22] I. J. Ratul, A. A. Monsur, B. Tabassum, A. M. Ar-Rafi, M. M. Nishat and F. Faisal, "Early risk prediction of cervical cancer: A machine learning approach," *ECTI-CON,* 2022, pp. 1-4, doi: 10.1109/ECTI-CON54298.2022.9795429.

[23] 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," *AIIoT*, 2022, pp. 351-356, doi: 10.1109/AIIoT54504.2022.9817237.

[24] I. A. Jahan, et al., "Accident Detection and Road Condition Monitoring Using Blackbox Module," *ICUFN*, 2022, pp. 247-252, doi: 10.1109/ICUFN55119.2022.9829589.

[25] A. A. Rahman *et al*., "Perceived Stress Analysis of Undergraduate Students during COVID-19: A ML Approach," *MELECON*, 2022, pp. 1129-1134, doi: 10.1109/MELECON53508.2022.9843081.

[26] F. Faisal, M. M. Nishat and M. A. M. Oninda, "Spectroscopic Characterization of Biological Tissue using Quantitative Acoustics Technique," *iCEEiCT*, doi: 10.1109/CEEICT.2018.8628146

[27] Z. K. Eisham *et al.,* "Chimp optimization algorithm in multilevel image thresholding and image clustering," *Evolving Systems,* 2022. https://doi.org/10.1007/s12530-022-09443-3.

[28] F. Faisal, M. A. Salam, M. B. Habib, M. S. Islam and M. M. Nishat, "Depth estmation from Video using Computer Vision and Machine Learning with Hyperparameter Optimization," *ICSSA,* 2022, pp. 39-44, doi: 10.1109/ICSSA54161.2022.9870961.

[29] M. M. Nishat, F. Faisal, M. A. M. Oninda and M. A. Hoque, "Modeling Simulation and Performance Analysis of SEPIC Converter Using Hysteresis Current Control and PI Control Method", *2018 Int. Conf. on Inv. in Science Engg. and Tech. (ICISET)*, pp. 7-12

[30] 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", *1st Int. Conf. on Advances in Science Engg. and Robotics Tech. (ICASERT)*, pp. 1-6, 2019.

[31] A. A. 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", *International Conference on Science & Cont. Technology (ICSCT)*, pp. 1-6, 2021.

[32] A. A. Mehadi, M. A. Chowdhury, M. M. Nishat, F. Faisal and M. M. Islam, "A software-based approach in designing a rooftop bifacial PV system for the North Hall of Residence IUT", *Clean Energy*, vol. 5, no. 3, pp. 403-422, 2021.

[33] M. M. Nishat, F. Faisal, A. J. Evan, M. M. Rahaman, M. S. Sifat and H. M. F. Rabbi, "Development of Genetic Algorithm (GA) Based Optimized PID Controller for Stability Analysis of DC-DC Buck Converter", *JPEE*, vol. 8, no. 9, pp. 8-19, 2020.

[34] Moshayedi, Ata Jahangir, Amin Kolahdooz, Atanu Shuvam Roy, Seyyed Ali Latifi Rostami, and Xiaoyun Xie. "Design and promotion of cost-effective IOT-based heart rate monitoring." CICA, vol. 12303, pp. 405-410. SPIE, 2022.

[35] Moshayedi, Ata Jahangir, et al. "Raspberry Pi SCADA zonal based system for agricultural plant monitoring." 6th ICISCE. IEEE, 2019.

[36] Moshayedi, Ata Jahangir, et al. "Review On: The Service Robot Mathematical Model." EAI Endorsed Transactions on AI and Robotics 1 (2022): 1-19.
[37] Xie, Stephan R., Gregory R. Stewart, James J. Hamlin, Peter J. Hirschfeld, and Richard G.Hennig. "Functional form of the superconducting critical temperature from machine learning." *Physical Review B* 100, no. 17 (2019): 174513
[38] Méndez-Moreno, R. M. "A Schematic Two Overlapping-Band Model for Superconducting Sulfur Hydrides: The Isotope Mass Exponent." *Advances in Condensed Matter Physics* 2019 (2019): 1-7.
[39] Lilia, Boeri, Richard Hennig, Peter Hirschfeld, Gianni Profeta, Antonio Sanna, Eva Zurek, Warren E. Pickett et al. "The 2021 room-temperature superconductivity roadmap." *Journal of Physics: Condensed Matter* 34, no. 18 (2022): 183002.
[40] Yazdani-Asrami, Mohammad, Wenjuan Song, Antonio Morandi, Giovanni De Carne, Joao Murta-Pina, Anabela Pronto, Roberto Oliveira et al. "Roadmap on artificial intelligence and big data techniques for superconductivity." *Superconductor Science and Technology* 36, no. 4 (2023): 043501.
[41] Stanev, Valentin, Corey Oses, A. Gilad Kusne, Efrain Rodriguez, Johnpierre Paglione, Stefano Curtarolo, and Ichiro Takeuchi. "Machine learning modeling of superconducting critical temperature." *npj Computational Materials* 4, no. 1 (2018): 29.
[42] F. Faisal, M. M. Nishat and M. R. Mia, "An Investigation on DC Motor Braking System by Implementing Electromagnetic Relay and Timer", *International Conference on Electrical Computer and Communication Engineering*, pp. 1-6, 2019.
[43] M. M. Nishat, M. R. K. Shagor, H. Akter, S. A. Mim, F. Faisal, M. M. Nishat, et al., "An Optimal Design of PID controller for DC-DC Zeta converter using Particle Swarm Optimization", *23rd International Conference on Comp. and Info. Tech. (ICCIT)*, pp. 1-6, 2020.
[44] M. D. Rahman, F. Faisal, M. M. Nishat and M. R. K. Shagor, " Design and Analysis of Passive LC 3 Component Boost Converter ", *2021 IEEE Madras Section Conference (MASCON)*, pp. 1-6, 2021.
[45] M. R. K. Shagor, A. J. Mahmud, M. M. Nishat, F. Faisal, M. H. Mithun and M. A. Khan, "Firefly Algorithm Based Optimized PID Controller for Stability Analysis of DC-DC SEPIC Converter", *12th UEMCON*, pp. 0957-0963, 2021
[46] A. A. Mehadi et al., "Design simulation and analysis of monofacial solar pv panel based energy system for university residence: a case study", *IOP Conf. Series: Mater. Sci. and Eng.*, vol. 1045, no. 1, pp. 012011, 2021.
[47] F. Faisal, M. Rahman and F. B. Hashem, "*Speed control of dc motor using Self-tuned fuzzy PID controller"*, Diss. Department of EEE, IUT, OIC, Gazipur-1704, Bangladesh, 2015
[48] S. Hassan *et al.*, "Comparative Analysis of Machine Learning Algorithms in Detection of Brain Tumor," *IBDAP*, 2022, pp. 31-36, doi: 10.1109/IBDAP55587.2022.9907433. IEEE, 2022
[49] I. J. Ratul *et al.*, "Survival Prediction of Children Undergoing Hematopoietic Stem Cell Transplantation Using Different MLCs by Performing Chi-Square Test and Hyperparameter Optimization: A Retrospe. Analysis", *CMMM*, https://doi.org/10.1155/2022/9391136.
[50] T. Hasan, et al., "Employment of Ensemble Machine Learning Methods for Human Activity Recognition", *Journal of Healthcare Engineering*, vol. 2022, https://doi.org/10.1155/2022/6963891.
[51] M. R. Kabir, et al., "Procuring MFCCs from Crema-D Dataset for Sentiment Analysis using DL Models with Hyperparameter Tuning," *RAAICON*, pp. 50-55. doi: 10.1109/RAAICON54709.2021.9929975.
[52] M. K. Mahadi *et al.*, "Machine Learning Assisted Decision Support System for Prediction of Prostrate Cancer," *ECTI-CON*, 2023, pp. 1-5, doi: 10.1109/ECTI-CON58255.2023.10153167.
[53] A. A. Rahman, M. R. Kabir, R. H. Ratul, F. A. Shams, M. M. Nishat and F. Faisal, "An Efficient Analysis of EEG Signals to Perform Emotion Analysis", *4th Int. Conf. on Artificial Intelligence, Robotics and Control (AIRC 2023)*, Cairo, Egypt, IEEE, accepted, in press
[54] 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," *ICUFN,* 2023, pp. 43-48, doi: 10.1109/ICUFN57995.2023.10201075.
[55] 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", *AIRoSIP*, IEEE 2023, accepted, in press
[56] I. J. Ratul, M. M. Nishat, F. Faisal, S. Sultana, A. Ahmed, M A. A. Mamun, "Analyzing Perceived Psychological and Social Stress of University Students: A Machine Learning Approach", *Heliyon*, 9(6), E17207, doi: 10.1016/j.heliyon.2023.e17307.
[57] M. N. S. Shahi, et al., "Automated Vehicle Access and Exit Control: A Smart Parking Management System using Finite State Machine in VHDL," *RI2C*, doi: 10.1109/RI2C56397.2022.9910305.
[58] F. Faisal, et al., "Ex-Vivo Design of pH meter To Analyze Nutrients Feature Assessment for Economic Domestic Purpose," *ITC-CSCC*, 2022, pp. 637-640, doi: 10.1109/ITC-CSCC55581.2022.9895032.
[59] A. Shafiullah, F. Faisal, R. H. Badhon, S. Ali, M. M. Nishat, M. R. Muttaqi and M. A. Billah, "Detection of Skin Cancer: A Deep Learning Approach," *20th Int. Conf. on Ubiquitous Intelligence and Computing (UIC 2023)*, Portsmouth, UK, IEEE, accepted, in press
[60] N. Sakib, Z. B. Saif, M. T. Hasan, M. Adnan, F. Faisal, M. M. Nishat, et al., "EEG-Driven Age Prediction: Advancements in Machine Learning Models", *ICE3IS*, IEEE 2023, accepted, in press.
[61] Hamidieh, Kam. "A data-driven statistical model for predicting the critical temperature of a superconductor." Computational Materials Science 154 (2018): 346-354.
[62] Li, Shaobo, Yabo Dan, Xiang Li, Tiantian Hu, Rongzhi Dong, Zhuo Cao, and Jianjun Hu. "Critical temperature prediction of superconductors based on atomic vectors and deeplearning." Symmetry 12, no. 2 (2020): 262.
[63] García-Nieto, Paulino José, Esperanza García-Gonzalo, and José Pablo 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* 33 (2021): 17131-17145.
[64] Babu, R. Venkatesh, G. Ayyappan, and A. Kumaravel. "Comparison of Linear Regression and Simple Linear Regression for critical temperature of semiconductor"
[65] Moscato, Pablo, Mohammad Nazmul Haque, Kevin Huang, Julia Sloan, and Jon C. de Oliveira. "Learning to extrapolate using continued fractions: Predicting the critical temperature of superconductor materials." arXiv preprint arXiv:2012.03774 (2020).
[66] Superconducting Material (SuperCon) Database (2021) National Institute for Materials Science (NIMS), Japan. https://supercon.nims.go.jp/en.

[67] UCI Machine Learning Repository: Superconductivity Data Set, https://archive.ics.uci.edu/ml/datasets/Superconductivty+Data

[68] Nishat, Mirza Muntasir, Fahim Faisal, Ishrak Jahan Ratul, Abdullah Al-Monsur, Abrar Mohammad Ar-Rafi, Sarker Md. Nasrullah, Md. Taslim Reza, and Rezaul Karim 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* 2022 (March 9, 2022): 1–17.

[69] Zebari, Rizgar, Adnan Abdulazeez, Diyar Zeebaree, Dilovan Zebari, and Jwan Saeed. "A comprehensive review of dimensionality reduction techniques for feature selection and feature extraction." Journal of Applied Science and Technology Trends 1, no. 2 (2020): 56-70.

[70] Chandrashekar, Girish, and Ferat Sahin. "A survey on feature selection methods." Computers & Electrical Engineering 40, no. 1 (2014): 16-28.

[71] Olteanu, D. A., and M. J. Schleich. "F: Regression models over factorized views." Proceedings of the VLDB Endowment 9, no. 13 (2016).

[72] Van Dijck, Gert, and Marc M. Van Hulle. "Speeding up the wrapper feature subset selection in regression by mutual information relevance and redundancy analysis." In Artificial Neural Networks–ICANN 2006: 16th International Conference, Athens, Greece, September 10-14, 2006. Proceedings, Part I 16, pp. 31-40. Springer Berlin Heidelberg, 2006.

[73] Shu, Chang, and Donald H. Burn. "Artificial neural network ensembles and their application in pooled flood frequency analysis." Water Resources Research 40, no. 9 (2004).

[74] Xia, Rui, Yunpeng Gao, Yanqing Zhu, G. U. Dexi, and Cong Wu. "A Fast and EfficientMethod Combined Data-Driven for Detecting Electricity Theft to Secure the Smart Grid with Stacking Structure." Available at SSRN 4019865.

[75] Sagi, Omer, and Lior Rokach. "Ensemble learning: A survey." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 8, no. 4 (2018): e1249.

[76] Fortin, Vincent, Mabrouk Abaza, Francois Anctil, and Raphael Turcotte. "Why shouldensemble spread match the RMSE of the ensemble mean?." Journal of Hydrometeorology 15, no. 4 (2014): 1708-1713.

[77] Choudhury, BhaskarJ. "Evaluation of an empirical equation for annual evaporation using field observations and results from a biophysical model." Journal of Hydrology 216,no. 1-2 (1999): 99-110.

[78] Tayman, Jeff, and David A. Swanson. "On the validity of MAPE as a measure of population forecast accuracy." *Population Research and Policy Review* 18 (1999): 299-322.

[79] Lupón, Josep, Hanna K. Gaggin, Marta De Antonio, Mar Domingo, Amparo Galán, Elisabet Zamora, Joan Vila et al. "Biomarker-assist score for reverse remodeling prediction in heart failure: The ST2-R2 score." International journal of cardiology 184 (2015): 337-343.