Sustainable Machine Intelligence Journal

Journal Homepage: sciencesforce.com/smij

Sustain. Mach. Intell. J. Vol. 10 (2025) 50–66

Paper Type: Original Article

Enhancing Software Effort Estimation in Healthcare Informatics: A Comparative Analysis of Machine Learning Models with Correlation-Based Feature Selection

Muhammad Abid [1,*], Sama Bukhari [2] and Muhammad Saqlain [3]

[1] Department of Mathematics, North Carolina State University, Raleigh, 27606, NC, United States; mabid@ncsu.edu.
[2] Department of Mathematics, International Islamic University Islamabad, Islamabad, 44000, Pakistan; samabukhari@gmail.com.
[3] School of Mathematics, Northwest University (NWU), Xi'an Shaanxi, 710069, China; msgondal0@gmail.com.

Received: 05 Aug 2024 **Revised**: 02 Dec 2024 **Accepted**: 29 Dec 2024 **Published**: 01 Jan 2025

Abstract

Software effort estimation is one of the most crucial processes in the management of software projects predominantly related to the healthcare industry. It involves the prediction of efforts needed to develop and endorse different software applications. To render clinical projects on time within the budget range, flawless projection with efficient planning is incumbent. This paper discloses the techniques that utilize machine learning models for ameliorating software effort estimation by using biomedical datasets, including Breast Cancer Wisconsin, COVID-19, Sleepy Drivers EEG Brainwave, Heart Disease Prediction and Food Nutrition. All of these datasets are cleaned and prepared by handling missing values, converting categorical features, and splitting data into training and testing sets and are being trained by four popular machine learning models; Linear Regression, Gradient Boosting, Random Forest, and Decision Tree. Furthermore, correlation based features are selected in the feature matrix to investigate the influence of statistically linked features and to promote reliability. For evaluation and measurement of the effectiveness of these models, two performance metrics namely: R^2 and Root Mean Squared Error are employed. The outcomes of the study delineate that Linear Regression and Gradient Boosting models give substantially better results than other models when choosing features on the basis of correlation. R^2 scores are strikingly impressive for Food Nutrition, Breast Cancer, COVID-19, while RMSE scores are lowest for COVID-19 dataset, showing high accuracy. It has been noted that features selection on the basis of correlation can highly optimize the performance of machine learning models. This juxtaposed analysis provides a solid framework for future research, enabling project managers to further enhance these findings to build fact based effort prediction.

Keywords: Machine Learning; Software Effort; Healthcare Informatics; Feature Selection.

1 | Introduction

Economical management of software projects requires the competent ability to calculate the efforts needed for the development and maintenance of software applications. This convoluted process is a must for project success, entailing flawless predictions for powerful planning and hence affirming on-time and on-budget deliveries [1]. This paper will be deeply diving into the broad field of machine learning to substantially improve

Corresponding Author: mabid@ncsu.edu
https://doi.org/10.61356/SMIJ.2025.10451

the accuracy of the degree of software effort estimation [2]. By deploying real-world datasets, this study reflects on the extensive analysis of the software development process [3].

The experimental framework of this study is anchored in the usage of five public datasets-Breast Cancer Wisconsin [4], COVID-19 [5], Sleepy Drivers EEG Brainwave [6], Heart Disease Prediction [7], and Food Nutrition [8]. For the sake of Scrupulous preparation of the datasets, diverse pre-processing techniques were used [9], including the fastidious handling of missing values, the purposeful transformation of categorical features for suitable analysis, and the careful splitting of data into training and testing sets [10]. These preparation steps play quite a crucial role in the accurate analysis of the data.

The whole inspection is dependent on four core machine learning models: Linear Regression, Gradient Boosting, Random Forest, and Decision Tree [11]. Every model was tested for its potency in software effort estimation [12, 13]. The main component of this investigation was the usage of features having a strong correlation between them. This method is Indispensable for addressing the complexity of software development where enormous factors can affect Project deadlines and resource allocation [14]. Furthermore, for tracking the performance of all of these four models, two performance metrics namely R2 and RMSE were utilized [15]. The R2 scores were remarkable for Food Nutrition, Breast Cancer, and COVID-19, while RMSE scores were lowest for the COVID-19 dataset, illustrating high accuracy.

This research provides a crucial map between machine learning and software effort estimation. By analyzing this convergence [17], the analysis laid a solid foundation for future research [16]. In the dynamic field of tech in which industries are continuously facing new challenges, the findings of this study provide detailed guidance for software projects [18]. The apparent structure of the paper promises a systematic study of the topic. In the Introduction, it highlights the framework of the study by giving the basic information and purpose of the study [25-28]. The Ease-of-Access section explains important topics like data preparation, over-fitting, and Feature Selection and elaborates on the issues and the solutions being discovered during the research. The associated sections rigorously examine the prior research, emphasizing the elimination of nested features, tackling the shortcomings, and depicting the paper's impact on the field [29-33]. The utilization of machine-learning models in Software Estimation draws Attention to the importance of incorporating machine-learning techniques for system development [34-37]. The investigation method, datasets, pre-processing processes, and performance metrics are lucidly explicated in the methodology section. All research outcomes and insightful conclusions are attentively presented in the subsequent sections, serving a consistent and clear display of the research findings [38-41]. Figure 1 shows the main segments related to the problem configuration and provides a graphical exhibition of all sections.

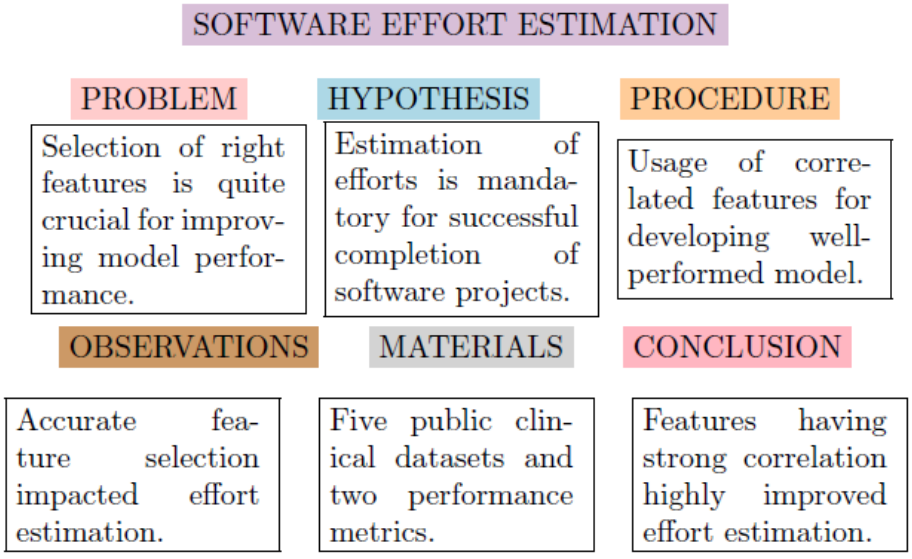

Figure 1. Illustration of software effort estimation.

2 | Accessibility and Optimization

2.1 | Data Preparation

Preparation of data is one of the most fundamental steps in machine learning which involves multiple steps. For instance, the transformation of raw data into tabulated data so that it can be easily understood and examined by machine learning models [20, 21]. Additionally, cleaning the tabulated data by managing null values and converting categorical features into numerical ones to function properly is a significant step in data preparation. Furthermore, it includes stacking data, selecting proper features, and splitting the data into training and testing sets. The models are being trained on the features being added to the feature matrix. Therefore, it is essential to choose them carefully. According to the survey conducted by Anaconda, professionals spend almost 50 percent of their time loading and cleaning, which highlights the importance of preparation of data [22].

2.2 | Overfitting

Over-fitting is a major challenge in machine learning. It happens when the model learns the underlying patterns in the training set so much that it performs exceptionally well in the training set but exceptionally poorly in the testing set. Therefore, we cannot come up with a generalized model. This happens due to poor choice of features. There is a need for extra attention while selecting features in the features matrix.

2.3 | Feature Selection

Features are independent variables in data based on which values of dependent variables are computed. One of the most pivotal steps in the development of any project is the selection of features [23]. Every feature included in a dataset influences the required value. Before splitting data into training and testing sets, the feature matrix and target vector are selected. The feature matrix contains all the features required for developing the project. Inaccurate choice of features in the feature matrix resulted in many different issues including leakage and over-fitting consequently ruining the performance of the model. Therefore, choosing highly relevant features is imperative. The best approach most of the developers use is choosing them based on correlation. Correlated Features significantly improve the performance of machine learning algorithms.

2.4 | Related Work

Forecasting project effort estimation is of paramount importance for bearing fruit in any project related to software. It requires meticulous and thorough attention while doing projects related to the medical field. For illustration, if someone is doing projects for predicting heart disease, he/she must have to pay scrupulous attention to the impediments like time and budget. Because even minor negligence might be brought severe consequences or fatal outcomes.

This study is an extension of the study on software effort estimation [24] where authors used datasets spanning multiple disciplines to estimate the extent of efforts needed to complete any project. However, in this inquiry, our main goal is to predict the level of efforts in the healthcare sector by employing clinical datasets: breast cancer, coronavirus, heart disease, EEG of the brain, and nutrients in food.

Despite all the previous studies, there is still the need to explore the impact of correlation on model performance to improve our estimations. Through this study, we try to fill the gaps by incorporating correlation in our machine learning models. The results of this study greatly enhanced software development and crafted the best possible models for this study.

3 | Machine Learning in Software Effort Estimation

The prediction of the intensity of efforts for the successful completion of any prominent software project is of utmost importance yet demands a significant amount of thoroughness. In the absence of meticulous attention, it is quite challenging to deliver the project by meeting the requirements.

Over multiple years, experts used a plethora of methods to test the efforts required for software estimation ranging from mere guesses to advanced technologies, but the results from machine learning models were impeccable. Machine learning models have the potential to learn patterns from previous data and make magnificent predictions based on given data. This technique proved to be more efficient than all the traditional practices.

The core objective of any estimation-based technique is to come up with figures that are more close to the real values. The machine learning models provide the experts with better ground to make timely and on-budget deliveries.

In this study, we have utilized the four most important supervised learning models namely linear regression, gradient boosting, random forest, and decision tree. By embedding correlation-based features, we come up with remarkable and precise models for our estimations of healthcare datasets.

4 | Methodology

In this section, we will discuss several key steps that we have taken in our study. Firstly, we will describe the approach of selecting features in our data, as the choice of features contributes substantially to the model performance. Next, we will elaborate on the clinical datasets we have utilized for developing the most exceptional outcomes. We will also outline the steps implemented for fine-tuning our datasets along with machine learning models and performance metrics.

4.1 | Correlation

Correlation is defined as the measurement of the relationship between two variables. The relationship is either positive or negative. There are several methods for calculating correlation, but the most common and popular method is entitled as Pearson correlation coefficient. This method is used for calculating the linear relationship between two variables. When the value of the Pearson correlation coefficient is +1, we say that the correlation is positive or with an increase(decrease) in one variable, another variable also increases(decreases) and if it is -1, it can be stated as negative or with an increase in one variable, other variable decreases. 0 Pearson correlation coefficient denotes that there is no relationship between the variables under study. In our study, while preparing our data, we set the correlation coefficient with a threshold of 0.4.

4.2 | Datasets Used

All of the five datasets used in our research are related to the Healthcare industry and are sourced from various dataset repositories including Kaggle. The detail of individual datasets is given below.

4.2.1 | Breast Cancer Wisconsin

Dr. William H. Wolberg collected the Breast Cancer Wisconsin dataset during his research at the University of Wisconsin in the outsets of the 1990's. It is composed of 569 rows and 33 columns. Each feature depicts the characteristic that has been taken from the image of the nuclei of cells present in breast mass. This dataset is ideal for machine learning project managers who want to apply different models to predict breast cancer and enhance the accuracy of their models. Access the dataset from the link Breast Cancer Wisconsin.

4.2.2 | COVID-19

This dataset is publicly provided by Johns Hopkins University (JHU). The dataset is composed of confirmed cases, number of deaths, and recoveries in different countries and regions from the beginning of this

contagious virus. Every entry is written along with time, presenting an ideal dataset for time-series analysis. You can obtain the dataset from the link COVID-19.

4.2.3 | Sleepy Drivers EEG Brainwave

This dataset is collected by a Kaggle user Nadda Mohamed. The primary goal of this dataset is to examine the effect of exhaustion on driving and engineer the models for tracking the level of fatigue to prevent accidents that are caused by sleep deprivation. It consists of 3735 rows and 11 columns such as attention, meditation, delta, theta, lowAlpha, highAlpha, lowBeta, highBeta, lowGamma, highGamma, and classification. The EEG signals of drivers were taken by the NeuroSky MindWave sensor. One can access the dataset via the link Sleepy Drivers EEG Brainwave.

4.2.4 | Heart Disease Prediction

Heart disease is one of the prominent causes of death in many developed countries. The fundamental purpose of the heart disease prediction dataset is to assist the World by mitigating the risks of heart attack or stroke. It is composed of 12 columns: Age, Sex, ChestPainType, RestingBP, Cholesterol, FastingBS, RestingECG, MaxHR, ExerciseAngina, Oldpeak, ST-Slope, HeartDisease and 918 rows. This dataset presents an excellent opportunity for project managers to establish models that can detect heart diseases in patients so that they receive swift and appropriate care. The dataset is available in the link Heart Disease Prediction.

4.2.5 | Food Nutrition

This data set containing 551 rows and 37 columns presents comprehensive information regarding the nutrients present in food. The main incentive behind collecting this data is to make dietary planning and to do nutritional analysis. The given link provides access to Food Nutrition.

4.3 | Pre-processing Procedures

All of the five datasets used in our research are related to the Healthcare industry and are sourced from various dataset repositories including Kaggle. The detail of individual datasets is given below.

4.3.1 | Handling of Missing Data

Before commencing the analysis of data, it is imperative to clean the data because uncleaned or unprepared data do not give clear and precise insights. One of the pivotal steps in data preparation is handling missing values. It can be done in several ways. One way is to eliminate the missing values from datasets. But it is practicable only if there are more than 70 percent of values absent from a column. Otherwise, the best approach is to fill the column by taking the mean or median of its values. Because by deleting the column we might eliminate the data that is crucial for our prediction and eventually left with a biased model.

4.3.2 | Handling Categorical Features

For mathematical calculation, it is integral to convert categorical features as taking them in training sets without converting resulted in errors. OneHotEncoder and LabelEncoder are used for transforming categorical data into numerical data.

4.3.3 | Data Splitting

Another important step in data preparation is splitting it into training and testing datasets. Normally, the splitting ratio is taken at 80 percent for training and 20 percent for testing. The model is trained by fitting data present in the training set. After checking its performance in the training set, the prediction step is taken by the testing set. For the successful completion of a project, it is important to have a similar performance of the model on both datasets.

4.4 | Techniques Applied

4.4.1 | Linear Regression

Linear regression is the fundamental supervised machine learning model. It expounds the connection between features and labels and then makes the essential predictions.

4.4.2 | Gradient Boosting

One of the most renowned machine learning models that is used for both regression and classification is gradient boosting. The main motive of this model is to combine all the weak models and come up with a strong model having better accuracy [19].

4.4.3 | Random Forest

The random forest machine learning model operates by making multiple trees known as forests. Every tree is made by the subsets of training datasets.

4.4.4 | Decision Tree

Another supervised machine learning model that is used for both regression and classification problems is the decision tree. It functions by breaking datasets into smaller sections and then making trees to predict the targeted vector.

4.5 | Performance Metrics

4.5.1 | R Squared

R squared commonly written as R2 is a performance metric used to track how any regression model is performing. For a perfect model, its value must be 1. In reality, there is no such thing as the perfect model. However, R2 scores close to 1 guarantee the perfect model. The poor model has an R2 value of less than 0. Mathematically, we can write

$$R^2 = 1 - \frac{\sum_{j=1}^{n} \left(z_j - \hat{z}_j\right)^2}{\sum_{j=1}^{n} \left(z_j - \bar{z}_j\right)^2} \tag{1}$$

where z_j are the values that are present in the data, $\hat{z}_j$ are the values that are being predicted by machine learning models, $\bar{z}$ is the mean of the actual values, and n is the number of data points present in datasets.

4.5.2 | Root Mean Squared Error

Root mean squared error (RMSE) is another metric for checking the performance of models. The perfect model has an RMSE near 0. Higher RMSE values delineate that the model that we have made is not close to reality or is a poor model. The mathematical formula for RMSE is given by

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{j=1}^{n} \left(z_j - \hat{z}_j\right)^2} \tag{2}$$

Figure 2, presents the ML algorithm and data pre-processing;

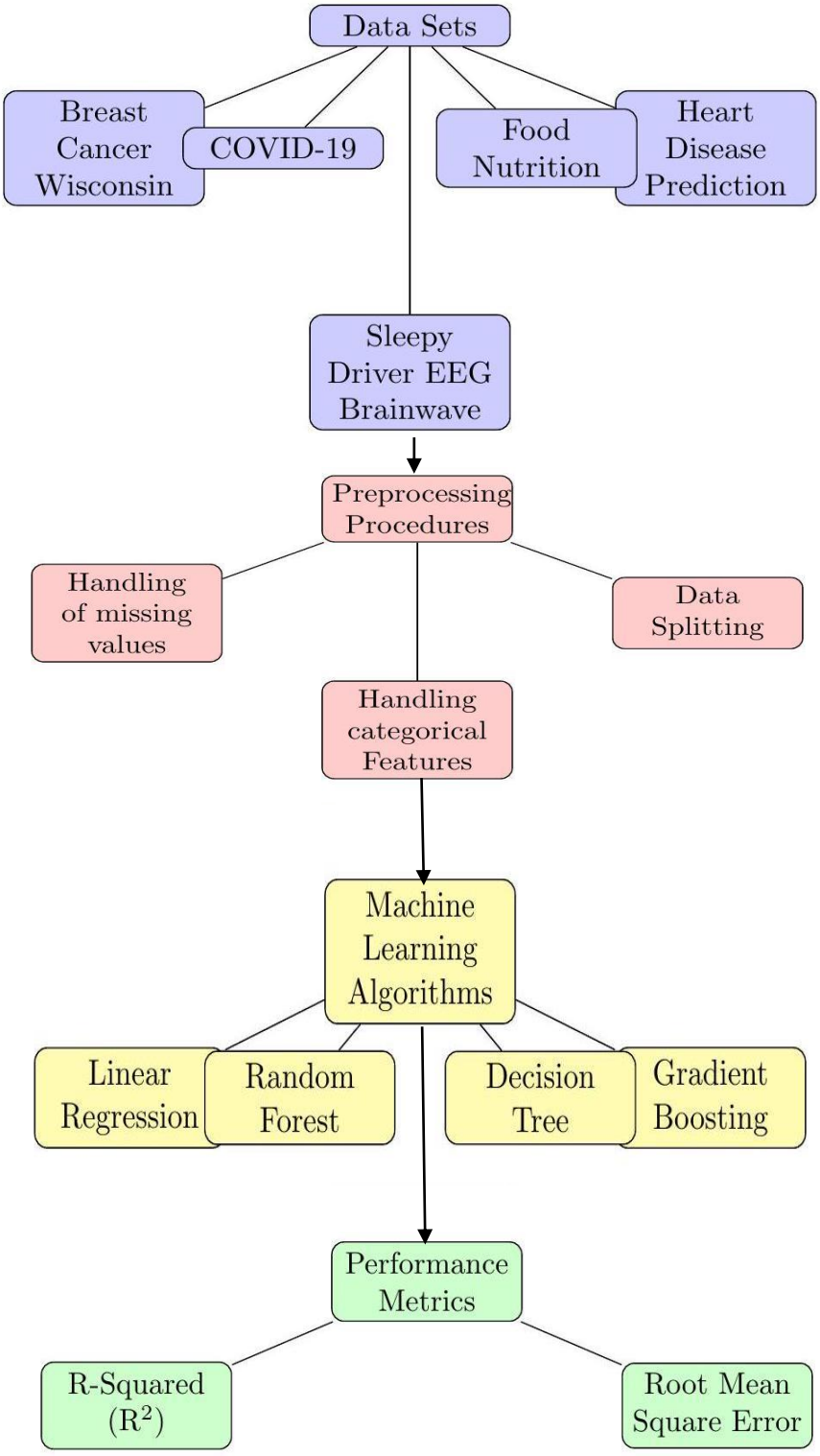

Figure 2. Data collection, pre-processing, and ML techniques algorithms.

5 | Results and Analysis

In this section, we will provide a detailed analysis of our study on software effort estimation by the utilization of machine learning models. All the results are concisely presented in subsections. We will start by detailing the performance of each model in all five datasets through the bar graphs elucidating R-squared scores embedded with correlation. Next, we will expound on the functionality of these models by providing the line graph of the scores of root mean squared error. Alongside these graphs, we will juxtapose the values of

machine learning models with the element of correlation and without correlation and comment on the performance of our algorithms in both of these situations.

5.1 | Breast Cancer Wisconsin

5.1.1 | Graph of R-squared and RMSE

The value of this performance metric ranges from 0 to 1. When our model predicts the value 1 or nearly 1, it means that the model is performing well and the approximations are closer to reality. However, if the values are less than 0, it elucidates that the presented model is poor.

In the case of the dataset of Breast Cancer Wisconsin, gradient boosting and random forest are performing extremely well surpassing the values of the other two models. The gap between the highest and the lowest value is just 0.18 as shown in Figure 3.

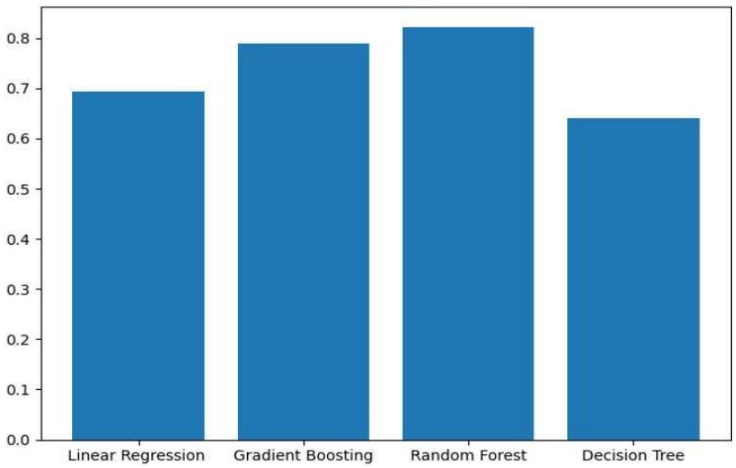

Figure 3. Bar graph of R2 for Breast Cancer Wisconsin.

The root mean squared error is regarded as one of the most widely used performance metrics for keeping track of the accuracy of machine learning models. RMSE scores of 0 delineate that the training model is perfectly fitted and its values are one hundred percent accurate. However, in the real world there is no such thing as perfection, the hundred percent might elucidate the phenomenon of over-fitting. Therefore, it is compulsory to check that the RMSE values in the training and testing set overlap each other. Generally any value close to zero shows perfect model performance.

If we look at the graph in Figure 4, We will see that the scores of RMSE of all the models for the Breast Cancer Wisconsin dataset are approximately equal to each other. However, the lowest values are exhibited by random forest with the figures of 0.21. The closer the values of RMSE to 0, the more accurate prediction will be. Hence, the random forest is doing better than all the other models.

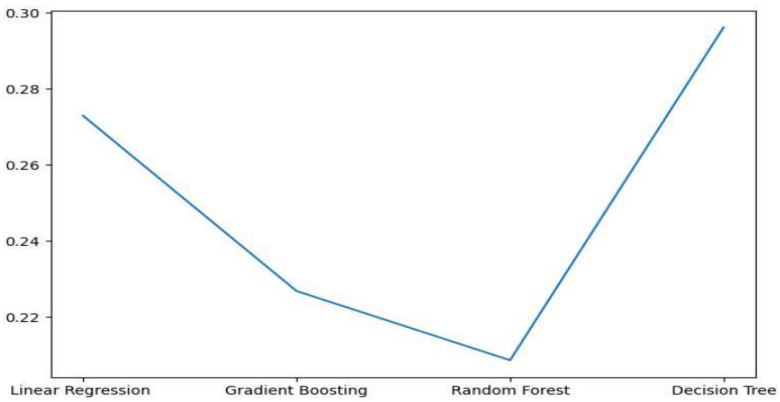

Figure 4. RMSE on the plot for Breast Cancer Wisconsin.

Overall, it is seen from Table 1 of Breast Cancer Wisconsin statistics, R^2 and RMSE scores are highly commendable with correlation than without correlation.

Table 1. Comparison table of Breast Cancer Wisconsin.

Breast Cancer Wisconsin	R^2 score with correlation	R^2 score without correlation	RMSE score with correlation	RMSE score without correlation
Linear Regression	0.75	0.3	0.24	0.40
Random Forest	0.93	0.1	0.13	0.43
Gradient Boosting	0.93	0.2	0.13	0.46
Decision Tree	0.81	-0.3	0.21	0.56

In the correlation table, among all the four supervised machine learning models, gradient boosting and random forest are making impressive predictions for Breast Cancer Wisconsin having an R2 score of 0.93. Linear regression is showing lower scores compared to its counterparts but these are just 0.18 points behind the highest one.

5.2 | COVID-19

5.2.1 | Graph of R-Squared and RMSE

All four models are performing exceptionally well in the dataset of COVID-19, but the highest value is exhibited by linear regression. Gradient boosting, random forest, and decision trees share the same value. The difference between the highest and lowest value is just 0.1 as shown in Figure 5.

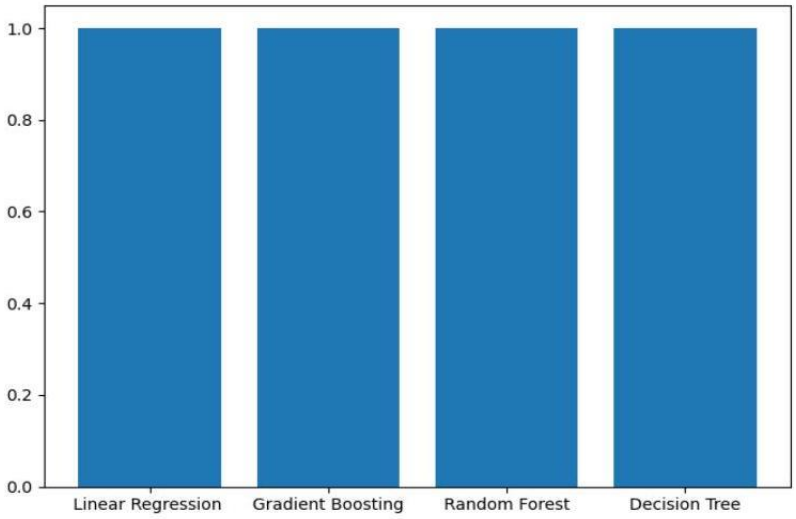

Figure 5. Bar graph of R2 for COVID-19.

The line graph in Figure 6 depicts quite compelling results. Among all the four models, linear regression is performing exceptionally well with the figure of 0.00000000065. The other three models are not elucidating good results or we can say that in comparison with linear regression, random forest, gradient forest, and decision tree are presenting us with poor results. The lowest value is attained by gradient boosting which is around 2318 points higher than that of the linear regression model. Higher values of RMSE illustrate poor performance. Therefore we can say that gradient boosting is not a good choice in the case of the dataset of COVID-19.

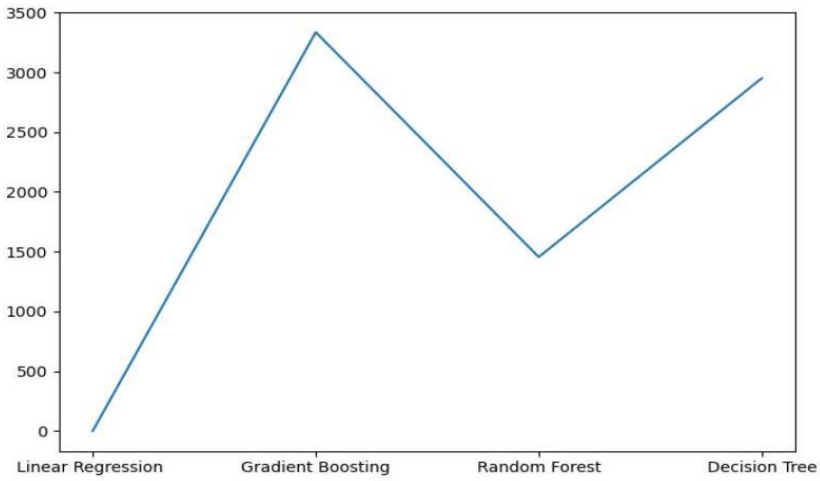

Figure 6. RMSE on plot for COVID-19.

The given Table 2 plainly shows that the correlation greatly improves the performance of our models. There is an evident distinction between correlation-based feature selection and correlation-based selection. Additionally, with correlation, the linear model is best fitted in the dataset of COVID-19. Linear regression bestows us with the perfect model having R2 scores of 1.0, however, the other three models are not far, sharing approximately the same figures of 0.99.

Table 2. Comparison table of COVID-19.

COVID-19	R^2 score with correlation	R^2 score without correlation	RMSE score with correlation	RMSE score without correlation
Linear Regression	1.0	0.0088	0.00000000065	137331.59
Random Forest	0.99	0.48	3218.67	99099.86
Gradient Boosting	0.99	0.49	1455.14	98396.35
Decision Tree	0.99	0.48	2950.98	98670.70

5.3 | Sleepy Drivers EEG Brainwave

5.3.1 | Graph of R-squared and RMSE

The gradient boosting model is performing better in the dataset of Sleepy Drivers EEG Brainwave than its counterparts, whereas the decision tree is giving poor scores. The difference between the scores of different models is clearly shown in Figure 7.

The RMSE values of the dataset of Sleepy Drivers EEG Brainwave are shown in the line in Figure 8. The lowest value of RMSE is presented by the model of gradient boosting ensuring the better model in comparison with the other one. The highest points are manifested by decisions depicting less precision in comparison with their counterparts.

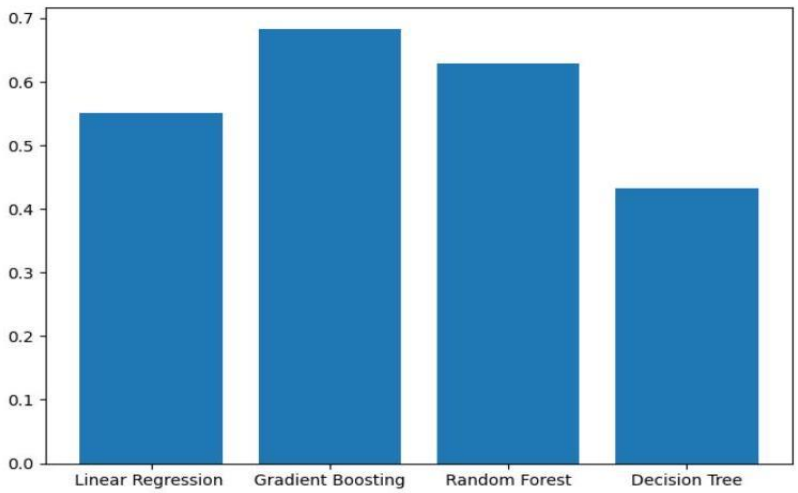

Figure 7. Bar graph of R2 for Sleepy Drivers EEG Brainwave.

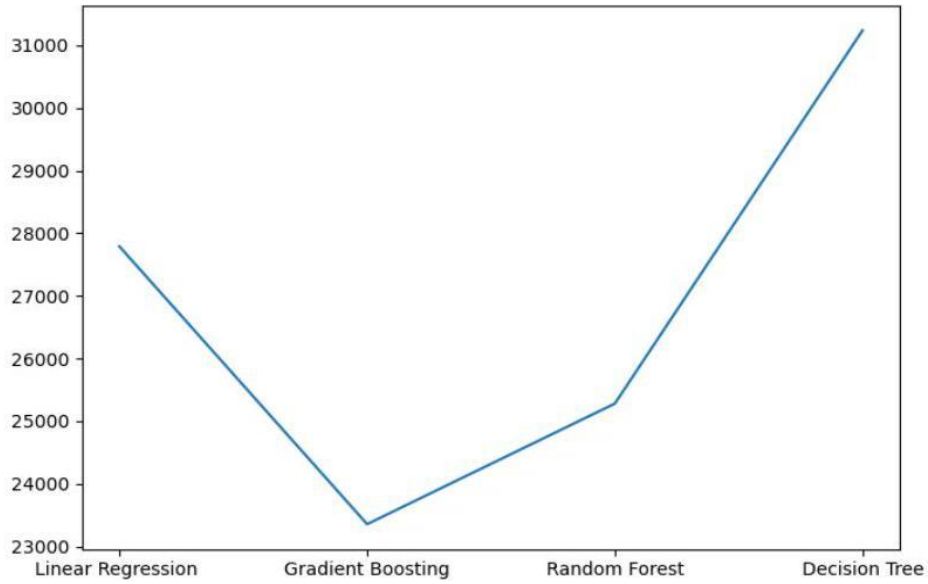

Figure 8. RMSE on the plot for Sleepy Drivers EEG Brainwave.

The table below patently suggests the incorporation of correlation in project management for the successful completion of software-based projects.

Table 3 illustrates that among all the models, random forest is performing quite well in most of the datasets. The second best performer in nearly all datasets is gradient boosting and then random forest. These values are in the favor that the decision tree might not be the astute selection when we are swamped with the constraints of time and costs.

Table 3. Comparison table of Sleepy Drivers EEG Brainwave.

Sleepy Drivers EEG Brainwave	**R^2 score with correlation**	**R^2 score without correlation**	**RMSE score with correlation**	**RMSE score without correlation**
Linear Regression	0.55	0.43	27790.09	31336.22
Random Forest	0.69	0.47	23017.18	30314.71
Gradient Boosting	0.62	0.44	25421.14	30989.59
Decision Tree	0.55	0.17	27927.86	37738.66

5.4 | Heart Disease Prediction

5.4.1 | Graph of R-Squared and RMSE

The figures of random forest are magnificent for Heart Disease Prediction Heart Disease Prediction. But, the values of the decision tree are exceptionally poor. We can witness the performance of random forest in Figure 9. This clear difference indicates the potential of random forest compared to all the other models.

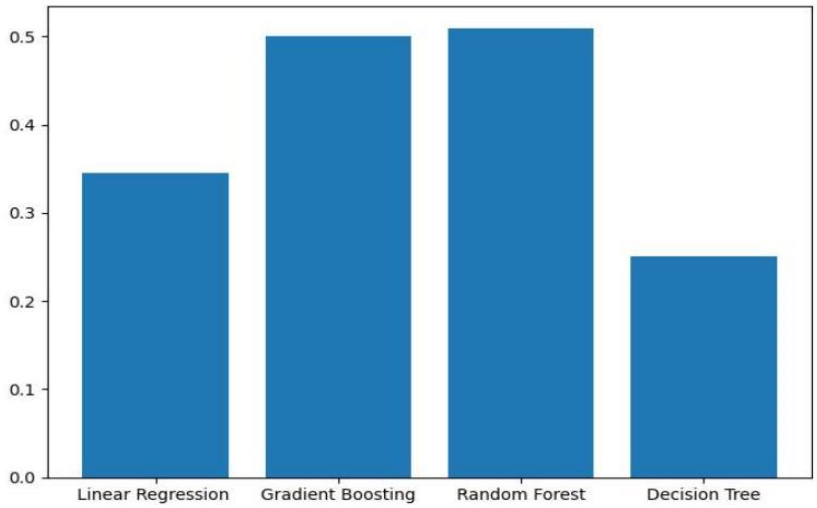

Figure 9. Bar graph of R2 for Heart Disease Prediction.

The four supervised machine learning models are giving equally eminent results with quite a minute difference in their values. The lowest RMSE value is possessed by gradient boosting and random forest models, evidently showing excellent performance. The RMSE values for linear regression and decision tree are 0.4 and 0.42, which are also quite close to 0. The difference between these values and the lowest is just 0.05 and 0.07, which is not very high. Therefore, the software managers are allowed to choose any of these models. However, the best choice will surely be to choose among gradient boosting and random forest. The line graph illustrating the RMSE for heart disease Prediction is shown in Figure 10.

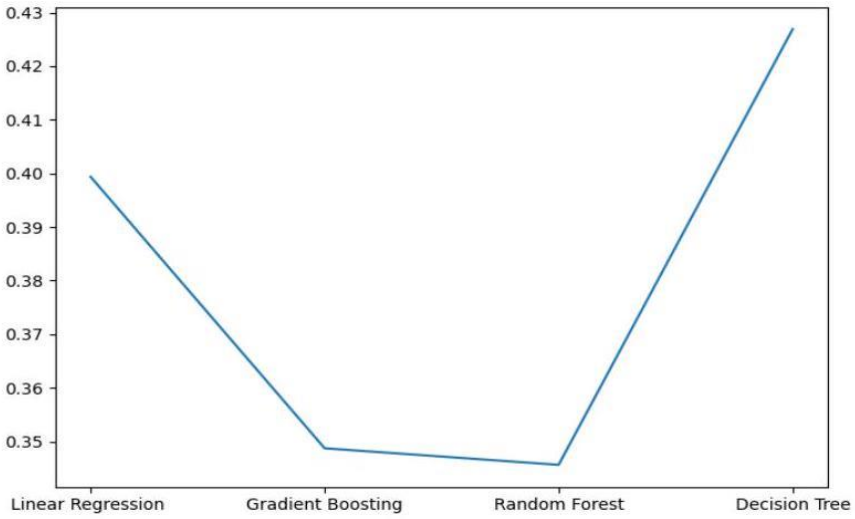

Figure 10. RMSE on the plot for Heart Disease Prediction.

When it comes to the dataset of Heart Disease Prediction, gradient boosting and random forest are enacting comparatively better with R2 values of 0.50 and 0.51 respectively.

Table 4. Comparison table of Heart Disease Prediction.

Heart Disease Prediction	R^2 score with correlation	R^2 score without correlation	RMSE score with correlation	RMSE score without correlation
Linear Regression	0.34	0.28	0.39	0.42
Random Forest	0.5	0.31	0.35	0.40
Gradient Boosting	0.51	0.17	0.35	0.45
Decision Tree	0.25	0.08	0.43	0.47

5.5 | Food Nutrition

5.5.1 | Graph of R-Squared and RMSE

The linear regression model is setting up the stage with remarkable scores for the dataset of Food Nutrition. However, there is no clear difference between the figures of linear regression, gradient boosting, and random forest as depicted in Figure 11. The lowest value is achieved by the decision tree with only 0.15 from the highest one.

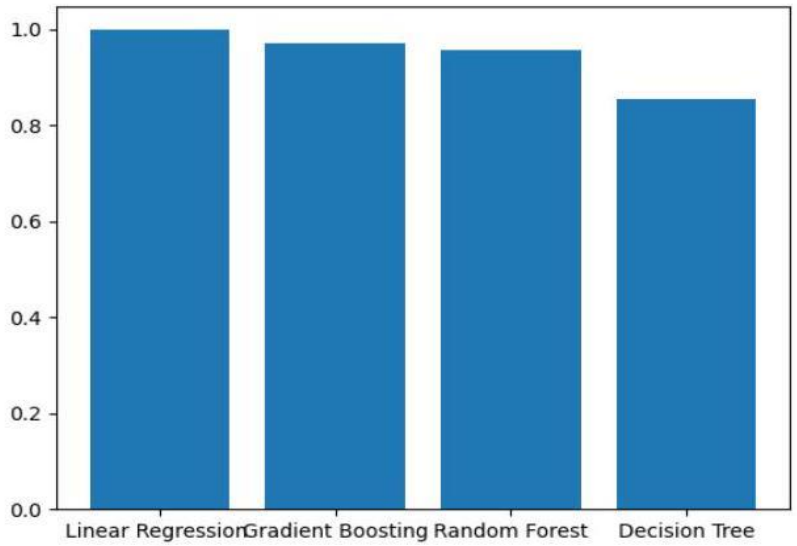

Figure 11. Bar graph of R2 for Food Nutrition.

The RMSE score of each of the models on the dataset of Food Nutrition is moderate. The lowest value in the case of this dataset is 6.19 manifested by the linear regression model. Hence, we can say that the linear regression model is performing well here as shown in Figure 12. The highest RMSE value is achieved by the decision tree which delineates that this model is although not performing poorly, the managers have the availability of other good options like linear regression and gradient boosting.

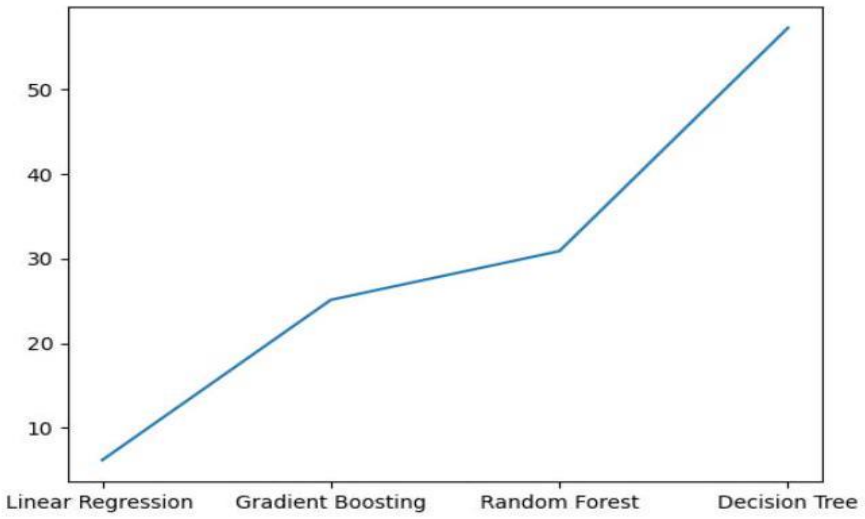

Figure 12. RMSE on the plot for Food Nutrition.

All four models fit perfectly well in the dataset of Food Nutrition, but linear regression is dominating with 0.99 points on the points table as shown in Table 5.

Table 5. Comparison table of Food Nutrition.

Food Nutrition	R^2 score with correlation	R^2 score without correlation	RMSE score with correlation	RMSE score without correlation
Linear Regression	0.99	0.39	6.2	117.1
Random Forest	0.97	0.82	25.1	63.2
Gradient Boosting	0.96	0.83	30.9	62.4
Decision Tree	0.85	0.63	57.3	90.7

6 | Discussion

The performance of four supervised machine learning models: linear regression, gradient boosting, and decision tree is being observed in five different software projects. The main observation is that choosing correlation-based features in the feature matrix greatly enhanced the performance of projects, the scores of RMSE and R2 better reflect it. Additionally, linear regression, gradient boosting and random forest are delivering finer outputs as compared to others. Our results approve the previous studies about the choice of models, however, we also observed that the choice of correlated features is also a big deal for the successful accomplishments of any software-based analysis.

Recent advancements in fuzzy systems and machine learning have tackled uncertainty across diverse fields. It includes T-S fuzzy in time delay [42], and fuzzy decision-making method for wastewater treatment [43]. The AI is being widely used in accessing education. The use of AI methods in political education [44] and deep learning to assess English material readability [45]. The study of enhanced trade risk assessment using edge computing by [46] and the use of CNNs for text readability has been done by [47]. The method of iris detection for pandemic attendance systems was introduced by [48].

Future studies could explore the impact of correlation-based feature selection in more complex domains by integrating fuzzy systems or AI-driven techniques for feature engineering. The scalability of models like gradient boosting and random forests could be tested across diverse projects, while advanced

neural networks like CNNs may enhance feature interaction analysis. Hybrid approaches combining machine learning with fuzzy logic could offer robust, interpretable solutions for software-based analyses.

7 | Conclusions

In conclusion, the discoveries of this analysis appended the foremost endowment towards the innovation of software estimation with the incorporation of machine learning models. The preference of features especially exhibiting strong correlation is peremptory for efficacious consummation of any software appraisal. The sturdy and tenacious achievements of linear regression, gradient boosting, and random forest proved by noteworthy scores of R2 and RMSE showcases the triumph of our approach. This highly reputable accuracy of our prediction can encourage the project managers to take action according to time and costs. Since all of the techniques are being imposed on real-time datasets which outline the robustness of our analysis in actual projects primarily for the healthcare industry. This study adroitly formulates the function of machine learning models enriched by correlation-based feature selection. The novel strategy of predictability by the machine learning models offers more streamlined sources for projects. Ultimately, this research alleviates all the inefficiencies and provides a more concrete ground for software application managers.

Acknowledgments

The author is grateful to the editorial and reviewers, as well as the correspondent author, who offered assistance in the form of advice, assessment, and checking during the study period.

Author Contributions

All authors contributed equally to this work.

Funding

This research was conducted without external funding support.

Data Availability

The datasets generated during and/or analyzed during the current study are not publicly available due to the privacy-preserving nature of the data but are available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declares that there is no conflict of interest in the research.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

References

[1] Liu, J., Du, Q., Xu, J. (2018). A learning-based adjustment model with genetic algorithm of function point estimation.In 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), Exeter, UK, pp. 51-58.

[2] Moharreri, K., Sapre, A.V., Ramanathan, J., Ramnath, R. (2016). Cost-effective supervised learning models for software effort estimation in agile environments. In 2016 IEEE 40th Annual computer software and applications conference (COMPSAC), Atlanta, GA, USA, pp. 135140.

[3] Shah, M.A., Jawawi, D.N.A., Isa, M.A., Younas, M., Abdelmaboud, A., Sholichin, F. (2020). Ensembling artificial bee colony with analogy-based estimation to improve software development effort prediction. IEEE Access, 8: 58402-58415.
[4] Street, W. N., Thorne, N. L.,& Wolberg, C. M. (1992). Breast Cancer Wisconsin dataset. https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data.
[5] Gardner, L., Sauber, K. B., Wong, C. R. D. R., & Lee, J. F. H. (2020). COVID-19 dataset. https://www.kaggle.com/datasets/imdevskp/corona-virus-report
[6] Mohammed, N. (2002). Sleepy Drivers EEG Brainwave. https://www.kaggle.com/datasets/naddamuhhamed/sleepy-driver-eeg-brainwave-data.
[7] Aha, D. W., Baker, L., & Kauffman, J. J. (1991). Heart disease dataset. UCI Machine Learning Repository.
[8] Dey, U. (n.d). Food Nutrition dataset. https://www.kaggle.com/datasets/utsavdey1410/food-nutrition-dataset.
[9] Goyal, S. (2022). Effective software effort estimation using heterogenous stacked ensemble. In 2022 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES), Thiruvananthapuram, India, pp. 584-588.
[10] Mahmood, Y., Kama, N., Azmi, A., Khan, A.S., Ali, M. (2022). Software effort estimation accuracy prediction of machine learning techniques: A systematic performance evaluation. Software: Practice and Experience, 52(1): 39-65.
[11] Setiadi, A., Hidayat, W.F., Sinnun, A., Setiawan, A., Faisal, M., Alamsyah, D.P. (2021). Analyze the datasets of software effort estimation with particle swarm optimization. In 2021 International Seminar on Intelligent Technology and Its Applications (ISITIA), Surabaya, Indonesia, pp. 197-201.
[12] Idri, A., Khoshgoftaar, T.M., Abran, A. (2002). Can neural networks be easily interpreted in software cost estimation? Fuzzy Sets and Systems, 132(2): 225-236.
[13] Sarro, F., Petrozziello, A., Harman, M. (2016). Multiobjective software effort estimation. In Proceedings of the 38th International Conference on Software Engineering, pp. 619-630.
[14] Mittas, N., Angelis, L. (2008). Comparing cost prediction models by resampling techniques. Journal of Systems and Software, 81(6): 816-824.
[15] Jeon, H., Oh, S. (2020). Hybrid-recursive feature elimination for efficient feature selection. Applied Sciences, 10(9): 3211.
[16] Liu, H., Motoda, H. (1998). Feature extraction, construction and selection: A data mining perspective. The Springer International Series in Engineering and Computer Science.
[17] Guyon, I., Elisseeff, A. (2003). An introduction to variable and feature selection. Journal of Machine Learning Research, 3: 1157-1182.
[18] Bishop, C. M. (2006). Pattern recognition and machine learning. Springer.
[19] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT Press.
[20] Abulalqader, F.A., Ali, A.W. (2018). Comparing different estimation methods for software effort. In 2018 1st annual international conference on information and sciences (AiCIS), Fallujah, Iraq, pp. 13-22.
[21] Fukui, S., Monden, A., Yücel, Z. (2018). Kurtosis and skewness adjustment for software effort estimation. In 2018 25th Asia-Pacific Software Engineering Conference (APSEC), Nara, Japan, pp. 504-511.
[22] Khoshgoftaar, T.M., Yuan, X., Allen, E.B., Jones, W.D., Hudepohl, J.P. (2000). Uncertain classification of fault prone software modules. Empirical software Engineering. 5(4): pp.297-318.
[23] Siham, A., Sara, S., Abdellah, A. (2021). Feature selection based on machine learning for credit scoring: An evaluation of filter and embedded methods. In 2021 International Conference on INnovations in Intelligent Systems and Applications (INISTA), Kocaeli, Turkey, pp.1-6.
[24] Meharunnisa, M., Saqlain, M., Abid, M., Awais, M., & Stević, Ž. (2023). Analysis of Software Effort Estimation by Machine Learning Techniques. Ingénierie des Systèmes d'Information, 28(6), pp. 1445-1457. http://iieta.org/journals/isi.
[25] Zhang, X., Zheng, Y., Jiang, Z., & Byun, H. (2023). Numerical algorithms for corner-modified symmetric Toeplitz linear system with applications to image encryption and decryption. Journal of Applied Mathematics and Computing, 69(2), 1967-1987.
[26] Bao, J., Fu, L., Zhang, H., Zhang, A., Guo, W., & Chen, T. (2022). An adaptive proportional plus damping control for teleoperation systems with asymmetric time-varying communication delays. Mathematics, 10(24), 4675.
[27] Abid, M., & Shahid, M. (2024). Data-driven evaluation of background radiation safety using machine learning and statistical analysis.Big Data and Computing Visions, 4(2), 110-134. doi: 10.22105/bdcv.2024.476542.1186.
[28] Zhang, X., Zheng, Y., Jiang, Z., & Byun, H. (2024). Efficient algorithms for real symmetric Toeplitz linear system with low-rank perturbations and its applications. Journal of Applied Analysis & Computation, 14(1), 106-118.
[29] Xu, M., Liu, S., & Lou, Y. (2024). Persistence and extinction in the antisymmetric Lotka-Volterra systems. Journal of Differential Equations, 387, 299323.
[30] Zhang, X., Zheng, Y., & Jiang, Z. (2024). Fast algorithms for the solution of perturbed symmetric Toeplitz linear system and its applications. Computational and Applied Mathematics, 43(4), 252.
[31] Zhang, S., Hou, Y., Zhang, S., & Zhang, M. (2017). Fuzzy control model and simulation for nonlinear supply chain system with lead times. Complexity, 2017(1), 2017634. (8). 2.
[32] Hamid, M. T., & Abid, M. (2024). Decision Support System for Mobile Phone Selection Utilizing Fuzzy Hypersoft Sets and Machine Learning. Journal of Intelligent Management Decisions, 3(2), 104-115.
[33] Zhang, S., Zhang, P., & Zhang, M. (2019). Fuzzy emergency model and robust emergency strategy of supply chain system under random supply disruptions. Complexity, 2019(1), 3092514.

[34] Ge, J., & Zhang, S. (2020). Adaptive inventory control based on fuzzy neural network under uncertain environment. Complexity, 2020(1), 6190936.
[35] Ullah, W., Siddique, I., Zulqarnain, R. M., Alam, M. M., Ahmad, I., & Raza, U. A. (2021). Classification of arrhythmia in heartbeat detection using deep learning. Computational Intelligence and Neuroscience, 2021(1), 2195922.
[36] Abid, M., & Shahid, M. (2024). Tumor Detection in MRI Data using Deep Learning Techniques for Image Classification and Semantic Segmentation. Sustainable Machine Intelligence Journal, 9, 1-13. https://doi.org/10.61356/SMIJ.2024.9380.
[37] Mahboob, A., Asif, M., Zulqarnain, R. M., Siddique, I., Ahmad, H., Askar, S. S., & Pau, G. (2023). An Innovative Technique for Constructing Highly Non-Linear Components of Block Cipher for Data Security against Cyber Attacks. Comput. Syst. Sci. Eng.,47(2), 2547-2562.
[38] Asif, M., Mairaj, S., Saeed, Z., Ashraf, M. U., Jambi, K., & Zulqarnain, R. M. (2021). A novel image encryption technique based on mobius transformation. Computational Intelligence and Neuroscience, 2021(1), 1912859.
[39] Abidin, M. Z., Marwan, M., Ullah, N., & Mohamed Zidan, A. (2023). WellPosedness in Variable-Exponent Function Spaces for the Three-Dimensional Micropolar Fluid Equations. Journal of Mathematics, 2023(1), 4083997.
[40] Marwan, M., Han, M., Dai, Y., & Cai, M. (2024). The Impact Of Global Dynamics On The Fractals Of A Quadrotor Unmanned Aerial Vehicle (Quav) Chaotic System. Fractals, 32(02), 2450043.
[41] Mahboob, A., Asif, M., Zulqarnain, R. M., Saddique, I., Ahmad, H., & Askar, S. (2023). A Mathematical Approach for Generating a Highly Non-Linear Substitution Box Using Quadratic Fractional Transformation. Computers, Materials 8 Continua, 77(2).
[42] Gao, M., Zhang, L., Qi, W., Cao, J., Cheng, J., Kao, Y., Wei, Y., & Yan, X. (2020). SMC for semi-Markov jump T-S fuzzy systems with time delay. Applied Mathematics and Computation, 374, 125001.
[43] Saeed, M., Kareem, K., Razzaq, F., & Saqlain, M. (2024). Unveiling Efficiency: Investigating Distance Measures in Wastewater Treatment Using Interval-Valued Neutrosophic Fuzzy Soft Set. Neutrosophic Systems with Applications, 15, 1-15.
[44] Saqlain, M. (2023). Revolutionizing Political Education in Pakistan: An AI-Integrated Approach. Education Science Management, 1(3), 122-131.
[45] Saqlain, M. (2023). Evaluating the Readability of English Instructional Materials in Pakistani Universities: A Deep Learning and Statistical Approach. Education Science Management, 1(2), 101-110.
[46] Abid, M. & Saqlain, M. (2023). Utilizing Edge Cloud Computing and Deep Learning for Enhanced Risk Assessment in China's International Trade and Investment. International Journal of Knowledge and Innovation Studies, 1(1), 1-9.
[47] Zulqarnain, M. & Saqlain, M. (2023). Text Readability Evaluation in Higher Education Using CNNs, Journal of Industrial Intelligence, 1(3), 184-193.
[48] Haq, H. B. U. & Saqlain, M. (2023). Iris Detection for Attendance Monitoring in Educational Institutes Amidst a Pandemic: A Machine Learning Approach, Journal of Industrial Intelligence, 1(3), 136-147.