ICIC Express Letters
Part B: Applications
Volume **16**, Number **7**, July **2025**

ICIC International ©2025 ISSN 2185-2766
pp. 709–716

THE NUMBER OF PATIENTS PREDICTION USING DATA MIGRATION, MACHINE LEARNING AND MODIFIED PARTICLE SWARM OPTIMIZATION

Somkiat Kosolsombat[1], Chiabwoot Ratanavilisagul[2]
and Taratep SiraAksorn[3]

[1]Data Science and Innovation
[3]Interdisciplinary Center of Excellence in Data Science and Innovation for Thai Traditional and Alternative Medicine
College of Interdisciplinary Studies
Thammasat University
2 Prachan Road, Phra Barom Maha Rathawang, Phra Nakhon, Bangkok 10200, Thailand
somkiatk@tu.ac.th; taratep.sira@sci.tu.ac.th

[2]Department of Computer and Information Sciences
Faculty of Applied Science
King Mongkut's University of Technology North Bangkok (KMUTNB)
1518 Pracharat 1 Road, Wongsawang, Bangsue, Bangkok 10800, Thailand
chaibwoot.r@sci.kmutnb.ac.th

Received August 2024; accepted November 2024

Abstract. *This research aims to develop a model for predicting the number of patients. The number of patients is crucial information as it can be used for budget planning in the next year. Researchers aim to develop a model to provide more accurate predictions using machine learning techniques and Particle Swarm Optimization (PSO) to adjust parameters. There are many available machine learning techniques, each with different prediction performances depending on the problem being solved. Therefore, this research proposes using hybrid machine learning techniques to create a better model. Additionally, PSO encounters the problem of trapping in local optima. To reduce and address this problem, this research proposes adding a re-initialization technique into the PSO process. The proposed technique can yield higher accuracy and was tested on two standard datasets, the heart disease dataset (heart dataset) and the boston housing dataset (housing dataset), as well as the number of patient datasets (patient dataset), which can be used in practice. The experimental results show that the proposed technique provided more satisfactory results across all datasets compared to other techniques.*
Keywords: Particle swarm optimization, Support vector machine, Extreme gradient boosting, Trapping in local optimum, Machine learning, Data migration

1. Introduction.

The National Health Security Office (NHSO) Thailand is a government agency that provides healthcare services to all Thai citizens. NHSO needs to plan the budget allocation for each hospital, which requires knowing the number of patients in each hospital. The number of patients can be predicted using machine learning, a widely used technology for prediction [1-13]. Machine learning requires past data on the number of patients at each hospital in order to predict future patient numbers.

The data migration problem involves creating a plan to move data objects stored on network devices from one configuration to another [14]. For the data of the National Health Security Office of Thailand, the existing system uses the COGNOS program and Oracle database. This setup cannot be used for data analysis and prediction. Additionally, searching and displaying data from Big Data [15] are slow. In order to enable data

DOI: 10.24507/icicelb.16.07.709

analysis and prediction, as well as faster searching and displaying data from Big Data, this research necessitates a Data Migration using Apache Hive. To measure the efficiency of algorithms, two standard datasets were used and another dataset from the prediction of the number of patients from NHSO. The first dataset is heart dataset [16]. This dataset is used to compare the classification performance of algorithms. The secondary dataset is housing dataset [17] and the third dataset is patient dataset used to compare the regression performance of algorithms.

Many researchers [1-7] have modified or hybridized techniques such as Support Vector Machine (SVM), Gradient Boosting, Random Forest (RF), Naïve Bayes, Logistic Regression (LR), and K-Nearest Neighbors (KNN) to improve recognition rates in solving classification problems. Many researchers [8-11] have modified or hybridized techniques such as SVM, PSO, and Genetic Algorithms to predict various factors like market trends, stock prices, rainfall forecasting, and node number prediction. To improve recognition rates or prediction accuracy, several researchers [12,13] have recently proposed integrating PSO into Extreme Gradient Boosting (XGBoost) or SVM. Experimental results show that PSO can enhance model performance. Majumder et al. [13] proposed using PSO to adjust parameters between XGBoost and SVM to enhance the performance model. This algorithm is referred to as a hybrid model using XGBoost and SVM with parameters adjusted by PSO (XSPSO). XSPSO can be easily implemented, applied to other similar problems, and further developed. Therefore, this research aims to further develop XSPSO to improve its efficiency.

PSO is a stochastic global optimization technique used to solve optimization problems with the primary goal of finding the global optimum of a fitness function [18]. It offers several advantages, such as fast convergence, simplicity, and having only a few parameters to adjust. However, it has the drawback of an increased risk of becoming trapped in local optima [19-21]. This research proposes the re-initialization technique to address the issue of trapping in local optima. Additionally, the XSPSO process uses PSO to adjust parameters and select the optimal results between XGBoost and SVM. This approach is effective because PSO performs well in choosing optimal results. Moreover, this research proposes incorporating more prediction techniques, such as KNN, LR, and RF, and using PSO to optimize their parameters. This will result in a more efficient model with improved accuracy. The proposed technique was tested on the heart dataset, housing dataset, and patient dataset, providing more satisfactory results compared to other methods.

The remainder of this document is structured as outlined below. Section 2 explains the background. Section 3 explains the proposed method. Section 4 explains the experiment results. Section 5 concludes this research and directions for development of this research in future work.

2. Backgrounds.

2.1. Particle swarm optimization.

Each member of the particle population is known as a "particle" or "X", possessing its distinct position and velocity. These particles navigate the search space based on their velocities, the best position discovered among all particles (GBEST), and their individual best position (PBEST). PSO starts by randomly initializing particle positions and velocities, evaluating each particle's position based on the objective function of the optimization problem. In each generation, every particle updates its position and velocity using the formula below:

$$V'_{i,d} = \varpi V_{i,d} + \eta_1 \text{rand}()(P_{i,d} - X_{i,d}) + \eta_2 \text{rand}()(G_{i,d} - X_{i,d}) \tag{1}$$

$$X'_{i,d} = X_{i,d} + V'_{i,d} \tag{2}$$

where $X_{i,d}$ is the previous positions of the i-th particle in the d-th dimension, $X'_{i,d}$ is the current positions of the i-th particle in the d-th dimension, $V_{i,d}$ is the previous velocity of the i-th particle in the d-th dimension, $V'_{i,d}$ is the current velocity of the i-th particle in the d-th dimension, $P_{i,d}$ is the individual best position (PBEST) of the i-th particle in the d-th dimension, and $G_{i,d}$ is the global best position (GBEST) in the d-th dimension. ϖ is the inertia weight, η_1 and η_2 are acceleration constants. rand() denotes a uniformly generated random number within the range of $[0,1]$.

2.2. **LASSO.** The Least Absolute Shrinkage and Selection Operator Regression (LASSO) is a significant method in regression analysis for both variable selection and regularization. It removes unimportant features, helping to prevent overfitting, and shrinks the coefficients of less influential features towards zero, making them easier to identify. The main idea of LASSO can be simply explained as follows: LASSO regression adds a penalty to the standard regression formula, based on the absolute values of the model's coefficients. LASSO consists of two key components: the least squares term (which measures how well the model fits the data) and the penalty term (which encourages some coefficients to become smaller). LASSO can shrink some coefficients exactly to zero, effectively ignoring unimportant features. This makes LASSO a valuable tool for feature selection by focusing on the most significant features [13,22,23].

2.3. **XGBoost, SVM, LR, RF, KNN.** XGBoost uses gradient-boosted decision trees [24]. The benefits of it include high performance, suitability for large datasets, easy parameter adjustments, and effective handling of missing values. The drawbacks of it are that it can be computationally intensive, memory-heavy, and time-consuming. It is also prone to overfitting, especially with small datasets. SVM aims to find the optimal hyperplane in an N-dimensional space that separates data points into different classes. The benefits of it include its effectiveness with non-linear data and high-dimensional feature spaces, its ability to separate classes well, and its suitability for moderately sized datasets. The drawbacks of it include limited interpretability, and the requirement for careful parameter adjustment [25]. LR aims to predict the probability of an instance belonging to a specific class. It analyzes the relationship between two data factors and involves various types and implementations. The benefits of it include wide usage, and good accuracy. The drawbacks of it include difficulty with complex patterns, suitability for small datasets, and limitation to linear relationships [25]. RF creates multiple decision trees using random subsets of data and features. The benefits of it are preventing overfitting, robustness and accuracy. The drawbacks include being suitable for medium-sized datasets, computational intensity, and complexity [25]. KNN uses the concept of making predictions based on the similarity of data points. The benefits are that it requires no training period and is easy to implement. The drawbacks are that it does not work well with large datasets and is sensitive to noisy and missing data [26].

2.4. **XSPSO.** Its core concept of XSPSO is detailed as follows. Initially, feature selection is performed using LASSO, selecting only the features with result values from LASSO calculations that are greater than the threshold value. In the next step, the data is divided into two groups: one for training and one for testing. The training data is then used to train both XGBoost and SVM to create two models. These two models are merged together, as explained in the formula below:

$$f_{merge}(x) = w1 \times \mathrm{SVM}(x) + w2 \times \mathrm{XGBoost}(x) \tag{3}$$

Subsequently, PSO is used to adjust the weights, $w1$ and $w2$, to achieve the best results, with the condition that $w1 + w2$ equals 1. The merged model or hybrid model with the adjusted parameters is then applied to the test data to measure performance. The development of these techniques can be easily implemented because all techniques can use the scikit-learn library, which is a standard library. The hybrid model with the adjusted weights yields better results. This method is a good idea because PSO is highly effective for solving optimization problems. When the hybrid model is applied in practice, it results in higher accuracy.

3. **Proposal Works.** As mentioned earlier, the XSPSO method can create an effective model. However, it can be further developed to improve the efficiency of model creation by incorporating additional machine learning techniques, such as LR, KNN, and RF, into the hybrid model as described previously. Each algorithm has different strengths in solving problems. For example, XGBoost may perform well on the heart dataset, while RF might handle the housing dataset better. Therefore, adding other algorithms as options for model building can enhance performance. In practice, we do not know which algorithm is better for solving a problem or adjusting the parameters is suitable for solving the problem. This issue can be addressed by applying PSO to automatically adjust parameters and create a better model. Hence, this research proposes integrating LR, RF, and KNN into the hybrid model and using PSO to adjust parameters, as shown in the formula below and this method is called that hybrid model using XGBoost, SVM, LF, RF, and KNN with parameters adjusted by PSO or XSLRKPSO.

$$\begin{aligned} f_{merge}(x) = {} & w1 \times \mathrm{SVM}(x) + w2 \times \mathrm{XGBoost}(x) + w3 \times \mathrm{LR}(x) + w4 \times \mathrm{RF}(x) \\ & + w5 \times \mathrm{KNN}(x) \end{aligned} \tag{4}$$

The application of PSO for XSLRKPSO requires some changes as follows: $w1$, $w2$, $w3$, $w4$, and $w5$ are the dimensions of the particle, which must sum up to 1.0. So, $0 \leq X_{i,d} \leq 1$ and $0 \leq V_{i,d} \leq 1$. The Root Mean Square Error (RMSE) is the fitness equation to RMSE used with XSLRKPSO and XSPSO. It needs to be modified as follows:

$$\text{Fitness(RMSE)} = \sqrt{\frac{\sum_{j=1}^{M} \left[\sum_{d=1}^{D} \left(\left|Y_j - \bar{Y}_j\right| \times X_{i,d}\right)\right]^2}{M}} \tag{5}$$

where Y_j denotes predicted values, and $\bar{Y}_j$ denotes actual values. M is the number of train data. D is the number of dimensions. As mentioned earlier, when the swarm becomes trapped in a local optimum, the solution stagnates until the search is completed. This scenario indicates the end of the search process in PSO. Both XSLRKPSO and XSPSO lack mechanisms for addressing being trapped in a local optimum. To solve this problem, this research introduces re-initialization. The process involves re-initializing velocities, positions, PBEST of all particles, and GBEST when local optimum trapping occurs. For re-initialization, instead of using random starting values for each particle, the initial value is set to 1 for the model (XGBoost, SVM, LR, RF, and KNN) that achieved the best results from the training step, and other models are set to 0. This ensures that the parameters adjusted by PSO do not worsen the model before merging. An indicative sign of being trapped in a local optimum is when the number of consecutive GBEST values remaining unchanged exceeds a threshold value. This threshold is referred to as the Re-initialization Period (RP). RP is easily adjustable and straightforward to conceptualize during fine-tuning [21]. This research proposed adding re-initialization into the PSO process of both XSPSO (XSPSOR) and XSLRKPSO (XSLRKPSOR). This paper introduces XSLRKPSOR to achieve better search results in the training phase and consequently improve outcomes in the test phase or practical usage. The pseudo code for XSLRKPSOR is provided in Figure 1.

In the context of these parameters: N represents the total number of particles, $F(X_i)$ corresponds to the fitness of the 'i' particle, $F(\mathrm{PBEST}_i)$ is the fitness derived from PBEST for the 'i' particle, PBEST_i designates the PBEST of the 'i' particle, $F(\mathrm{GBEST})$ signifies the fitness of the global best (GBEST), CGBEST denotes the consecutive number of unchanged GBEST values, TGBEST is the temporary fitness used to monitor changes in GBEST, and MAXGEN is the maximum number of generations.

```
Using feature selection with LASSO
Apply XGBoost, SVM, LR, RF, KNN
The best index = Select index from the best result from RMSE of XGBoost, SVM, LR,
RF, KNN
Initial velocity of each particle, PBEST, GBEST, Initial position set the best index of
position = 1 and other position set 0
For gen = 0 to MAXGEN
    For i = 0 to N
        Evaluate particle fitness according to Formula (5)
        If F(X_i) < F(PBEST_i)
            PBEST = X_i
        End If
        If F(X_i) < F(GBEST)
            GBEST = X_i
        End If
    Update particle position and velocity according to Formulae (1) and (2)
    End For
    If F(GBEST) = TGBEST
        CGBEST++
    Else
        CGBEST = 0
        TGBEST = F(GBEST)
    End If
    If CGBEST ≥ RP
        CGBEST = 0
        Initial velocity of each particle, PBEST, GBEST, Initial position set the best
        index of position = 1 and other position set 0
    End If
Next gen
```

Figure 1. The pseudo code for XSLRKPSOR

From the pseudo code for XSLRKPSOR, feature selection is applied using LASSO in the training phase. Then, the processes of XGBoost, SVM, LR, RF, and KNN are applied. The result from each technique is its accuracy. The technique that achieves the highest accuracy has its parameter set to 1, while the other techniques are set to 0. This ensures that the parameter adjustment with PSO does not yield a result worse than the best performing technique. For example, if XGBoost achieves an accuracy of 0.8, SVM 0.7, LR 0.5, RF 0.6, and KNN 0.4, the initial parameters are set as $w1 = 1$, $w2 = 0$, $w3 = 0$, $w4 = 0$, and $w5 = 0$. Then, a modified PSO with re-initialization is used to search for the parameters that yield the best accuracy. In this example, the parameters may change to 0.4, 0.2, 0.2, 0.1, and 0.1. In the testing phase, feature selection is again applied using LASSO, followed by the processes of XGBoost, SVM, LR, RF, and KNN. The results from these processes are multiplied by the parameters obtained from the PSO search. Suppose the results from these processes are 1×0.4, 0×0.2, 0×0.2, 0×0.1, 0×0.1, which gives a value of 0.4. In this example, if the value is less than 0.5, the output is set to 0. So, the result from the XSLRKPSOR model is 0.

4. **Experiments and Results.** This research uses three datasets. All three datasets are divided into 75% of the entire dataset for training and 25% of the entire dataset for testing. The heart dataset used to compare the performance of algorithms for classifying

data, and the details of the dataset can be found and downloaded from [16]. The housing dataset is used to compare the performance of algorithms for regressing data, and the details of the dataset can be found and downloaded from [17]. The patient dataset is obtained from NHSO data. The data on the number of patients for each hospital from 2020 to 2022, covering 3 years, represents 75% of the dataset for training. The data for 2023, covering 1 year, represents 25% of the dataset used for testing. The dataset consists of a total of 23,045 rows and includes 83 attributes: hospital name, year of service, gender (2 attributes), age group (6 attributes), right group (49 attributes), patient type (2 attributes), disease group (21 attributes), and the number of patients in each hospital, which is the target for prediction. The patient dataset is used to compare the performance of algorithms for regressing data. The housing dataset and the heart dataset are employed as standard benchmarks for assessing algorithm performance, whereas the patient dataset is utilized to demonstrate practical applicability.

The performance metrics used in the experiments are described as follows: The Average Accuracy (AA) is calculated as the average of correctly identified predictions divided by the total number of data across all runs. AA serves as an indicator of the algorithm's classification performance. Higher AA values correspond to better classification performance. The Average RMSE (AR) is calculated as the average of RMSE across all runs. Smaller AR values correspond to better regression performance. The smaller the AR value, the closer the model's predictions are to the actual values. AA is used in the case of the heart dataset, while AR is used in the case of the housing dataset and the patient dataset.

The parameters used in all experiments are as follows. The population size is 200. The maximum number of iterations is limited to 2000. The number of experiments of each dataset is set as 20 runs. RP is set to 10. The parameters for the compared algorithm (XSPSO) and the proposed algorithm are configured based on the recommendations outlined in the original paper [13] and the standard values from the standard scikit-learn library. These specific values for RP are determined through experimentation. The experimentation process involves starting with RP set at 1 and then incrementing it gradually up to 100, ultimately selecting the values of RP that yield the best results. This research is conducted by a personal computer of Intel i7-14700F with 2.1 GHz CPU, 16 GB RAM and Python 3 as the programming language.

The experimental results in Tables 1 and 2 clearly indicate that XSLRKPSOR surpasses XSPSO, XSPSOR, and XSLRKPSO in terms of classification and regression performance, as evidenced by its superior AA and AR across all datasets. XSPSOR achieves better classification and regression performance in both the test and training phases compared to XSPSO. Similarly, XSLRKPSOR achieves better classification and regression performance

Table 1. Comparative AA of XGBoost, SVM, KNN, RF, LR, XSPSO, XSPSOR, XSLRKPSO, and XSLRKPSOR on the heart dataset

	XGBoost	**SVM**	**KNN**	**RF**	**LR**	**XSPSO**	**XSPSOR**	**XSLRKPSO**	**XSLRKPSOR**
Train	96.62	70.87	91.44	96.62	83.81	96.96	96.99	97.05	100
Test	84.6	70.28	73.58	84.93	82.58	91.8	92.86	95.1	99.55

Table 2. Comparative AR of XGBoost, SVM, KNN, RF, LR, XSPSO, XSPSOR, XSLRKPSO, and XSLRKPSOR on the housing dataset and the patient dataset

	Dataset	**XGBoost**	**SVM**	**KNN**	**RF**	**LR**	**XSPSO**	**XSPSOR**	**XSLRKPSO**	**XSLRKPSOR**
Train	Housing	132.37	256.5	417.62	117.39	483.6	128.74	120.74	107.63	106.13
Test	Housing	365.89	482.47	477.34	247.15	507.37	345.9	271.48	225.98	215.2
Train	Patient	168.26	512.06	367.10	132.13	425.78	157.32	144.32	101.09	94.34
Test	Patient	238.73	522.22	403.04	182.52	547.42	236.31	191.05	130.03	121.67

in both the test and training phases compared to XSLRKPSO. This proves that the re-initialization can enhance the performance of PSO parameter adjustment, leading to the creation of better models with higher accuracy and lower RMSE. In the case of the heart dataset, the results show that XGBoost can create a model that performs the best. However, for the housing dataset and the patient dataset, RF can create a model that performs the best. In practice, we do not know which algorithm can solve a problem most effectively. Having a variety of algorithms to choose from provides a better chance of creating a better model compared to having fewer algorithms to choose. Additionally, experiments show that XSLRKPSO achieves better classification and regression performance in both the test and training phases compared to XSPSO. Similarly, XSLRKPSOR achieves better classification and regression performance in both the test and training phases compared to XSPSOR. This demonstrates that having more algorithms to choose from leads to the creation of better models with higher accuracy and lower RMSE.

5. **Conclusion.** Machine learning includes various algorithms such as XGBoost, SVM, KNN, RF, and LR for model creation. Each algorithm has different effectiveness for creating models for prediction. In practice, we do not know which algorithm performs the best. This research proposes a hybrid model to create a model that is more effective than using a single model. Additionally, PSO has a problem with trapping local optima. To address this, this research suggests using a re-initialization technique. The result from the re-initialization technique affects the creation of better models. This research proposes using both techniques together to result in the most effective algorithm for model creation. The experiments show that the solutions produced by the proposed algorithm outperform those obtained through comparison with other algorithms in terms of classification and regression performance for both standard datasets and practical usage datasets. In the future, researchers plan to incorporate additional algorithms into the hybrid model, such as Gradient Boosting, Convolutional Neural Networks (CNN), Recurrent Neural Networks (RNN), and Artificial Neural Networks (ANN), which may further improve model performance.

Acknowledgment. This research was financially supported by the Research Fund of the College of Interdisciplinary Studies, Thammasat University, under the category of research collaborative research grant for the fiscal year 2024, and was further supported with data from the National Health Security Office (NHSO) Thailand. The research team would like to express their sincere gratitude for this support.

REFERENCES

[1] K. G. Dinesh, K. Arumugaraj, K. D. Santhosh and V. Mareeswari, Prediction of cardiovascular disease using machine learning algorithms, *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, Coimbatore, India, pp.1-7, 2018.

[2] H. Jindal, S. Agrawal, R. Khera, R. Jain and P. Nagrath, Heart disease prediction using machine learning algorithms, *IOP Conference Series: Materials Science and Engineering, 1st International Conference on Computational Research and Data Analytics (ICCRDA 2020)*, vol.1022, 2020.

[3] V. Guda, K. Shalini and C. Shivani, Heart disease prediction using hybrid technique, *Journal of Interdisciplinary Cycle Research*, vol.12, no.6, 2020.

[4] A. Garg, B. Sharma and R. Khan, Heart disease prediction using machine learning techniques, *IOP Conference Series: Materials Science and Engineering*, 2021.

[5] L. Chandrika and K. Madhavi, A hybrid framework for heart disease prediction using machine learning algorithms, *E3S Web of Conferences*, DOI: 10.1051/e3sconf/202130901043, 2021.

[6] R. E. Saragih, D. Gloria and A. J. Santoso, Classification of ambarella fruit ripeness based on color feature extraction, *ICIC Express Letters*, vol.15, no.9, pp.1013-1020, 2021.

[7] W. Yang, L. Wei, Z. Zhang and J. Liu, Road surface condition classification method based on multi-feature fusion and SVM classifier, *ICIC Express Letters, Part B: Applications*, vol.7, no.2, pp.461-468, 2016.

[8] Y. Lin, H. Guo and J. Hu, An SVM-based approach for stock market trend prediction, *The 2013 International Joint Conference on Neural Networks (IJCNN)*, Dallas, TX, USA, pp.1-7, 2013.

[9] V. Khatibi, E. Khatibi and A. Rasouli, A new support vector machine genetic algorithm (SVMGA) based method for stock market forecasting, *International Journal of the Physical Sciences*, vol.6, no.25, pp.6091-6097, 2011.

[10] R. E. Caraka, R. C. Chen, T. Toharudin, M. Tahmid, B. Pardamean and R. M. Putra, Evaluation performance of SVR genetic algorithm and hybrid PSO in rainfall forecasting, *ICIC Express Letters, Part B: Applications*, vol.11, no.7, pp.631-639, 2020.

[11] C. Sun, H. Yuan, X. Mao, X. Lu and R. Hamzaoui, Enhancing octree-based context models for point cloud geometry compression with attention-based child node number prediction, *IEEE Signal Processing Letters*, vol.31, pp.1835-1839, 2024.

[12] E. I. Elsedimy, S. M. M. AboHashish and F. Algarni, New cardiovascular disease prediction approach using support vector machine and quantum-behaved particle swarm optimization, *Multimedia Tools and Applications*, vol.83, no.8, pp.23901-23928, 2024.

[13] A. B. Majumder, S. Gupta, S. Majumder and D. Singh, A heart disease prediction model using merged XGBoost-SVM classifier and particle swarm optimization, *2024 5th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI)*, Lalitpur, Nepal, pp.241-248, 2024.

[14] E. Anderson et al., An experimental study of data migration algorithms, in *Algorithm Engineering. WAE 2001. Lecture Notes in Computer Science*, G. S. Brodal, D. Frigioni and A. Marchetti-Spaccamela (eds.), Berlin, Heidelberg, Springer, 2001.

[15] T. White, *Hadoop: The Definitive Guide*, O'Reilly Media, 2012.

[16] *https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset*, Accessed on August 27, 2024.

[17] *https://www.kaggle.com/datasets/altavish/boston-housing-dataset*, Accessed on August 27, 2024.

[18] J. Kennedy and R. Eberhart, Particle swarm optimization, *Proc. of International Conference on Neural Networks (ICNN'95)*, vol.4, pp.1942-1948, 1995.

[19] C. Ratanavilisagul and B. Kruatrachue, A modified particle swarm optimization with mutation and reposition, *International Journal of Innovative Computing, Information and Control*, vol.10, no.6, pp.2127-2142, 2014.

[20] C. Ratanavilisagul, Modified ant colony optimization with route elimination and pheromone reset for multiple pickup and multiple delivery vehicle routing problem with time window, *Journal of Advanced Computational Intelligent and Intelligent Informatics*, vol.26, no.6, pp.959-964, 2022.

[21] C. Ratanavilisagul, Modified ant colony optimization for vehicle routing problem with time windows using limited search space and novel updating pheromone and re-initialization pheromone techniques, *ICIC Express Letters, Part B: Applications*, vol.13, no.6, pp.571-580, 2022.

[22] S. Wang, Y. Chen, Z. Cui, L. Lin and Y. Zong, Diabetes risk analysis based on machine learning LASSO regression model, *Journal of Theory and Practice of Engineering Science*, vol.4, no.1, pp.58-64, 2024.

[23] X. Zhao et al., Effective combination of 3D-DenseNet's artificial intelligence technology and gallbladder cancer diagnosis model, *Frontiers in Computing and Intelligent Systems*, vol.6, no.3, pp.81-84, 2024.

[24] T. Q. Chen and C. Guestrin, XGBoost: A scalable tree boosting system, *Proc. of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, San Francisco, pp.785-794, 2016.

[25] A. Gudur, V. Patil, L. Gopal and S. Thapliya, Multi-objective optimization for breast cancer risk prediction models with particle swarm optimization, *International Journal of Intelligent Systems and Applications in Engineering*, vol.12, no.3s, pp.531-541, 2023.

[26] S. G. Kulkarni and M. V. Babu, Introspection of various K-nearest neighbor techniques, *UACEE Int. J. Adv. Comput. Sci. Appl.*, vol.3, no.2, pp.103-106, 2013.