

Comparison of Machine Learning Algorithms for Estimating the Natural Frequencies and Dynamic Response of a Planar Cantilever Truss Under Impulsive Loading

Mojtaba Shahraki*

Department of Civil Engineering, Faculty of Engineering, Ferdowsi University of Mashhad, Mashhad, Iran.

*Email: mojtaba.shahraki@alumni.um.ac.ir

ABSTRACT

Optimal design of steel trusses subject to natural-frequency constraints requires repeated dynamic analysis over a very large number of candidate geometries and member cross-sections, which is computationally prohibitive when relying on finite element analysis (FEA) alone. In this study, the performance of three machine learning algorithms - a feed-forward artificial neural network (ANN), a random forest, and a gradient boosting ensemble - is compared for building a surrogate model that simultaneously predicts the first three natural frequencies and the peak dynamic displacement of a ten-member, two-bay planar cantilever truss under a half-sine impulsive load. A complete finite element model, comprising assembly of the truss stiffness and mass matrices, solution of the generalized eigenvalue problem for the natural frequencies and mode shapes, and modal superposition for the dynamic response, was developed for this purpose. A dataset of 1500 samples was generated by randomly varying the bay length, truss height, and the cross-sectional areas of three member groups (chords, verticals, and diagonals), and all three machine learning models were trained and evaluated on this dataset. The results show that the artificial neural network achieves the highest accuracy for all four output variables (coefficient of determination R^2 above 0.99), followed closely by gradient boosting, while the random forest yields a somewhat lower accuracy. A feature-importance analysis further reveals that the bay length is the most influential design parameter affecting both the natural frequencies and the dynamic response. These findings provide a basis for replacing repeated direct dynamic analyses in frequency-constrained truss optimization with a fast and accurate machine learning surrogate.

Keywords: machine learning, planar cantilever truss, structural dynamic analysis, natural frequency, artificial neural network, random forest, gradient boosting, surrogate model, modal analysis, impulsive load

1. INTRODUCTION

Steel trusses are widely used in long-span structures such as bridges, industrial roofs, telecommunication towers, and large antenna structures, owing to their high stiffness-to-weight ratio. In many such applications, in addition to static safety requirements, natural-frequency constraints must also be satisfied, either to avoid resonance with operational or environmental excitations or to ensure occupant comfort. Truss optimization under frequency constraints is a well-known problem in structural engineering, in which the optimal member cross-sections are sought so as to minimize structural weight while satisfying prescribed frequency bounds.

The main computational challenge in such problems is the cost of repeated dynamic analysis, since metaheuristic optimization algorithms typically require thousands to millions of objective-function evaluations, making direct finite element analysis for every evaluation practically infeasible. To overcome this challenge, several researchers have applied metaheuristic algorithms to frequency-constrained truss optimization - for example, Dede and Togan employed a teaching-learning-based optimization algorithm for this purpose [3] - although such approaches still rely on direct analysis for every evaluation.

In recent years, building machine-learning-based surrogate models has been proposed as a way to reduce this computational cost. Yu used a back-propagation (BP) neural network to predict the natural frequency of a four-node, four-bar truss from a combination of five design parameters, and subsequently used the trained model for frequency optimization [7]. However, most existing studies employ a single machine learning algorithm (typically a neural network), and a systematic comparison among different machine learning algorithms - particularly alongside simultaneous prediction of dynamic response together with natural frequencies - has received comparatively little attention.

The aim of the present study is to develop a complete finite element model for the dynamic analysis of an example planar cantilever truss, and to compare the performance of three widely used machine learning algorithms - artificial neural network, random forest, and gradient boosting - in simultaneously predicting the first three natural frequencies and the peak dynamic displacement of this truss. The remainder of the paper is organized as follows: Section 2 reviews the theoretical background of truss dynamic analysis (stiffness and mass matrix assembly, the eigenvalue problem, and modal superposition); Section 3 describes the dataset generation procedure and the three compared machine learning models; Section 4 presents the numerical results and model comparison; and Section 5 concludes the paper.

2. THEORETICAL BACKGROUND

2.1 Truss Stiffness and Mass Matrices

The truss considered in this study is a planar, two-bay cantilever truss with six joints and ten pin-jointed members (Fig. 1). Each truss member carries axial force only, and its stiffness matrix in the local coordinate system, transformed to the global coordinate system using the direction cosines $c = \cos\theta$ and $s = \sin\theta$, is given by [2]:

$$k_e = (EA/L) \cdot \begin{bmatrix} [c^2, cs, -c^2, -cs], [cs, s^2, -cs, -s^2], [-c^2, -cs, c^2, cs], [-cs, -s^2, cs, s^2] \end{bmatrix} \quad (1)$$

where E is the elastic modulus, A is the cross-sectional area, L is the member length, and θ is the angle of the member with respect to the horizontal axis. By assembling the stiffness matrices of all ten members in the global coordinate system and removing the degrees of freedom restrained at the fixed supports, the reduced global stiffness matrix K of the structure is obtained. The mass matrix M is formed as a lumped mass matrix, assigning half of each member's mass (ρAL) to each of its two end nodes. The natural frequencies and mode shapes of the truss are then obtained from the following generalized eigenvalue problem:

$$(K - \omega_j^2 M) \cdot \varphi_j = 0, \quad j = 1, 2, \dots, n \quad (2)$$

where ω_j is the angular frequency and φ_j is the mode shape vector associated with the j -th mode. In this study, the mode shape vectors are mass-normalized such that:

$$\varphi_j^T \cdot M \cdot \varphi_j = 1, \quad j = 1, 2, \dots, n \quad (3)$$

2.2 Modal Superposition for Dynamic Response

To predict the dynamic response of the truss under a half-sine impulsive load applied at a specified joint (Fig. 1), the modal superposition method is used, retaining the first n_m modes. Owing to mass normalization, the equation of motion for each modal coordinate $q_j(t)$ can be written as an independent single-degree-of-freedom system:

$$\ddot{q}_j(t) + \omega_j^2 \cdot q_j(t) = P_j(t), \quad P_j(t) = \varphi_j^T \cdot F(t) \quad (4)$$

where $F(t)$ is the dynamic load vector (in this study, a single non-zero entry corresponding to the loaded degree of freedom, equal to $F_0 \sin(\pi t/t_d)$ for $0 \leq t \leq t_d$) and $P_j(t)$ is the corresponding modal force. After numerically solving Eq. (4) for each mode, the physical displacement of any degree of freedom is reconstructed via modal superposition:

$$u(t) = \sum_{j=1}^{n_m} \varphi_j \cdot q_j(t) \quad (5)$$

In this study, the first six modes ($n_m = 6$) are retained in the modal superposition, and the peak absolute displacement of the loaded joint over the simulation time window is taken as the target quantity “peak dynamic displacement.”

3. METHODOLOGY

3.1 Dataset Generation

To generate the dataset, the geometry of the truss shown in Fig. 1 - with bay length L_b and height h - and the cross-sectional areas of the three member groups (top and bottom chords, verticals, and diagonals) were randomly varied, with a uniform distribution, within the ranges listed in Table 1. The truss material was taken as steel, with the elastic modulus and density subject to a limited random variation about their reference values. For each of the 1500 generated samples, the reduced stiffness and mass matrices [Eq. (1)] were assembled, the eigenvalue problem [Eqs. (2)-(3)] was solved to extract the first three natural frequencies, and the dynamic response [Eqs. (4)-(5)] was computed for a fixed impulsive load to obtain the peak displacement. The complete implementation of this step - including stiffness/mass assembly, the eigenvalue solution, and modal superposition - is openly available in the code repository described in the Code and Data Availability section (module `truss_fe.py`).

Table 1. Ranges of the input parameters used to generate the dataset

Parameter	Symbol	Unit	Range
Bay length	L_b	m	1.5 - 3.0
Truss height	h	m	1.5 - 3.0
Chord cross-sectional area	A_{chord}	cm^2	8 - 30
Vertical member cross-sectional area	A_{vert}	cm^2	5 - 20
Diagonal member cross-sectional area	A_{diag}	cm^2	5 - 20
Steel elastic modulus	E	GPa	190 - 210 ($\pm 5\%$)
Steel density	ρ	kg/m^3	7615 - 8085 ($\pm 3\%$)
Impulse amplitude and duration	F_0, t_d	N, s	fixed: 5000, 0.03

Fig. 1 shows an example of the truss geometry, the support conditions, and the location and direction of the applied impulsive load.

Two-bay, ten-member planar cantilever truss (example geometry)

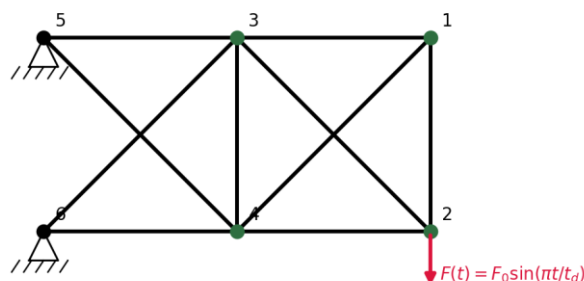


Fig. 1. Example geometry of the two-bay, ten-member planar cantilever truss considered in this study

3.2 Machine Learning Models Compared

Three machine learning algorithms were used to build the surrogate model and were compared with one another. First, a feed-forward artificial neural network with two hidden layers (32 and 16 neurons, ReLU activation) was used. Second, a random forest, which averages the predictions of a large number of independently trained decision trees over random subsets of the data:

$$\hat{y}(x) = (1/T) \cdot \sum_{t=1}^T h_t(x) \quad (6)$$

where T is the number of trees and $h_t(x)$ is the prediction of tree t [5]. Third, gradient boosting, which builds the final model incrementally by progressively correcting the residual error of the previous models:

$$F_m(x) = F_{m-1}(x) + v \cdot h_m(x) \quad (7)$$

where $F_m(x)$ is the ensemble model after adding the m -th weak learner, $h_m(x)$ is a model trained to reduce the gradient of the residual loss function, and v is the learning rate [6]. Table 2 summarizes the main hyperparameters of the three models. For all three models, the same 80/20 training/test split and the same input/output standardization were used, to ensure a fair comparison. The complete definition, training, and evaluation of all three models is openly available in the code repository (module train_models.py).

Table 2. Main hyperparameters of the three compared machine learning models

Model	Main hyperparameters
Artificial Neural Network (ANN)	Two hidden layers (32 and 16 neurons), ReLU activation, Adam optimizer, early stopping
Random Forest (RF)	Number of trees = 300, maximum depth = unlimited
Gradient Boosting (GB)	Number of boosting stages = 300, maximum depth = 3, learning rate = 0.05

4. RESULTS AND DISCUSSION

After training all three models on 1200 samples and evaluating them on 300 test samples, the mean absolute error (MAE), root mean square error (RMSE) and coefficient of determination (R^2) were computed for all four output variables. Table 3 reports these metrics for the three models.

Table 3. Comparison of accuracy metrics for the three machine learning models on the test set

Model	Output variable	MAE	RMSE	R^2
ANN	f1 (Hz)	0.932	1.291	0.994
ANN	f2 (Hz)	1.789	2.310	0.996
ANN	f3 (Hz)	1.892	2.684	0.994
ANN	y max (mm)	0.026	0.037	0.990
Random Forest (RF)	f1 (Hz)	2.233	2.983	0.970
Random Forest (RF)	f2 (Hz)	5.819	7.889	0.949
Random Forest (RF)	f3 (Hz)	5.946	7.655	0.949
Random Forest (RF)	y max (mm)	0.046	0.068	0.969
Gradient Boosting (GB)	f1 (Hz)	1.385	1.897	0.988
Gradient Boosting (GB)	f2 (Hz)	5.138	6.755	0.962
Gradient Boosting (GB)	f3 (Hz)	4.039	5.293	0.976
Gradient Boosting (GB)	y max (mm)	0.032	0.046	0.986

As shown in Table 3, the artificial neural network achieves the highest coefficient of determination and the lowest error for all four output variables (R^2 above 0.99). Gradient boosting ranks second, performing particularly close to the neural network for the first natural frequency ($R^2 \approx 0.988$). The random forest, although

still reasonably accurate (R^2 between 0.949 and 0.970), is noticeably less accurate than the other two models; this can be attributed to the continuous and relatively smooth nature of the relationship between the truss geometric/cross-sectional parameters and its dynamic quantities, which gradient-based models (the neural network and gradient boosting) tend to fit better than partition-based models (random forest). Fig. 2 illustrates this comparison graphically for all four output variables.

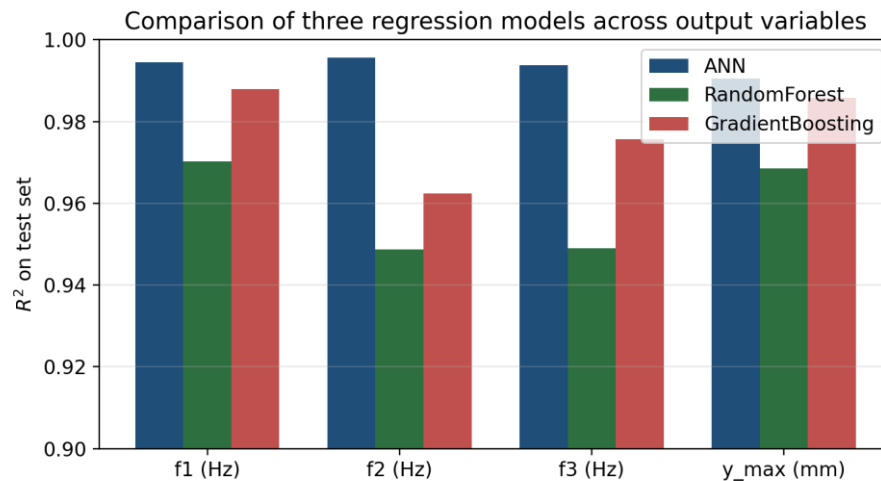


Fig. 2. Comparison of the coefficient of determination (R^2) of the three machine learning models for the four output variables

To identify the most influential design parameters affecting the dynamic behavior of the truss, the relative importance of each of the seven input parameters was extracted from the random forest model (which provides this metric directly) and averaged over the four output variables, as shown in Fig. 3. The truss bay length (Lb) is by far the most influential parameter, which is consistent with the quadratic dependence of geometric stiffness on span length as well as the associated increase in overall structural mass. The truss height and the chord cross-sectional area follow in importance, while the elastic modulus and material density - which in this study were varied only within a limited range about the reference steel properties - show the smallest influence.

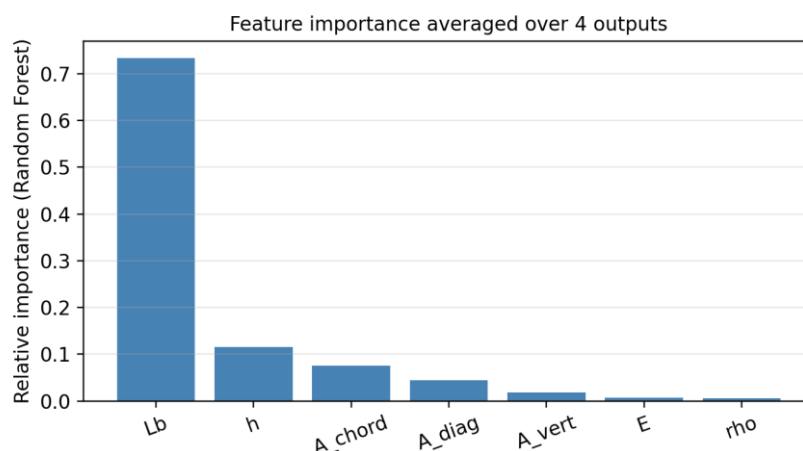


Fig. 3. Relative importance of the input parameters based on the random forest model

Fig. 4 shows the predicted versus actual values obtained from the best-performing model (the artificial neural network) for the first natural frequency and the peak dynamic displacement, on the test set. The close clustering of points around the 45° line for both quantities confirms, qualitatively, the quantitative results of Table 3.

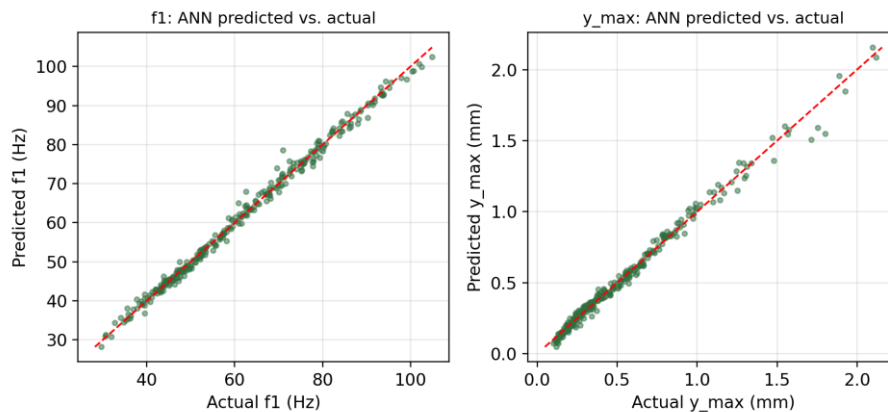


Fig. 4. Predicted versus actual values from the artificial neural network for the first natural frequency (left) and the peak dynamic displacement (right)

5. CONCLUSION

In this study, a complete finite element model was developed for the dynamic analysis of a ten-member, two-bay planar cantilever truss, and used as the basis for comparing three machine learning algorithms - artificial neural network, random forest, and gradient boosting - in simultaneously predicting the first three natural frequencies and the peak dynamic displacement of this truss, on a dataset of 1500 samples.

The results showed that the artificial neural network was the most accurate model for all four output variables (R^2 above 0.99), followed by gradient boosting and then random forest. The feature-importance analysis further showed that the truss bay length is the most influential parameter affecting the natural frequencies and dynamic response. Given the high accuracy and the very fast execution of the proposed surrogate model compared to direct finite element analysis, this approach can effectively replace or complement direct analysis in frequency-constrained truss optimization processes that require repeated evaluation over a large number of configurations.

Limitations of this study include the fixed truss topology (only geometric and cross-sectional parameters were varied), the omission of structural damping, and the consideration of only a single impulsive loading pattern. Future work is suggested to consider variable truss topologies (number of bays and member arrangement), the effect of damping, more diverse dynamic loading patterns (such as moving loads or earthquake records), and additional machine learning algorithms (such as XGBoost and graph neural networks).

CODE AND DATA AVAILABILITY

The complete Python implementation used in this study - including the finite element model (truss_fe.py), dataset generation (generate_dataset.py), model training and comparison (train_models.py), and figure generation (make_figures.py) - together with the generated dataset, trained-model metrics, and all figures, is openly available at the following GitHub repository:

<https://github.com/MojtabaShahraki/truss-ml-dynamic-analysis>

The repository includes a README file with step-by-step instructions for reproducing every number and figure reported in this paper (Tables 1-3 and Figs. 1-4) from scratch.

REFERENCES

1. Chopra, A.K. (2017). Dynamics of Structures: Theory and Applications to Earthquake Engineering, 5th ed. Pearson, Boston, USA.
2. Logan, D.L. (2017). A First Course in the Finite Element Method, 6th ed. Cengage Learning, Boston, USA.
3. Dede, T., Togan, V. (2015). A teaching learning-based optimization for truss structures with frequency constraints. Structural Engineering and Mechanics, vol. 53, no. 4, p. 833-845.
4. Adeli, H., Yeh, C. (1989). Perceptron learning in engineering design. Computer-Aided Civil and Infrastructure Engineering, vol. 4, no. 4, p. 247-256.
5. Breiman, L. (2001). Random forests. Machine Learning, vol. 45, no. 1, p. 5-32.
6. Friedman, J.H. (2001). Greedy function approximation: a gradient boosting machine. Annals of Statistics, vol. 29, no. 5, p. 1189-1232.
7. Yu, Z. (2021). Frequency prediction and optimization of truss structure based on BP neural network. Proceedings of the 2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM2021), Manchester, UK, p. 354-360.
8. Goodfellow, I., Bengio, Y., Courville, A. (2016). Deep Learning. MIT Press, Cambridge, MA, USA.