Uniport Journal of Engineering and Scientific Research (UJESR)

Vol. 8, Issue 2, 2024, Page 241-248
ISSN: 2616-1192

(www.ujesr.org)

Harnessing Artificial Intelligence for Enhancing Short-Term Load Forecasting and Microgrid Operation

Gwaivangmin, B.I[1]*, Bakare, G.A[2], Haruna, Y.S[3] and Amoo, A.L[4]
[1,2,3,4]Department of Electrical and Electronics Engineering Abubakar Tafawa Balewa University, Bauchi
*Corresponding author's email: bgwaivangmin@gmail.com

Abstract
Microgrids are increasingly recognized as vital for delivering reliable, decentralized energy supply, especially in off-grid locations. Efficient microgrid management and resource optimization hinge on precise short-term load forecasting. This research evaluated the effectiveness of three different machine learning models, namely SVM-PSO, SVM-NN, and SVM-SA to predict short-term load demand in a university microgrid connected to an 11 kV dedicated feeder. Utilizing five years of load data from Jos Electricity Distribution Company and meteorological data from NASA, the models were assessed for accuracy using metrics such as Root Mean Square Error (RMSE), Mean Squared Error (MSE), and Mean Absolute Percentage Error (MAPE). The results showed that SVM-NN achieved the lowest RMSE (109.9204) and MSE (12,082.4913), indicating superior accuracy, while SVM-PSO and SVM-SA exhibited comparable performance with slightly better MAPE values (0.1535) compared to SVM-NN (0.1604). These offer consistent and competitive performance, making them viable alternatives depending on specific forecasting requirements and performance priorities.

Keywords: *Energy, Forecasting, Load demand, Management, Meteorological, Microgrid*

Received: 7th August, 2024 *Accepted: 18th October, 2024*

1. Introduction

Uddin *et al.* (2023) notes the rising popularity of microgrids for supplying localized and dependable electricity, especially in remote areas. These systems can operate in conjunction with the main grid or independently, providing increased energy security, reduced power outages, and better integration of renewable energy sources. However, efficient microgrid management requires accurate short-term electricity load demand forecasting to optimize the allocation of resource, operational costs reduction, and maintain the stability of the grid.

A range of machine learning techniques, including expert systems, fuzzy systems, Particle Swarm Optimization (PSO), neural networks, and Support Vector Machines (SVM), have been devised for predicting electricity load demand (Zhao, 2023; Aquila *et al.*, 2023).

Fida *et al*. (2024) proposed utilizing an improved sparrow search algorithm (ISSA) to optimize the hyperparameters of a Support Vector Machine (SVM) for enhanced mid- to long-term load forecasting. The ISSA-SVM model demonstrated superior performance compared to traditional SVM, neural networks, and linear regression models.

Moradzadeh *et al.* (2020) and Li *et al.* (2022) emphasized on the increasing prominence of microgrids as a reliable electricity supply solution, particularly in remote areas, highlighting the importance of enhancing energy security, minimizing power outages, and maximizing renewable energy utilization. Musa *et al.* (2021) developed a method for forecasting multi-state load demand and optimizing the sizing of a standalone hybrid energy system combining PV, wind, and battery technologies. The study employs a Support Vector Regression (SVR) approach to forecast the variability of load demand across four different states in Nigeria. To address the limitations of SVR, hybrid SVR algorithms, SVR-HHO and SVR-PSO, were developed to improve prediction accuracy.

This study aims to predict short-term load demand for a microgrid system with an 11 kV feeder at the University of Jos, Nigeria, using artificial intelligence techniques: Hybrid Support Vector Machine optimized by Particle swarm Optimization (SVM-PSO), Hybrid Support Vector Machine optimized by Neural network (SVM-NN), and Hybrid Support Vector Machine optimized by Simulated Annealing (SVM-SA).The study is focused on evaluating and comparing the performance of SVM-PSO, SVM-NN, and SVM-SA for short-term load demand forecast within a microgrid context. While existing literature, such as studies by Khantach *et al.* (2019) and Park *et al.* (2020), has explored various load forecasting models and techniques, this study provides a comprehensive analysis of these techniques' predictive capabilities using real-world data from a microgrid system in Plateau State, Nigeria. Furthermore, this study goes beyond just prediction accuracy to providing valuable insights for energy management and decision-making within a microgrid systems, addressing a crucial need in the field. The study employs three statistical methods, namely SVM-PSO, SVM-NN, and SVM-SA, to forecast short-term load demand, which is essential for microgrid management and allocation of resource. As microgrids gain popularity for localized electricity supply, accurate load forecasts become vital for efficient operation. Leveraging machine learning techniques and historical data, the study evaluates the performance of these methods, providing insights for effective management of energy and making important decision in microgrid systems.

2. Materials and methods

2.1 SVM mathematical model

Hassanien and Emary (2020), Lazzeri (2020) Ji *et al.* (2023) and Pettit *et al.* (2019) in their submission made the following presentation that SVM-PSO is a hybrid algorithm that integrates the strengths of SVM and PSO. This hybrid algorithm is often used to enhance the parameters of SVM for enhanced performance in classification or regression tasks.

2.1.1 SVM objective function

For a given training dataset $\{(x_i y_i)\}_{i=1}^{N}$, where $x_i \in R^n$ represents the feature vector and $y_i \in \{-1, 1\}$ denotes the class label, the objective of Support Vector Machine (SVM) is to identify a hyperplane that maximizes the margin between the two classes. The decision function is expressed as:

$$f(x)=w \cdot x+b \quad (1)$$

The minimized objective function is represented as:

$$min_{w,b}\left(\frac{1}{2}\|w\|^2 + C\sum_{i=1}^{N}\xi_i\right) \quad (2)$$

with the following constraints:

$$y_{i(w.x_i+b)} \geq 1-\xi_i, \xi_i \geq 0, \qquad i=1,\ldots\ldots,N$$

In this context, w refers to the weight vector, b refers to the bias term, ξ_i are slack variables, and C is the regularization parameter that balances the trade-off between maximizing the margin and minimizing classification error.

2.1.2 Mathematical model of PSO

PSO is a population-based optimization algorithm inspired by the collective behaviour observed in flocks of birds and schools of fish. In PSO, a swarm of particles explores the solution space to find the optimal solution. (Khan *et al.*,2023; Wang *et al.*2017). Individual particles are characterized by position and velocity vectors (pi, vi) in n-space. Their dynamical behaviour is determined by interactions with their local environment as shown in Equations (3) and (4).

The velocity update rule is: $v_{ij}(t+1) = \omega v_{ij}(t)+c_1r_1[p_{ij}^{best}-p_{ij}(t)] + c_2r_2[g_{ij}^{best}-p_{ij}(t)]$ (3)

The position update rule is:

$$p_{ij}(t+1) = p_{ij}(t)+v_{ij}(t+1) \quad (4)$$

where $v_{ij}(t)$ refers to the velocity of particle i in dimension j at time step t, $p_{ij}(t)$ refers to the position of particle iii in dimension j at time step t, ω refers to the inertia weight, c_1 and c_2 refers to cognitive and social constants, r_1 and r_2 refers to random numbers uniformly distributed in [0,1], p_{ij}^{best} refers to the best position found by particle i so far, and g_{ij}^{best} refers to the best position found by any particle in the swarm.

2.1.3 Neural network model

According to Goodfellow *et al.* 2016; Aggarwal (2018) and Zhang and Xu (2017). The output of the SVM $f_{SVM}(x)$ is fed as input into the Neural Network for further refinement. Let $f_{NN}(x)$ denote the function learned by the Neural Network, which is typically a multilayer perceptron (MLP). The NN adjusts the initial forecast by learning from the residuals of the SVM output. The neural network is made up of layers of neurons with the general form shown in Equations (5) and (6).

$$z^l = \sigma(W^l a^{l-1} + b^l) \tag{5}$$

where z^l refers to the activation of layer l, W^l and b^l refers to the weights and biases of layer l, σ refers to the activation function (e.g., ReLU, sigmoid), a^{l-1} refer to the activation from the previous layer, and with $a^0 = f_{SVM}(x)$. The final output of the NN is:

$$f_{NN}(x) = z^{(L)} \tag{6}$$

where L refers to the number of layers in the network.

Objective function: The neural network's training aims to reduce the average squared difference between predicted and actual values as shown in Equation (7)

$$min_{w,b} \frac{1}{N} \sum_{i=1}^{N} (r_i - f_{NN}(f_{SVM}(x_i)))^2 \tag{7}$$

where w and b refer to the weights and biases of the neural network. By combining the strengths of SVM in handling non-linear relationships and the neural network's capability to model complex patterns, the hybrid SVM-NN model is to improve the accuracy and robustness of electricity load demand prediction.

2.1.4 Simulated annealing (SA) model

Simulated Annealing is employed to optimize SVM hyperparameters, including the regularization parameter C, the insensitive loss parameter ε, and the kernel parameter γ for the Radial Basis Function (RBF) kernel. (Ibrahim *et al.*,2020; Khajehzadeh *et al.*,2023).

Objective Function**:** Minimize the Mean Squared Error (MSE) on the validation set shown in Equation (8).

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y})^2 \tag{8}$$

where$\hat{y}$ is the predicted load.

Temperature Schedule: The temperature T is gradually decreased according to a cooling schedule. A common schedule is shown in Equation (9).

$$T_{k+1} = \alpha T_k \text{ With } \alpha \in (0,1) \tag{9}$$

Acceptance Probability: The probability of accepting a worse solution is represented as in Equation (10).

$$\mathrm{P}(\mathrm{E}, \mathrm{E}', \mathrm{T}) = e^{\frac{E - \mathrm{E}\prime}{T}} \tag{10}$$

where E and E′are the objective function values (MSE) of the current and new solutions, respectively.

2.2 Data normalization

According to Hastie et al. (2015); Kuhn and Johnson (2013); Alpaydin (2020); Iglewicz and Hoaglin (2017) and Bishop (2021). Normalization adjusts the scale of the data to a common range, typically [0, 1] or [-1, 1], without distorting differences in the ranges of values.

2.2.1 Mathematical equations for normalization

There are several methods for normalizing data. Min-Max normalization and Z-score normalization are the most common. For this study Min-Max normalization and Decimal scaling were explored.

2.2.2 Min-Max normalization

Min-Max normalization rescales data to a fixed range, typically between 0 and 1. Min-Max normalization is represented as in Equation (11)

$$x' = \frac{\{x - min(x)\}}{\{max(x)^{-min(x)}\}} \tag{11}$$

where, *x* is the original value that needs to be normalized. Whereas *min(x)* is the smallest value present in the range of *x*, *max(x)* is the largest value present in the range of *x*, and the normalized value is represented by *x'*. This formula transforms the original data to a new scale ranging from 0 to 1, where the minimum value of the original data maps to 0 and the maximum value maps to 1 with 0 and 1 as mean and standard deviation respectively.

2.2.3 Decimal scaling

Decimal scaling normalizes data by shifting the decimal point a fixed number of places as shown in Equation (12).

$$x' = \frac{x}{10^j} \qquad 12)$$

Here *j* refers to the smallest integer such that max(|x'|) <1.

2.3 Model evaluation metrics

According to (Kuhn & Johnson (2021), Hastie *et al*. (2020), James *et al*. (2021), Friedman *et al*. (2022) and Bishop (2006). Model evaluators are metrics or criteria used to measure the precision and effectiveness of predicting models. They give insights into how well the model generalizes to unseen data and help identify areas where the model can be improved. For this study the model evaluators in Equations (13), (14) and (15) were used.

$$MSE = \frac{1}{n}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \quad (13)$$

where n denotes total number of instances, y_i denotes projected value, $\hat{y}_i$ represents the factual value. MSE disproportionately penalizes larger errors due to the squaring operation.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (14)$$

where n refers to the number of samples, y_i refers to the actual observed values. $\hat{y}_i$ refers to the predicted values. RMSE offers an interpretable error metric aligned with the original data units.

$$MAE = \frac{1}{n}\sum_{i=1}^{N}\left\|\frac{y_i - \hat{y}_i}{y_i}\right\| x\ 100 \quad (15)$$

where n denotes total number of examples, y_i denotes anticipated value, $\hat{y}_i$ signifies the factual value. MAPE expresses error as a percentage of actual values, enabling comparison of forecast accuracy across different scales. Lower RMSE, MSE, and MAPE values generally indicate improved model performance, as they signify smaller prediction errors. The components employed in predicting the load demand were:

i. Load demand data: Load demand data is the 5-year historical primary input data sourced from the Jos Electricity Distribution Company (JEDC).
ii. Meteorological data: Humidity, temperature, solar insolation, wind speed, and precipitation data were sourced from NASA (2023).
iii. SVM-PSO, SVM-NN, and SVM-SA software: SVM, NN, and SA software Modules in Python (Spyder-Anaconda)

2.4 Method

2.4.1 Outline of energy demand forecasting using SVM-Based techniques

(i)Data Collection and Preparation(a)Primary Load Demand Data: This was Collected from the University of Jos.(b)Meteorological Data: Included solar irradiance, temperature, and humidity were collected from the NASA website. (c)Data Cleaning: This was done consistently and filled missing values. Anomalies were removed to maintain data integrity. (ii)Data Preprocessing: (a)Normalization: The data were scaled using the Min-Max method to normalize values. (b)Training and Testing Split: The data was split into training and testing sets, using a ratio of 70:30. (iii)Model Implementation (a) Python coding on the Spyder platform for forecasting implementation was done using the following models(b)SVM-NN (Support Vector Machine - Neural Network), (b) SVM-PSO (Support Vector Machine - Particle Swarm Optimization),(c) SVM-SA (Support Vector Machine - Simulated Annealing) SA was employed to minimize the error function of the SVM model, ensuring optimal parameter values.
The methodology for load demand forecasting using SVM-PSO, SVM-NN, and SVM-SA encompassed four primary stages: data collection and preprocessing, model development, model evaluation, and model selection.

2.5 Short-term load forecasting implementation based on SVM-PSO, SVM-NN and SVM-SA

The execution of SVM-PSO, SVM-NN and SVM-SA for Short-term Load Forecasting was carried out by writing a code in Python and was executed on the Spyder-Anaconda platform guided by data collection, processing and cleaning.

2.6 Collection of data, pre-processing and cleaning

Primary load demand data was collected from the 11kV dedicated feeder at JEDC substation at the University of Jos's Naraguta campus using a digital logger. This data spanned five years, from January 2018 to January 2023. Humidity, temperature, wind speed, solar insolation, and precipitation data were collected for the specified period. The gathered data underwent pre-processing via the subsequent procedures. Data cleaning involved eliminating duplicates, addressing missing values, rectifying formatting errors, ensuring consistency, and verifying accuracy within both the load demand and weather datasets.

3. Results and discussion

3.1 Outcome of SVM-PSO, SVM-NN, and SVM-SA models for short-term load forecasting

This section presents the results obtained from the implementation of SVM-PSO, SVM-NN, and SVM-SA models for short-term load forecasting. The performance of each model was evaluated using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and Mean Absolute Percentage Error (MAPE). SVM-NN emerged as the most effective model for short- term load forecasting, outperforming SVM-PSO and SVM-SA in terms of accuracy and precision. This is evident in the lower RMSE and MSE values achieved by SVM-NN.

3.2 Model performance

All three models (SVM-PSO, SVM-NN, and SVM-SA) demonstrated comparable performance in terms of overall trend prediction, capturing the general shape and peak periods of load demand as shown in Fig. 1, 2 and 3. Fig. 1 shows a comparative analysis of actual versus predicted load demand over the next 24 hours reveals that both curves follow a similar trend, reflecting the overall shape and peak periods of the load demand. The actual load peaks around 08:00 and then decreases, with the predicted load also peaking around this time but slightly underestimating the peak. Both curves show a notable drop in demand from 01:00 to 04:00, with predicted values closely matching actual values in this low-demand period. Between 10:00 and 18:00, both loads fluctuate, with the predicted values reasonably capturing these fluctuations but slightly underestimating around 12:00. In the evening from 18:00 to 22:00, the predicted load closely tracks the actual load with minor underestimations, and from 22:00 to 24:00, both curves decrease sharply, with the forecast values being slightly more than the actual values in the final hours.

Fig. 2 shows a comparative analysis of actual versus predicted demand for the next 24 hours, showing that both curves follow a similar trend, capturing the overall shape and peak periods of load demand. The actual load peaks around 08:00 and then decreases, with the predicted load also peaking around this time but slightly underestimating the actual peak. Both curves show a significant drop in demand from 01:00 to 04:00, with predicted values close to the actual values during this low-demand period. Between 10:00 and 18:00, both actual and predicted loads fluctuate, with the predicted values capturing these fluctuations reasonably well, though slightly lower than actual around 12:00. In the evening, from 18:00 to 22:00, both curves decrease, with the predicted load closely tracking the actual load but slightly underestimating at some points. From 22:00 to 24:00, the load decreases sharply, with predicted values following this trend but slightly higher than actual values in the final hours.

Fig. 3 shows a comparative analysis of actual versus predicted demand over the next 24 hours, showing that both curves follow a similar trend, capturing the general shape and peak periods of load demand. The actual load peaks around 08:00 and then decreases, while the predicted load also peaks at this time but underestimates the actual peak. Both curves show a significant drop in demand from 01:00 to 04:00, with predicted values closely matching actual values during this low-demand period. Between 10:00 and 18:00, both actual and predicted loads exhibit fluctuations, with the predicted values capturing these reasonably well but slightly underestimating some midday peaks. From 18:00 to 22:00, both curves decrease, with the predicted load closely tracking the actual load, though slightly overestimated at some points. In the final hours from 22:00 to 24:00, the load decreases sharply, with predicted values following this trend but remaining slightly higher than the actual values.

The SVM-NN model outperformed the SVM-PSO and SVM-SA models in terms of both MSE and RMSE, indicating superior accuracy and model fit. While SVM-PSO and SVM-SA demonstrated comparable performance, SVM-NN's lower error metrics suggest its potential for more precise load demand forecasting. The findings align with the studies by Selakov *et al*. (2014), Ren *et al*. (2016) and Chauhan *et al*. (2022) using SVM, PSO and others. Hybrid models combining SVM and NN have shown promising results in various forecasting tasks, including load demand prediction.

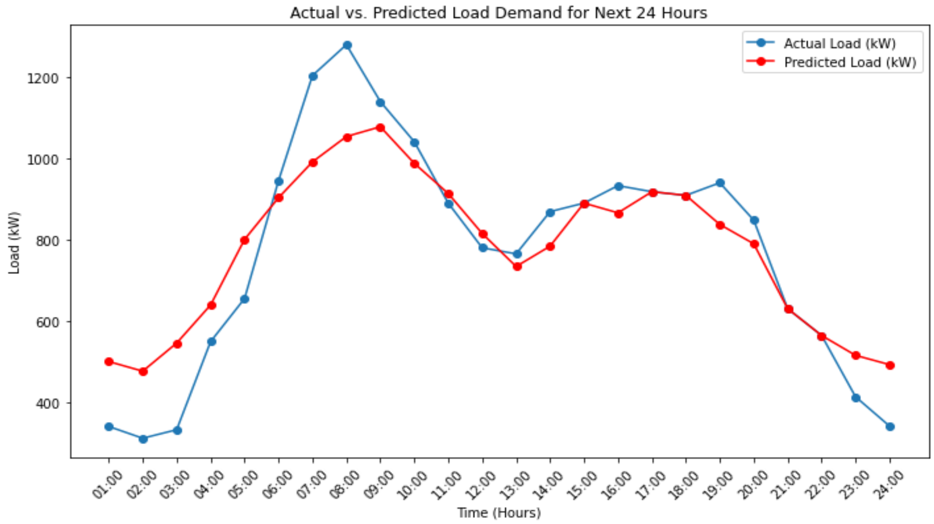

Fig. 1: SVM-PSO line diagram of 24-hour load demand forecast

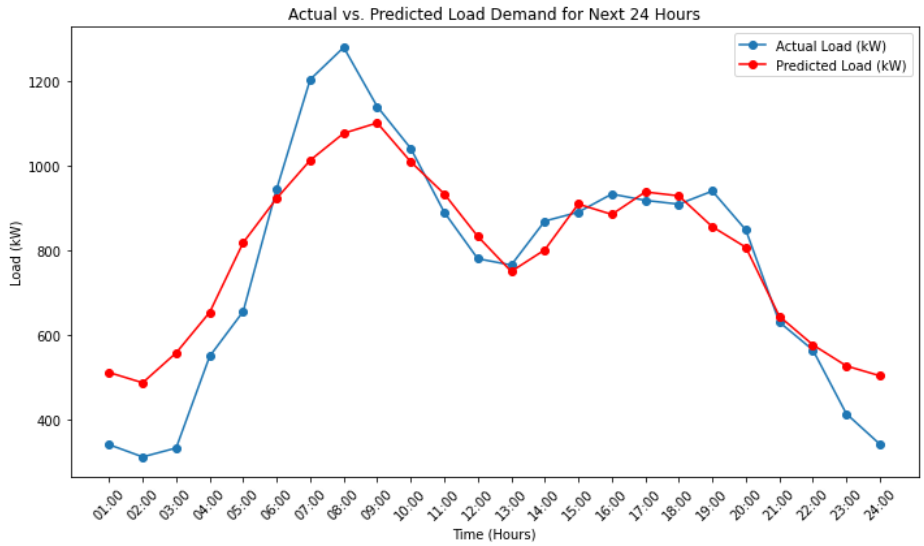

Fig. 2: SVM-NN line diagram of 24-hour load demand forecast

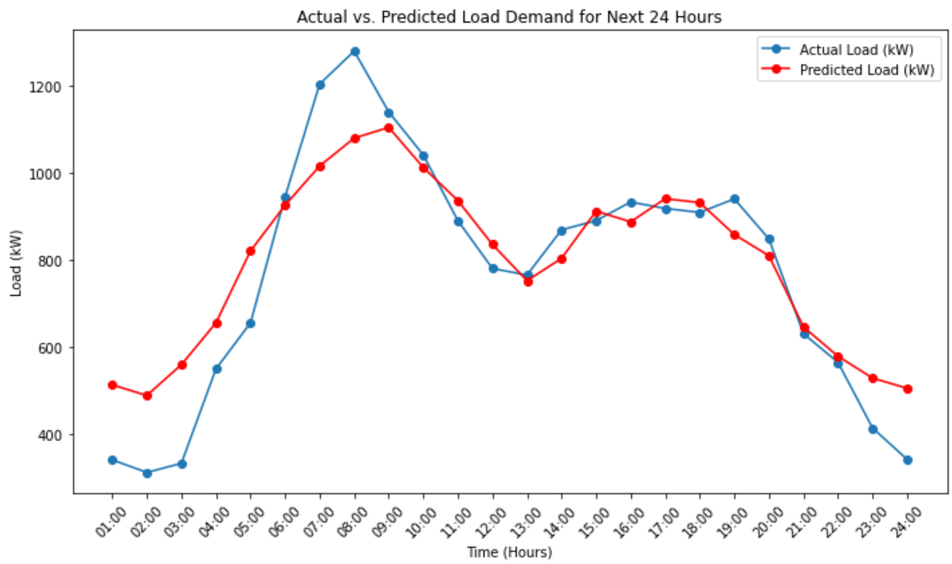

Fig. 3: SVM-SA Line Diagram of 24-Hour Load Demand forecast

3.3 Comparison of SVM-PSO, SVM-NN, and SVM-SA performance

Table 1 compares the performance of SVM-PSO, SVM-NN, and SVM-SA models using RMSE, MSE, and MAPE. A comparative analysis of SVM-PSO, SVM-NN, and SVM-SA models for predicting 24-hour load demand reveals significant performance disparities. The SVM-NN model emerges as the superior choice, exhibiting the lowest RMSE (109.9204) and MSE (12,082.4913) among the three. These metrics indicate superior accuracy and model fit in predicting load demand. While SVM-PSO and SVM-SA demonstrate comparable performance with RMSE values of 110.7015 and 110.7016, respectively, and identical MAPE of 0.1535, their overall performance lags behind SVM-NN. Consequently, SVM-NN is recommended for applications requiring precise 24-hour load demand forecasting.

Table 1: Comparison of SVM-PSO, SVM-NN, and SVM-SA performance

Method	RMSE	MSE	MAPE
SVM-PSO	110.7015	12254.8311	0.1535
SVM-NN	109.9204	12082.4913	0.1604
SVM-SA	110.7016	12254.8525	0.1535

4. Conclusion

In conclusion, the comparison of SVM-PSO, SVM-NN, and SVM-SA models reveals that SVM-NN achieves the highest accuracy in absolute terms, with the lowest RMSE of 109.9204 and MSE value of 12082.4913, signifying effective error minimization. However, SVM-PSO and SVM-SA exhibit comparable performance, with slightly better relative error percentages as indicated by their MAPE values. While SVM-NN excels in accuracy, SVM-PSO and SVM-SA offer consistent and competitive performance, making them viable alternatives depending on specific forecasting requirements and performance priorities.

References

Aggarwal, C.C. (2018) Neural networks and deep learning: A textbook. Springer. ISBN: 978-3319944623.

Alpaydin, E. (2020) Introduction to machine learning (4th ed.). Massachusetts Institute of Technology Press. ISBN: 978- 0262043793.

Aquila, G., Morais, L.B.S., de Faria, V.A.D., Lima, J.W.M., Lima, L.M.M. and de Queiroz, A.R. (2023) An overview of short-term load forecasting for electricity systems operational planning: Machine learning methods and the Brazilian experience Energies, 16(21), 7444.

Bishop, C.M. (2006) Pattern recognition and machine learning. Springer Science Business Media, LLC. ISBN-13: 978-0387-31073-2.

Chauhan, M., Gupta, S. and Sandhu, M. (2022) Short-term electric load forecasting using support vector machines. ECS Transactions, 107(1), 9731.

Fida, K., Abbasi, U., Adnan, M., Iqbal, S. and Eldeen, S.G.M. (2024) A comprehensive survey on load forecasting hybrid models: Navigating the Futuristic demand response patterns through experts and intelligent systems. Results in Engineering, 23, 102773.

Goodfellow, I., Bengio, Y. and Courville, A. (2016) Deep learning. Massachusetts Institute of Technology Press. ISBN: 978-0262035613.

Hassanien, A.E. and Emary, E. (2020) Swarm intelligence: Principles, advances, and applications (1st ed.). CRC Press. ISBN 9780367737542.

Hastie, T., Tibshirani, R. and Wainwright, M.J. (2015) Statistical learning with sparsity: The lasso and generalizations. Chapman and Hall. ISBN: 9780429171581.

Ibrahim, Y., Okafor, E. and Yahaya, B. (2020) Optimization of RBF-SVM hyperparameters using genetic algorithm for face recognition. Nigerian Journal of Technology, 39(4): 1190–1197.

Iglewicz, B. and Hoaglin, D.C. (1993) How to detect and handle outliers (ASQC basic references in quality control) (1st ed.). American Society for Quality Press. ISBN-13: 978-0873892476.

James, G., Witten, D., Hastie, T. and Tibshirani, R. (2013) An introduction to statistical learning with applications in R. Springer Science Business Media New York. ISBN: 978-1-4614-7138-7 (eBook).

Ji, X., Huang, H., Chen, D., Yin, K., Zuo, Y., Chen, Z. and Bai, R. (2023) A hybrid residential short-term load forecasting method using attention mechanism and deep learning. Buildings, 13(1), 72.

Khajehzadeh, M. and Keawsawasvong, S. (2023) Predicting slope safety using an optimized machine learning model. Heliyon, 9(12), e23012.

Khan, A., Shafi, I., Khawaja, S.G., de la Torre Díez, I., Flores, M.A.L., Galván, J.C. and Ashraf, I. (2023) Adaptive filtering: Issues, challenges, and best-fit solutions using particle swarm optimization variants. Sensors (Basel), 23(18), 7710.

Khantach, A., Hamlich, M. and Belbounaguai, N. (2019) Short-term load forecasting using machine learning and periodicity decomposition. Advances in Materials Science Energy, 7(3): 382–394.

Kuhn, M. and Johnson, K. (2013) Applied predictive modeling. Springer. ISBN: 978-1-4614-6849-3 (eBook).

Lazzeri, F. (2020) Machine learning for time series forecasting with Python. Wiley. ISBN: 978-1-119-68236-3.

Li, J., Lei, Y. and Yang, S. (2022) Mid-long term load forecasting model based on support vector machine optimized by improved sparrow search algorithm. Energy Reports, 8: 491–497.

Moradzadeh, A., Zakeri, S., Shoaran, M., Mohammadi-Ivatloo, B. and Mohammadi, F. (2020) Short-term load forecasting of microgrid via hybrid support vector regression and long short-term memory algorithms. Sustainability, 12(17), 7076.

Musa, B., Yimen, N., Abba, S. I., Adun, H. H. and Dagbasi, M. (2021). Multi-state load demand forecasting using hybridized support vector regression integrated with optimal design of off-grid energy systems—A metaheuristic approach. Processes, 9(7), 1166.

National Aeronautics and Space Administration (NASA) (2023) Weather Data. Retrieved 30 December 2023 from https://www.nasa.gov/weather-data.

Park, R.B., Song, K. B., and Kwon, B. S. (2020) Short-term load forecasting algorithm using a similar day selection method based on reinforcement learning. Energies, 13, 2640.

Pettit, R.W., Fullem, R., Cheng, C. and Amos, C.I. (2021) Artificial intelligence, machine learning, and deep learning for clinical outcome prediction. Emerging Topics in Life Sciences, 5(6): 729–745.

Ren, G., Wen, S., Yan, Z., Hu, R., Zeng, Z. and Cao, Y. (2016) Power load forecasting based on support vector machine and particle swarm optimization. In 2016 12th World Congress on Intelligent Control and Automation (WCICA) (pp. 1853–1858). IEEE.

Selakov, A., Cvijetinović, D., Milović, L., Mellon, S. and Bekut, D. (2014) Hybrid PSO–SVM method for short-term load forecasting during periods with significant temperature variations in the city of Burbank. Applied Soft Computing, 16, 80–88.

Uddin, M., Mo, H., Dong, D., Elsawah, S., Zhu, J. and Guerrero, J.M. (2023) Microgrids: A review, outstanding issues and future trends. Energy Strategy Reviews, 49, 101127.

Wang, D., Tan, D. and Liu, L. (2017) Particle swarm optimization algorithm: An overview. Soft Computing.

Zhao, Z. (2023) Comprehensive review of machine learning-based methods for electricity load forecasting. Environment, Energy, and Earth Sciences Web of Conferences., 394, 01002.