Developing Machine Learning Models to Predict Outdoor Thermal Comfort of Kinetic Shading Devices

An approach for global optimization

Yiğit Dağlier[1], Berk Ekici[2], Koray Korkmaz[3]
[1,2,3] Izmir Institute of Technology (IZTECH)
[1,2,3] {yigitdaglier|berkekici|koraykorkmaz}@iyte.edu.tr

Utilizing artificial intelligence (AI) methods in the design process supports the achievement of sustainable alternatives during the conceptual design. In various AI methods, optimization and machine learning (ML) algorithms are the most common methods to develop predictive models and discover favorable design alternatives with significantly reduced computational time. Recent works focused on limited datasets, as well as the evaluation of the developed prediction models based on collected data. During the optimization process of complex design problems, the number of design parameters becomes enormous; thus, search areas contain many design alternatives that might lead the search outside of the collected data. Therefore, evaluating the accuracy of prediction models only based on the collected samples may result in scenarios where the predicted outcome during the optimization process aligns with an unrealistic solution. This study investigates how accurately prediction models developed using different ML algorithms can perform in optimization processes. The proposed framework is used to cope with outdoor thermal performance, considering kinetic shading devices with rigid origami techniques. A parametric shading device model with kinematic principles and 10 design parameters is created in Grasshopper 3d. LadyBug is used to analyze the performance of the universal thermal climate index (UTCI). To minimize the UTCI, the radial basis function optimization (RBFOpt) algorithm in the Opossum plugin is used. To compare the optimization results with the prediction results, multiple linear regression, support vector machines, random forest, polynomial regression algorithms, and artificial neural networks (ANN) are developed to predict outdoor thermal comfort performance targets on each collected data set with 2000 samples. Results showed that ANN models can provide more accurate predictions during the optimization process. The paper aims to discuss the way ML algorithms are applied and evaluated for ML-based optimization domains in design problems.

Keywords: *optimization, machine learning, universal thermal climate index, artificial neural network, kinetic shading devices.*

.

INTRODUCTION

Integrating machine learning (ML) models into architectural and urban design processes has gained significant attention recently. With increasing computational capabilities, ML-driven predictive models have been employed to optimize environmental performance, such as outdoor thermal comfort (Roman et al., 2020). As urbanization progresses, outdoor comfort has become vital, particularly in hot climate regions where pedestrian pathways and open spaces often cope with excessive heat and discomfort.

The reviewed literature showed recent works focused on limited datasets (Roman et al., 2020). Prediction models developed by collecting data from design models usually include limited design alternatives (Shi et al., 2024). Since heuristic search methods can navigate outside the boundaries of the collected samples to reach the best design alternative, evaluating the accuracy of prediction models only based on the collected samples may result in scenarios where the predicted outcome aligns with an unrealistic solution (Goldberg and Holland, 1988). Thus, the performance values may be incorrect, leading to focusing on the wrong parameters when making design decisions. Addressing this gap, this study investigates the accuracy and applicability of ML-based predictive models in supporting optimization processes and explores the potential of ML models to predict outdoor thermal comfort performance.

In the prior phase of this research, kinetic shading devices designed with rigid origami principles were optimized with the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) and the Radial Basis Function Optimization (RBFOpt) algorithm within the Opossum (Wortmann, 2017) plugin alongside the genetic algorithms. The results proved that adaptive shading devices could significantly improve pedestrian comfort by reducing UTCI values up to 20% during peak sunlight hours. However, the computational demand was a key limitation of the optimization process. Due to the number of design parameters and their values, search areas contain relatively high alternatives during optimization. ML algorithms are the most common methods for developing predictive models and discovering favorable design alternatives with significantly reduced computational time (Deb et al., 2002; Boussaïd et al., 2013). By integrating ML into the optimization workflow, this study aims to improve computational efficiency by decreasing the reliance on long simulations and iterative searches.

To achieve that, this paper presents a comparative evaluation of multiple ML algorithms, including Multiple Linear Regression (MLR), Support Vector Regression (SVR), Random Forest (RF), Gradient Boosting (GB), and Artificial Neural Networks (ANN), in predicting outdoor thermal comfort performance and compares the results with an optimization algorithm used in simulation-based optimization problems (RBFOpt).

METHODOLOGY

The proposed method includes a parametric model of the adaptive structure, analysis of the UTCI with LadyBug, data generation, various ML algorithms, and comparison using optimization with surrogate-based optimization (RBFOpt).

Parametric model

With today's technology, it is possible to simulate origami structures in the early design phase. In the proposed method, the kinetic device is designed parametrically with rigid origami principles on Grasshopper for Rhinoceros. Kangaroo Physics® (Piker, 2013) is integrated to simulate its kinetic ability. The shape is adaptive in terms of the sun's position. The model was arrayed on both sides of the pathway. It has a folding ability between 0 to 150 degrees (Figure 1). Besides folding, it can rotate to north-south (N-S) and east-west (E-W) axes.

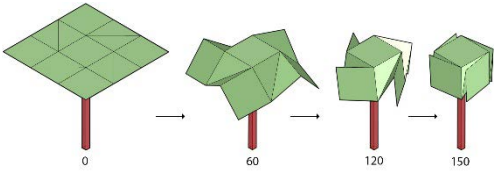


Figure 1
Kinetic shading device model

Table 1
Parameters of the shading device

	Fold 1	Fold 2	Copy 1	Copy 2	N-S 1	N-S 2	E-W 1	E-W 2	Z 1	Z 2
Value range	0-150	0-150	1-8	1-8	0-6.0	0-6.0	0-6.0	0-6.0	0-6.0	0-6.0

The origami structure can move in the z direction. Table 1 shows the parameters and their value range.

Performance simulation and data generation

For the next step of this framework, the kinetic model was tested through LadyBug (Sadeghipour Roudsari et al., 2013) tools on a pathway that measures 20 meters in length and 3 meters in width to analyze the UTCI results. The analysis was performed on one of the hottest days of the year, according to the EPW file of Izmir.

The model has 10 parameters, and a sinus function was formulated in Grasshopper for each of these parameters of the shading device. It allows each copy's form to experience a different adjustment with a single parameter change. This method demonstrated more adaptive shading behavior by changing rotation angle and height along each array on the sinusoidal curve.

A dataset of 2,000 samples was generated using Latin hypercube sampling (LHS) from the parametric model, and the UTCI results were collected through simulations using Ladybug tools for each sample.

Machine learning

The collected data is used to train ML models. Next, ML models are evaluated for accuracy in predicting UTCI values. MLR, SVR, RF, GB, and ANN algorithms were applied for each data set. While testing the machine learning algorithm, the data sets were divided into two learning and test sets at a rate of 0.2. This process was completed using Scikit-learn (Pedregosa et al., 2011), Keras (Chollet et al., 2015), and TensorFlow (Abadi et al., 2016) libraries in the Python programming language.

In the SVR model, a radial basis function (RBF) kernel was used, with the regularization parameter set to C=10 and the epsilon-insensitive loss set to 0.1. The number of estimators was selected as 100 in the random forest model, while default parameters were used in other algorithms. All models were trained and tested on standardized datasets reached through feature scaling. The accuracy of predictions on the test set was evaluated using the coefficient of determination (R^2), mean squared error (MSE), and mean absolute error (MAE).

The artificial neural network model had a 1-n-n-n-n-1 architecture with four hidden layers, each consisting of 200 neurons with the ReLU activation function. A rate of 0.1 dropout was applied after each hidden layer to prevent overfitting. The Stochastic Gradient Descent (SGD) algorithm was preferred for optimization in the model's learning process. MSE was used as the loss function, and MAE was used as the evaluation metric. The model was trained for 300 epochs with a batch size of 20, with 20% of the data reserved for validation to monitor the model's performance on unseen data.

Optimization with RBFOpt

A single objective optimization problem was formulated to minimize the UTCI using Opossum (RBFOpt). Figure 2 illustrates the result of the run.

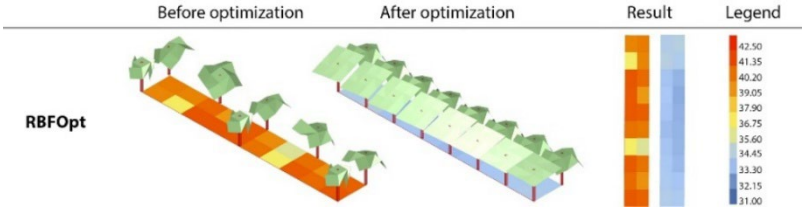


Figure 2
RBFOpt result

Unlike the many examples in the literature, the accuracy of the ML models was evaluated not only with the collected data sets but also with the parameters obtained from optimization that may be outside of the collected data set. Parameter sets of 1st, 2nd, 3rd, 500th, 1000th, 1500th, and 2000th iterations were predicted with ML models. Since this study was designed to test the accuracy of prediction models on a limited number of samples, the predicted results with ML may correspond to a much different result than the real one.

RESULTS AND DISCUSSION

In this section, the accuracy of the prediction models is discussed, and the machine learning results are compared with RBFOpt optimization results. Table 2 shows the first comparison of the prediction performances of machine learning algorithms. Each model was assessed using the R^2, MSE, and MAE for training and test datasets.

GB achieved the highest overall performance among the evaluated algorithms, with a test R^2 of 0.91, a low MSE of 0.13, and an MAE of 0.28. It also maintained a high training R^2 of 0.94, which indicates minimal overfitting. ANN also performed remarkably well, reaching a test R^2 of 0.91, with MSE and MAE values of 0.14 and 0.30. Similarly, RF presented a performance with an R^2 of 0.90, MSE of 0.15, and MAE of 0.29.

SVR and MLR produced moderate results in R^2. However, both had higher error values than GB, RF, and ANN. Among all, SVR performed the worst, and ANN was relatively good regarding R^2 score and prediction accuracy. So, ANN and SVR graphs are examined in Figure 3. Figure 3 demonstrates that ANN provided accurate and consistent predictions. In the ANN plot, the data points are closely clustered around the ideal diagonal line. It indicates a stronger correlation between predicted and actual values. The SVR plot also showed predictions close to the line, but there is a slightly wider spread than the ANN. For this reason, SVR has a slightly higher prediction error than ANN.

	R^2(test)	R^2(train)	MSE	MAE
MLR	0,89	0,90	0,14	0,29
SVR	0,88	0,94	0,17	0,30
RF	0,90	0,98	0,15	0,29
GB	0,91	0,94	0,13	0,28
ANN	0,91	0,93	0,14	0,30

Table 2
Evaluation of ML algorithms

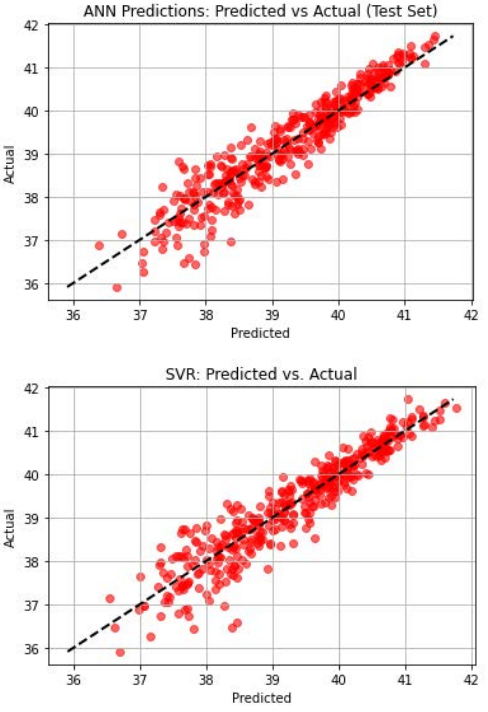


Figure 3
ANN and SVR graphs

After training the collected data and achieving these results, each ML model was utilized to predict the selected parameter sets from the RBF result. To examine different areas of the research space, the first three parameter sets and the 500th, 1000th,

1500th, and 2000th parameters were used, which achieved more optimal results.

Figure 4 illustrates the predicted UTCI values over selected iterations of the optimization process. The figure compares different machine learning models to the RBFOpt results. While most ML models exhibited relatively stable predictions across the first three iterations, they failed to capture the optimum range of UTCI achieved by the RBFOpt algorithm. The RBFOpt reached values below 34 by the 500th iteration and maintained a low range throughout the optimization process. In contrast, even the best-performing prediction models consistently overestimated the UTCI values for the same parameter sets. It highlighted the limitations of these models in extrapolating beyond the training data domain. This inconsistency shows the importance of validating predictive ML with optimization-generated parameters.

Figure 4
Accuracy of prediction models

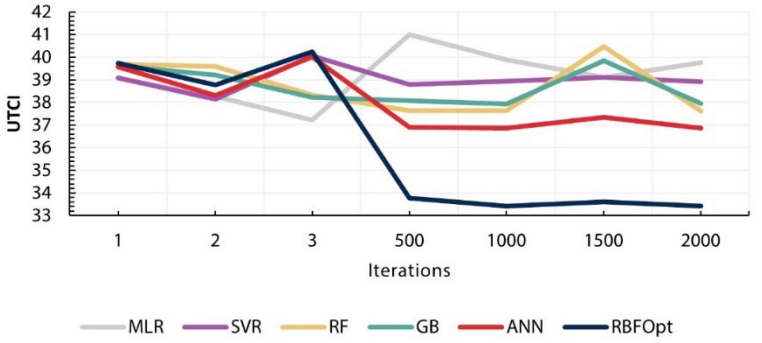
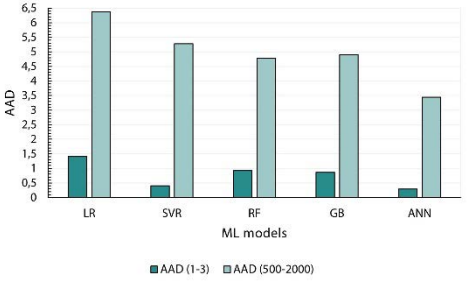


Figure 5 compares the average absolute difference (AAD) between predicted and actual UTCI values across all ML models. The graph is categorized into AAD for the early-stage iterations (1–3) and optimal range iterations (500–2000), as illustrated in Figure 5.

Figure 5
The average absolute difference between predicted and actual values



The AAD values between the 500th and 2000th iterations were significantly higher than the early-stage AAD (1–3) values for all models. In this study, ML models struggled to maintain accuracy when predicting the performance of optimized parameter sets. That means our data samples do not include optimal UTCI range parameter sets. The ANN model performed the lowest overall AAD (2,09) and performed best in early and late iterations. This shows that the ANN model has a higher generalization capacity than the other models.

Although the SVR model predicted the first three iterations well, it performed poorly in predicted results for the 500-2000 iteration range. Similarly, the LR model demonstrated the highest AAD values because of their limited ability to extrapolate to unseen parameter spaces. The ANN model has significantly better results, with 0,29 AAD for the first three iterations and 3,44 for the optimal range.

CONCLUSION

This study investigated the reliability of various machine learning algorithms in predicting outdoor thermal comfort performance for a kinetic shading device. Although an initial dataset is required, the proposed approach reduces simulation time during

optimization by enabling performance predictions across a large number of design configurations.

The study included a digitally produced kinetic model and building performance simulations. ANN achieved the highest predictive performance on test data among the proposed models and predicted parameter sets beyond the original data range better. However, all ML models showed poor accuracy for predicting optimal range iterations of optimization-generated configurations by the RBFOpt algorithm. The data obtained from the study showed that using the prediction models developed only from the collected samples can lead to wrong choices. While machine learning algorithms can make accurate predictions about the collected data, these predictions can lead to incorrect prediction results when they go beyond the data collected during the optimization process.

The methodology does not aim to build a generalizable model across all climates or orientations. Instead, it tried to enhance the optimization phase by offering reliable predictions. While each new context may require dataset adaptation, this approach supports efficient exploration in constrained design problems. Future studies may address generalizability limitations and a more accurate optimization process with ML.

These findings highlighted the importance of validating predictive ML models not only through conventional test sets but also in the context of optimization-driven exploration.

REFERENCES

Abadi, M., Barham, P., Chen, J., et al. (2016). TensorFlow: A system for large-scale machine learning. *arXiv preprint* [Online]. Available at: https://arxiv.org/abs/1605.08695 (Accessed: 7 June 2025).

Boussaïd, I., Lepagnot, J. and Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences*, 237, pp.82–117.

Chollet, F. et al. (2015). *Keras* [Software]. Available at: https://github.com/fchollet/keras (Accessed: 7 June 2025).

Deb, K., Pratap, A., Agarwal, S., et al. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2), pp.182–197.

Goldberg, D.E. and Holland, J.H. (1988). Genetic Algorithms and Machine Learning. *Machine Learning*, 3(2/3), pp.95–99.

Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12, pp.2825–2830.

Piker, D. (2013). Kangaroo: Form Finding with Computational Physics. *Architectural Design*, 83(2), pp.136–137.

Roman, N.D., Bre, F., Fachinotti, V.D., et al. (2020). Application and characterization of metamodels based on artificial neural networks for building performance simulation: A systematic review. *Energy and Buildings*, 217, 109972.

Sadeghipour Roudsari, M., Pak, M. and Viola, A. (2013). Ladybug: A parametric environmental plugin for Grasshopper to help designers create an environmentally-conscious design. In: *Proceedings of Building Simulation 2013: 13th Conference of International Building Performance Simulation Association*, Chambéry, France, pp.3128–3135.

Shi, C., Emadikhiav, M., Lozano, L., et al. (2024). Constraint learning to define trust regions in optimization over pre-trained predictive models. *INFORMS Journal on Computing*, 36(6), pp.1382–1399.

Wortmann, T. (2017). Opossum – Introducing and evaluating a model-based optimization tool for Grasshopper. In: Janssen, P., Loh, P., Raonic, A. and Schnabel, M.A. (eds.) *Protocols, Flows and Glitches – Proceedings of the 22nd International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA)*, Hong Kong, pp. 283–292. CAADRIA. DOI: 10.52842/conf.caadria.2017.283