Proceedings of the Design Society, Volume 5: ICED25
https://doi.org/10.1017/pds.2025.10335

Towards precision in bolted joint design: a preliminary machine learning-based parameter prediction

Ines Boujnah*†, Nehal Afifi*✉†, Andreas Wettstein and Sven Matthiesen

Karlsruhe Institute of Technology (KIT), Germany

✉ nehal.afifi@kit.edu

ABSTRACT: Bolted joints are critical for maintaining structural integrity and reliability. Accurate prediction of parameters is essential for optimal performance. Traditional methods often fail to capture the non-linear behavior or require significant computational resources, limiting accuracy and efficiency. This study addresses these limitations by combining empirical data with a feed-forward neural network. Leveraging experimental data and systematic preprocessing, the model effectively captures nonlinear relationships, including rescaling output variables to address scale discrepancies, achieving 95% predictive accuracy. While limited dataset size restricts generalizability, the findings demonstrate the potential of neural networks as a reliable, efficient alternative for bolted joint design. Future work aims to expand datasets and explore hybrid modeling techniques to enhance applicability.

KEYWORDS: Machine learning, Functional modelling, Artificial intelligence, Data-driven design, Bolted Joints

1. Introduction

Bolted joints, the most versatile machinery element (Roloff et al., 2023), are essential in technical devices and everyday applications, functioning as detachable connections between components (Kloos and Thomala, 2007). These joints facilitate load transmission between connected components (VDI 2230, 2015) by creating a preload force, achieved through the application of torque to either the head of the bolt or the nut (Oberg, 2013). In the design process of bolted joints subjected to tensile loads, the preload force is a critical parameter influencing their functional behavior and reliability (Bickford, 1995). The friction coefficients in the head and thread regions of the bolt are equally significant, as the majority of the tightening torque is expended in overcoming friction in these areas (VDI 2230, 2024). This friction not only contributes to the generation of preload force but also provides a self-locking effect, ensuring the joint remains secure under working conditions. As external loads are applied, the joint transitions through elastic and plastic deformation stages, potentially culminating in failure if critical load thresholds are exceeded (Steinhilper et al., 2012). Poorly designed bolts may experience excessive stresses during tightening, increasing the risk of joint failure, equipment malfunction, or operational downtime (Kloos and Thomala, 2007). Thus, understanding the relationships between tightening variables like preload force and functional behavior parameters namely load-bearing capacity, and friction coefficients accurately and efficiently is essential for achieving a reliable design.

1.1. Related work

Bolted joint behavior has traditionally been analyzed using a variety of modeling approaches. Analytical, numerical, empirical, and data-driven methods have all been extensively applied to investigate the behavior of bolted joints. (VDI 2230, 2015; Chiang and Chiang, 2023), each offering distinct strengths

* These authors contributed equally to this work.
† Joint first authorship.

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

and facing inherent limitations. This section provides an overview of these approaches, highlighting their contributions and challenges, and identifies the research gap that motivates the proposed hybrid methodology. **Analytical methods** approximate bolted joint behavior through simplified mechanical representations, focusing on specific aspects of bolted joints. For instance, Lee et al. (2022) evaluates the tightening torque-clamping force relationship and friction coefficients, validating theoretical calculations through experimental measurements. Zeng et al. (2020) proposed a mathematical equation, derived from mechanical behavior, to predict the load-bearing capacity of beam-to-column joints in slim-floor composite frames. While analytical methods provide valuable insights into fundamental mechanics, they often lack the ability to capture the complex, non-linear interactions inherent in bolted joints.

To address these limitations, **numerical methods** such as finite element analysis (FEA), enable detailed representation of nonlinear material behavior, constraints, and complex geometries (VDI 2230, 2014). They have been widely applied to study the mechanical behavior of bolted joints. Fukuoka and Takaki (1998) focused on tension and torque characteristics during and after tightening through experimental and numerical methods. Yu et al. (2015) explored how factors like friction, pitch, elastic modulus, and strain-hardening impact the tightening torque and initial load in bolted connections, using FEA to model these influences. Similarly, Liang et al. (2024) examined load capacity and fracture behavior in flange bolts under various load cases, providing insights into joint performance under different conditions. Although numerical approaches like FEA excel at modeling detailed mechanical behaviors, they often focus on idealized conditions, demand significant computational resources and rely heavily on the quality of the model and the expertise of the user to achieve accurate predictions (VDI 2230, 2014).

To bridge the gap between simulation and real-world performance, **empirical methods** such as Design of Experiments (DoE) have been employed to systematically investigate factors influencing bolted joint behavior. Nassar et al. (2005, 2007) studied the effects of friction coefficients, tightening speed, and coating on the torque-tension relationship and wear patterns, while Holch et al. (2023) examined load-bearing behavior in bolts and lockbolts under combined loading. While DoE offers a systematic approach to studying the effects of multiple factors on bolted joint behavior, and capable of accounting for non-linearities to some degree, its implementation is not without challenges. The method requires carefully controlled experimental conditions to ensure repeatability and reliability of results (Chiang and Chiang, 2023). As the number of variables increases, the experimental design becomes increasingly complex, often demanding significant time and resources to execute (Selvamuthu and Das, 2024). Additionally, external factors can introduce variability, reducing the robustness of the findings. The dependency of results on specific factors further restricts their applicability across different configurations (Wettstein and Matthiesen, 2020). Although their widespread use, they face inherent challenges in accurately predicting the functional behavior variables of bolted joints. Nonetheless, DoE remains a valuable method when used with other approaches to validate findings and enhance model accuracy (Afifi et al., 2024).

Furthermore, **data-driven predictive modeling** has emerged as a promising alternative to overcome these limitations. By leveraging relationships between input features and output variables, data-driven models can capture complex nonlinear behaviors and make predictions about unseen scenarios, enabling more accurate and efficient predictions and optimizations of bolted joint performance (Montans et al., 2019). For instance, Fernandez-Ceniceros et al. (2012) used a neural network ensemble with FEA data to predict load capacity, while Zhong et al. (2021) combined FEA and neural networks for bearing capacity optimization in aluminum joints. Fei et al. (2016) modeled bolt force in flanges, analyzing the effects of bending and shear, and Coelho et al. (2024) used Machine Learning (ML) to classify and quantify torque loss due to vibration. Other studies, such as those by Yildirim et al. (2019), Ren and Sun (2023), and Li et al. (2024), employed neural networks and ML algorithms to predict load capacity and other performance metrics in varied joint contexts, leveraging FEA and experimental data. Chen et al. (2020) and Olejnik and Ayankoso (2023) further integrated optimization algorithms with neural networks for enhanced predictive precision, and Atta et al. (2019) focused on predicting failure stages in bolted joints with a neural network. This approach not only enhances the ability to predict critical parameters but also offers significant advantages in terms of time efficiency, accuracy, and ease of implementation. Yet they require high-quality datasets for accurate training and validation. Recent research highlights the potential of combining empirical data with data-driven methods to address these challenges (Afifi et al., 2024). This study aims to explore this synergy, offering a framework that integrates experimental data with a feed-forward neural network model to enhance predictions of bolted joint performance.

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

1.2. Problem formulation and task description

The reviewed studies illustrate the potential of data-driven modeling to advance the analysis of bolted joints. While these approaches offer accurate predictions of critical parameters, their integration with empirical data remains underexplored. A significant gap exists in methodologies that effectively combine the strengths of empirical experimentation and data-driven techniques to simultaneously predict load capacity and friction coefficients while capturing the nonlinear behavior of bolted joints.

This research aims to address this gap by leveraging the complementary advantages of empirical data and data-driven predictive modeling to accurately and efficiently predict critical function behavior parameters. It focuses on bolted joint behavior under tensile loading conditions under a torque-controlled tightening process, employing a supervised feed-forward neural network trained on key parameters, including bolt size, strength grade, tightening torque, head torque, thread torque, and preload force, this effectively models nonlinear relationships between inputs and outputs in a computationally efficient manner. This approach overcomes the limitations of traditional methods like analytical models, DoE, and FEA, which are hindered by idealized assumptions, scalability issues, and high computational demands. By integrating empirical data with predictive modeling, the study enhances accuracy, efficiency, and reliability, contributing to practical design applications.

2. Feedforward neural network background

Data-driven predictive modeling aims to uncover relationships between input and output features, enabling accurate predictions on unseen data (Montáns et al., 2019). Within this field, ML focuses on developing algorithms capable of learning patterns from data to make informed predictions, with supervised ML specifically predicting target variables by analyzing input-output pairs (Zhou, 2021; LeCun et al., 2015). Regression, as a key type of supervised learning problem, focuses on estimating numerical values based on input features (Bengio et al., 2016). Linear regression handles simple relationships, whereas feed-forward neural networks handle complex patterns by processing information in a unidirectional flow, mapping inputs to outputs without cycles (Yadav et al., 2015). These networks feature a layered structure comprising an input layer, one or more hidden layers, and an output layer (Zhou, 2021). Each layer consists of interconnected units called neurons, which represent input and output features (Santry, 2024; Wythoff, 1993). The number of neurons in the hidden layers is determined by the network design and task complexity (Bengio et al., 2016), enabling the processing of signals as outputs from one layer become inputs for the next (Santry, 2024; Apicella et al., 2021). Each layer‘s output is computed as the weighted sum of inputs plus a bias, with an activation function introducing non-linearity to enable complex mappings (Kuhn, 2013). Commonly used activation functions in neural networks include the sigmoid and Rectified Linear Unit (ReLU), while weight initialization methods often involve random or Xavier initialization (Santry, 2024).

In the context of optimization, the primary objective is to identify the optimal parameters of the network that minimize the loss function, which is a metric used to quantify the cost between the predicted values generated by the model and the actual target values (Montesinos Lopez et al., 2022). Gradient descent is one of the most commonly used algorithms for optimization. It works by iteratively updating the model‘s parameters in the direction opposite to the gradient until reaching a local or global minimum (Ruder, 2016). Adam, a gradient-based algorithm for stochastic objective functions, adapts the learning rate by computing individual scaling factors for different parameters based on exponentially moving averages of the first and second moments of the gradients (Kingma and Ba, 2014). The Huber loss represents one of the frequently utilized loss functions (Sadouk et al., 2020). The principal objective of training a neural network is to fit its predicted outputs with the target outputs (Santry, 2024). The testing phase evaluates the trained model‘s accuracy by generating predictions from the input data and calculating the error between the predicted and target outputs. (Santry, 2024; Bengio et al., 2016). Overfitting arises when a model performs exceptionally well on the training dataset but fails to generalize to unseen data, leading to poor performance on the test set. Conversely, underfitting occurs when a model performs poorly even on the training data set, suggesting that it has failed to capture the essential patterns in the data (Müller and Guido, 2016). Generalization is reached, when the error has been reduced to an adequately low level, thereby indicating that the model has acquired the capacity to make accurate predictions (Santry, 2024).

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

3. Methodology

This section outlines the methodology for predicting key parameters in bolted joint design. Experimental data was processed and essential features were selected. A feed-forward neural network was implemented to model the complex relationships in bolted joint behavior. The fundamental methodology is illustrated in Figure 1.

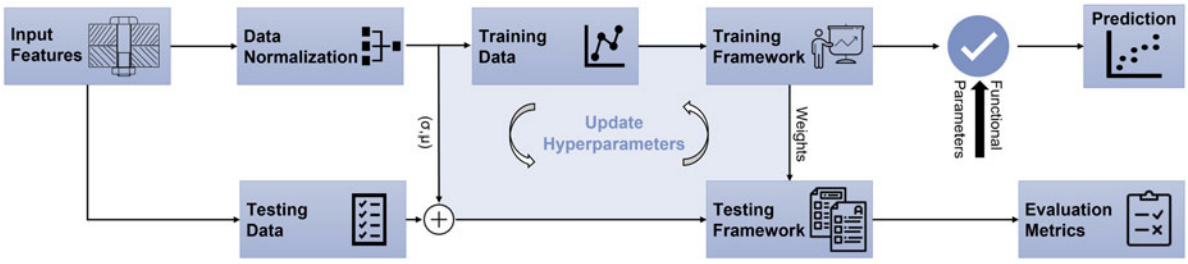


Figure 1. Task Description

3.1. Data description and feature selection

To establish a rationale for the chosen data, it is essential to consider the factors influencing bolted joint performance. The friction at the bolt surface and the load distribution are influenced by the bolt size, with larger bolts promoting a more uniform load distribution and greater friction generation (VDI 2230, 2015). Additionally, the strength grade of a bolt plays a crucial role in determining its load capacity and how stress is distributed throughout the material (Kloos and Thomala, 2007). Furthermore, head torque and thread torque are both directly proportional to the friction coefficients. The remaining tightening torque contributes to the generation of preload force, with the applied torque magnitude directly affecting the load introduced to the bolt (VDI 2230, 2024). Based on these considerations, the selected input parameters are bolt size, strength grade, preload force, tightening torque, head torque, and thread torque, while the output parameters are those influencing the functional behavior of the bolted joint, specifically load capacity and head and thread friction coefficients.

The experimental data were obtained from two sets of controls with the described experimental setup in Wettstein et al. (2020), each involving the continuous tightening of a bolted joint consisting of a bolt, two connected plates, a washer, and a nut. The initial dataset was gathered for an M10 8.8 ISO 4014 bolted connection, as described by Wettstein and Matthiesen (2020). The second dataset involved an M6 8.8 ISO 4017 bolted joint, obtained with the same methodology. Time series data consisting of preload force, tightening torque, head torque, and thread torque were measured. For the M6 bolt, a preload force of 8 kN was used, with 20 samples collected. The M10 model utilized a preload force of 12.5 kN, with 9 samples available, while the M10 bolt with a preload force of 25 kN included 5 samples. From these values, the friction coefficients of the head and threads and the remaining load capacity were further empirically estimated. The set finally had 34 samples for training and testing the neural network.

3.2. Data preprocessing

The data was presented in tabular format, representing time series measurements of the variables. The dataset was divided into two subsets: a training (80%) and a test (20%) set. Input and output features were subsequently extracted according to predefined feature specifications. Normalization was applied as a scaling method to transform the input features, minimizing the influence of skewness and outliers while ensuring that the data remained within an appropriate range for the model. The scale derived from the training set was retained to ensure consistent application during subsequent stages of analysis. The preprocessing procedures applied to the test set mirrored those of the training set, with a critical distinction: any scaling of the test data was performed using the parameters obtained from the training set. This approach ensured that the model operated on data with a consistent scale, thereby preserving the validity of the training and evaluation processes.

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

3.3. Model architecture

The input layer contained six nodes, representing bolt size, strength grade, tightening torque, head torque, thread torque, and preload force. The neural network included two hidden layers, with neuron counts matching the input and output layers. The network's output layer has three nodes, corresponding to the output features: load capacity, head friction coefficient, and thread friction coefficient (Figure 2).

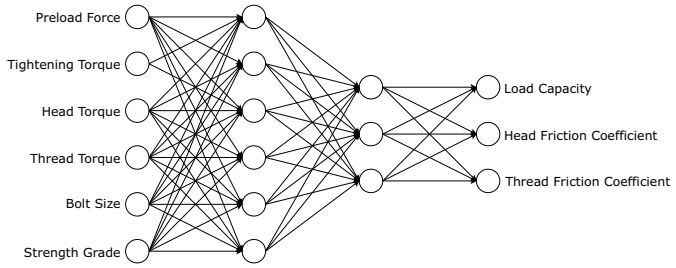


Figure 2. Visualization of the Used Network Architecture and the Input and Output Features

A total of 136 models were trained throughout the process. Four representative models were selected to illustrate the learning and optimization process. These models were evaluated with variations in key hyperparameters, including the activation function, weight initialization method, number of epochs, scaling method, units for preload force and load capacity, and the number of samples used (Table 1). The activation function determined the scale of outputs, while the weight initialization method affected training speed, convergence, and final accuracy. The number of epochs ensured sufficient training to generalize without overfitting. The scaling method aligned the data within an optimal range, reducing outliers and improving convergence. Choosing appropriate units for preload force and load capacity ensured output compatibility, and a larger dataset size enhanced pattern recognition.

Table 1. Hyperparameters of Chosen Models

Hyperparameters	Model 1	Model 2	Model 3	Model 4
Activation Function	Sigmoid	ReLU	Sigmoid	Sigmoid
Number of Epochs	1000	1000	4800	4800
Unit [Preload/Load]	N/N	kN/kN	kN/kN	kN/MN

The initialization stage involved configuring all hyperparameters prior to the training process, establishing the foundation for learning. These included the activation function, initialization method for weights and biases, network width and depth, proportion of data used for training and testing, number of epochs, batch size, loss function, optimization algorithm, and learning rate. The neural network structure was created by defining the total number of layers and the number of nodes per layer.

The training process commenced following the initialization of the neural network. The training dataset was loaded and shuffled to ensure varied data processing during each iteration, improving efficiency. The loss function, optimization algorithm, and learning rate were then configured. Over a specified number of epochs, the training loop iterated with a defined batch size, through the data. The network generated outputs from the input data in each iteration, calculated the loss, and updated learnable parameters based on the chosen optimization algorithm. Mean accuracy and loss values were computed and plotted after each epoch to track progress and evaluate performance throughout training. This process is described in Figure 3 by the blue and black path and by Algorithm 1.

The testing process involved loading the unshuffled test dataset for evaluation, limited to a single epoch and batch due to the smaller size of the test set, ensuring computational efficiency. The neural network computed outputs based on the input data and model accuracy was assessed for each data point, with the mean accuracy calculated for the overall model accuracy. Predictions were deemed accurate if they deviated by no more than 5% from the target values. To evaluate performance, error metrics such as

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

Algorithm 1 Training Framework

```
for epoch < number_of_epochs do
    for batch < batch_size do
        optimizer.zero_grad()
        predicted_output ← model(batch_input)
        loss ← loss_function(predicted_output, batch_output)
        Call loss.backward()
        Call optimizer.step()
    end for
end for
```

Mean Absolute Error (MAE), Mean Square Error (MSE), and Root Mean Square Error (RMSE) were computed for each data point. A scatter plot was also generated, displaying predicted outputs against target values, where accurate predictions aligned along a linear function through the origin. This process is described in Figure 3 by the green and black path and by Algorithm 2.

Algorithm 2 Testing Framework

```
With torch.no_grad
for batch < batch_size do
    model.eval()
    predicted_output ← model(batch_input)
end for
```

The evaluation and hyperparameter tuning process involved analyzing the accuracy and loss curves from training, as well as the error metrics and accuracy curves from testing, to assess the model's alignment with the data. Hyperparameters were iteratively adjusted until satisfactory accuracy was achieved. Once optimized, the trained model parameters were saved for future use.

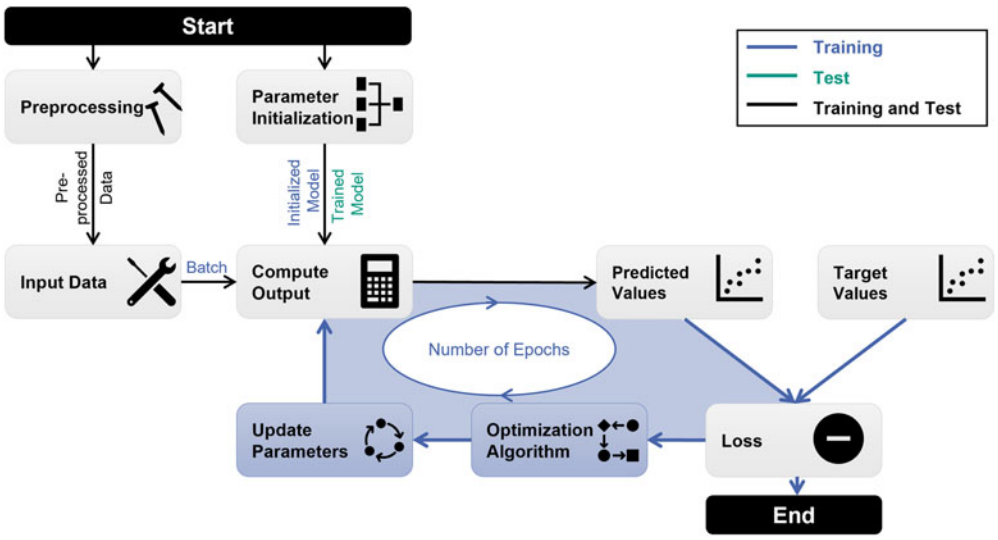


Figure 3. Visualization of the Implemented Workflow

4. Results

As the hyperparameters tuning process progressed, the accuracy of each model demonstrated an incremental improvement, accompanied by an increase in the elapsed time required for completing all steps, from preprocessing to training. The final accuracy attained for Model 4 was 95.24%, with a generation time of 90 seconds (Table 2).

The models were trained using Adam with a learning rate of 0.01, aiming to optimize convergence speed and minimize loss. The Huber loss function was applied to balance sensitivity to outliers and improve stability. The training was conducted with a batch size of 4, enabling incremental updates to the model's parameters. The Xavier weight initialization method was used to assign unique weights within a specific range, influencing the learning process by optimizing training speed, convergence rate, final accuracy, and overall model efficiency. Normalization was applied as a scaling method to minimize the influence

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

Table 2. Results of Chosen Models

	Model 1	Model 2	Model 3	Model 4
Accuracy [%]	9.52	33.33	61.90	95.24
Elapsed Time [s]	8.89	9.15	34.86	90.11

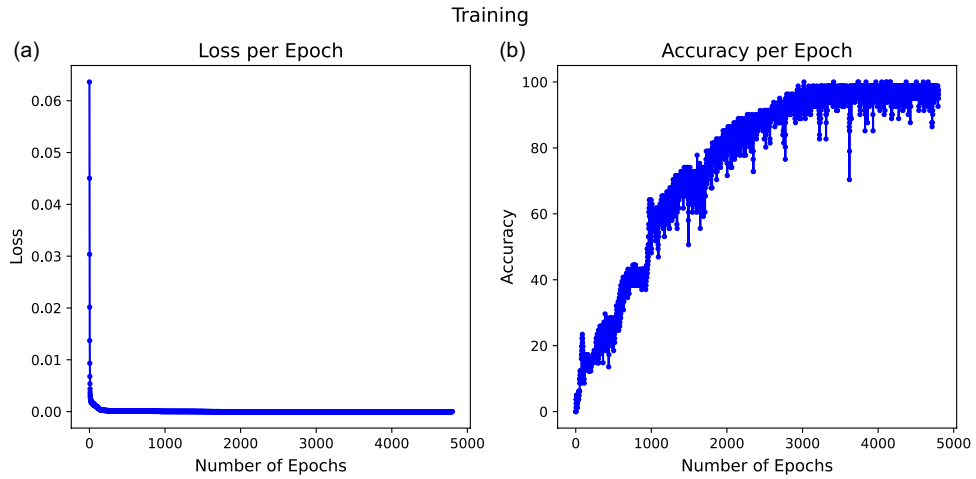


Figure 4. Results of Model 4: Training (a) Loss per Epoch (b) Accuracy per Epoch

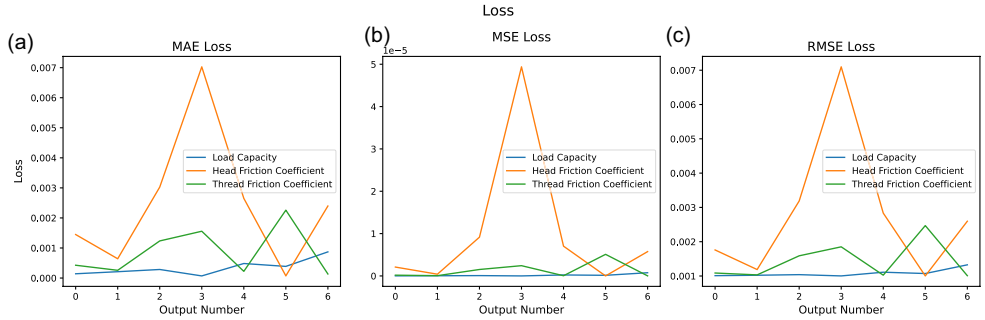


Figure 5. Results of the Model 4: Loss (a) MAE Loss (b) MSE Loss (c) RMSE Loss

of skewness and outliers, ensuring that input data remained within an appropriate range for the model. This preprocessing step optimized the activation function's performance, facilitating faster convergence and improving control over the data fed into the model. The bias was initialized to a constant value of zero for each neuron. After the hyperparameters tuning, the training curves followed the typical pattern for the loss and accuracy, with the loss decreasing logarithmically almost to zero and the accuracy increasing exponentially to a maximum of around 100% over the epochs (Figure 4). The loss between the predicted and actual values was nearly negligible for all metrics (MAE, MSE, RMSE), except one of the head friction coefficient values, which exhibited a considerable deviation from the other values as shown in Figure 5. Furthermore, in Figure 6 the predicted values for all output parameters aligned closely with the reference line through the origin for the scatter plot of the predicted to the target values. The load capacity and thread friction coefficients achieved 100% accuracy, while the head friction coefficients reached 85.71%, with only one value showing a minimal deviation from the line. All other values are within the 5% range of the blue line.

5. Discussion

This study achieves significant improvements in predicting load capacity and friction coefficients in bolted joints by addressing challenges in model architecture and preprocessing. A primary challenge was the scale discrepancy between the output variables, specifically load capacity and friction coefficients.

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

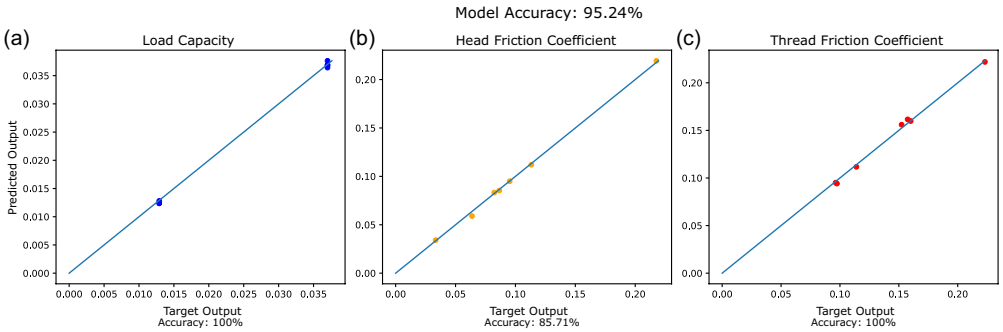


Figure 6 Results of Model 4: Test (a) Scatter of the Load Capacity (b) Scatter of the Head Friction Coefficients (c) Scatter of the Thread Friction Coefficients

Rescaling the load capacity to MN effectively aligned the scales, minimized noise in the loss function, and enhanced the stability of the training process, which ultimately led to the accuracy achieved. Furthermore, the adoption of the sigmoid activation function, appropriately matched to the range of the output variables, facilitated balanced prediction accuracy across all parameters. These results underscore the importance of systematic preprocessing and model optimization in developing robust predictive tools for complex engineering systems.

The dataset used in this study, consisting of only 34 samples, presents limitations in terms of generalizability and the stability of results across different bolt configurations. Since the model was trained using only two bolt sizes and three preload force cases, its direct transferability to other configurations is not ensured. While the model demonstrated high accuracy for certain cases, the limited diversity of the dataset raises concerns about overfitting, and the performance should, therefore, be interpreted with caution. To improve the generalizability of the model, future work will focus on expanding the data set to include a wider variety of bolt configurations and operational conditions, ensuring that the model can make reliable predictions for bolts in active use. Furthermore, we plan to explore techniques to generate synthetic data and augment the dataset through rule-based formulations of bolt behavior like VDI 2230 (2015) or FEA (VDI 2230, 2014). This data-driven approach could prove valuable, as the acquisition of empirical data is both costly and time-consuming.

Finally, we believe that the accessibility of the model‘s design enables users to achieve accurate predictions without requiring advanced mechanical expertise. With the described methods in Bengio et al. (2016) and Montesinos Lopez et al. (2022), the parameter evolution remains traceable, incorporating knowledge of the initial parameters. Hence, a mathematical model is established for each individual node, ensuring a structured representation. The final plausibility of the results can be verified using standardized reference tables, like VDI 2230 (2015). This versatility, coupled with the model‘s potential for time-efficient prediction of output variables, underscores its practical application. Nevertheless, expanding the dataset remains a critical next step to improve prediction consistency, generalizability, and robustness across a broader range of bolted joint configurations.

6. Conclusion

This study presented a hybrid approach integrating empirical data with a supervised feed-forward neural network to predict critical performance metrics of bolted joints, specifically load capacity and head and thread friction coefficients. The proposed methodology effectively addresses key limitations of traditional approaches such as analytical simplifications, computational demands of numerical models, and the scalability challenges of empirical methods. By leveraging empirical data and a computationally efficient neural network architecture, the model captured the nonlinear relationships between input parameters, such as tightening torque and preload force, and output parameters namely remaining load-bearing capacity and friction coefficients with a predictive accuracy of 95.24%. The findings highlight the potential of data-driven modeling as a transformative tool for bolted joint design, particularly in its ability to combine experimental rigor with advanced predictive capabilities. However, limitations such as the small dataset size and the reliance on simplified loading conditions pose challenges to the model‘s generalizability across diverse scenarios. These limitations suggest the need for future research to expand the dataset, incorporate varied load cases (e.g., shear and combined loading), and explore alternative neural network architectures or hybrid machine learning models to enhance robustness and accuracy.

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

This research bridges empirical experimentation and predictive modeling, offering a novel approach to bolted joint analysis. It integrates advanced computational techniques into engineering design, enabling more reliable and efficient optimization of bolted joint performance.

Acknowledgements

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - "Functional modeling of bolted joints under uncertain product conditions for remanufacturing" - with the project number 525034540.

References

Afifi, N., Kaiser, J.-P., Andreas, W., Gisela, L., and Matthiesen, S. (2024). Data-driven functional modeling of corroded bolted joints: A methodological framework for remanufacturing. *In Proceedings of the ASME 2024 International Mechanical Engineering Congress and Exposition (IMECE2024), Portland, Oregon*. https://doi.org/10.1115/IMECE2024-145271.

Apicella, A., Donnarumma, F., Isgrò,F., and Prevete, R. (2021). A survey on modern trainable activation functions. *Neural Networks*, 138:14-32. https://doi.org/10.1016/j.neunet.2021.01.026.

Atta, M., Abd-Elhady, A., Abu-Sinna, A., and Sallam, H. (2019). Prediction of failure stages for double lap joints using finite element analysis and artificial neural networks. *Engineering Failure Analysis*, 97:242-257. https://doi.org/10.1016/j.engfailanal.2019.01.042.

Bengio, Y., Goodfellow, I., and Courville, A. (2016). Deep learning, volume 1. *MIT press Cambridge*, MA, USA.

Bickford, J. (1995). An introduction to the design and behavior of bolted joint marcel dekker. *New York*, 894:41.

Chen, Y., Zhang, J., Liu, Y., Zhao, S., Zhou, S., and Chen, J. (2020). Research on the prediction method of ultimate bearing capacity of pbl based on iaga-bpnn algorithm. *IEEE Access*, 8:179141-179155. https://doi.org/10.1109/ACCESS.2020.3026091.

Chiang, Y. J. and Chiang, A. L. (2023). Fundamentals of Design of Experiments for Automotive Engineering Volume I, volume 1. *SAE International*. https://doi.org/10.4271/9781468606034.

Coelho, J. S., Machado, M. R., Dutkiewicz, M., and O. Teloli, R. (2024). Data-driven machine learning for pattern recognition and detection of loosening torque in bolted joints. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 46(2):75. https://doi.org/10.1007/s40430-023-04628-6.

Fei, Y., Pengdong, G., and Yongquan, L. (2016). Bolt force prediction using simplified finite element model and back propagation neural networks. *In 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, pages 520-523. IEEE. https://doi.org/10.1109/ITNEC.2016.7560415.

Fernández-Ceniceros, J., Sanz-García, A., Antoñanzas-Torres, F., and Martínez-de Pisón-Ascacibar, F. J. (2012). Multilayer-perceptron network ensemble modeling with genetic algorithms for the capacity of bolted lap joint. *In Hybrid Artificial Intelligent Systems: 7th International Conference, HAIS 2012, Salamanca, Spain*, March 28-30th, 2012. Proceedings, Part I 7, pages 545-556. Springer. https://doi.org/10.1007/978-3-642-28942-2_49.

Fukuoka, T. and Takaki, T. (1998). Mechanical behaviors of bolted joint during tightening using torque control. *JSME International Journal Series A Solid Mechanics and Material Engineering*, 41(2):185-191. https://doi.org/10.1299/jsmea.41.185.

Holch, A., Glienke, R., Dörre, M., and Henkel, K. (2023). Evaluation of the load-bearing behaviour of bolts and lockbolt systems under combined tension and shear loading. *In International Conference on Advanced Joining Processes*, pages 3-13. Springer. https://doi.org/10.1007/978-3-031-54732-4_1.

Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*. https://doi.org/10.48550/arXiv.1412.6980.

Kloos, K. and Thomala, W. (2007). Schraubenverbindungen. *Springer*.

Kuhn, M. (2013). Applied predictive modeling. *Springer*. https://doi.org/10.1007/978-1-4614-6849-3.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436-444. https://doi.org/10.1038/nature14539.

Lee, J., Kim, D., and Seok, C.-S. (2022). Methodology for evaluating the tightening torque-clamping force relationship and friction coefficients in bolted joints. *Journal of Mechanical Science and Technology*, 36(4). https://doi.org/10.1007/s12206-022-0328-y.

Li, X., Liu, C., Xue, Y., Xue, S., Liao, S., and Zhou, Y. (2024). Predicting bearing capacity of angle steel bolted connections using machine learning based on experimental and numerical database. *In Structures 67. Elsevier*. 10.1016/j.istruc.2024.107014.

Liang, K., Wang, Z., Yin, Z., and Hao, P. (2024). Investigation on bearing capacity and fracture behaviour of the bolt in a flange connection considering multiple load cases. *In Structures*, volume 62, page 106252. Elsevier. 10.1016/j.istruc.2024.106252.

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press

Montáns, F. J., Chinesta, F., Gómez-Bombarelli, R., and Kutz, J. N. (2019). Data-driven modeling and learning in science and engineering. *Comptes Rendus Mécanique*, 347(11):845-855. 10.1016/j.crme.2019.11.009.
Montesinos López, O. A., Montesinos López, A., and Crossa, J. (2022). Fundamentals of artificial neural networks and deep learning. *In Multivariate statistical machine learning methods for genomic prediction. Springer*. 10.1007/978-3-030-89010-0_10.
Müller, A. C. and Guido, S. (2016). Introduction to machine learning with Python: a guide for data scientists. *O'Reilly Media, Inc*.
Nassar, S., El-Khiamy, H., Barber, G., Zou, Q., and Sun, T. (2005). An experimental study of bearing and thread friction in fasteners. *J. Trib.*, 127(2):263-272. 10.1115/1.1843167.
Nassar, S. A., Ganeshmurthy, S., Ranganathan, R. M., and Barber, G. C. (2007). Effect of tightening speed on the torque-tension and wear pattern in bolted connections. *Journal of Pressure Vessel Technology*, pages 426-440. 10.1115/1.2749290.
Oberg, E. (2013). Machinery's Handbook. New York: Industrial Press Inc.
Olejnik, P. and Ayankoso, S. (2023). Friction modelling and the use of a physics-informed neural network for estimating frictional torque characteristics. *Meccanica*, 58(10):1885-1908. 10.1007/s11012-023-01716-8.
Ren, J. and Sun, X. (2023). Prediction of ultimate bearing capacity of pile foundation based on two optimization algorithm models. *Buildings*, 13(5):1242. 10.3390/buildings13051242.
Roloff, H., Spura, C., Fleischer, B., Wittel, H., and Jannasch, D. (2023). Maschinenelemente: Normung, Berechnung, Gestaltung. *Springer-Verlag*. 10.1007/978-3-658-02327-0.
Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*. 10.48550/arXiv.1609.04747.
Sadouk, L., Gadi, T., and Essoufi, E. H. (2020). Robust loss function for deep learning regression with outliers. *In Embedded Systems and Artificial Intelligence: Proceedings of ESAI 2019, Fez, Morocco*, pages 359-368. Springer. 10.1007/978-981-15-0947-6_34.
Santry, D. J. (2024). Demystifying deep learning: An introduction to the mathematics of neural networks. *Wiley*.
Selvamuthu, D. and Das, D. (2024). Introduction to Probability, Statistical Methods, Design of Experiments and Statistical Quality Control. *Springer*. 10.1007/978-981-99-9363-5.
Steinhilper, W., Sauer, B., and Feldhusen, J. (2012). Konstruktionselemente des Maschinenbaus 1: Grundlagen der Berechnung und Gestaltung von Maschinenelementen. *Springer-Verlag*.
VDI 2230 (2014). Blatt 2 - systematische berechnung hochbeanspruchter schraubenverbindungen - zylindrische einschraubenverbindungen.
VDI 2230 (2015). Blatt 1 - systematische berechnung hochbeanspruchter schraubenverbindungen - zylindrische einschraubenverbindungen.
VDI 2230 (2024). Blatt 3 - systematische berechnung hochbeanspruchter schraubenverbindungen - zylindrische einschraubenverbindungen.
Wettstein, A., Kretschmer, T., and Matthiesen, S. (2020). Investigation of dynamic friction during impact tightening of bolted joints. *Tribology International*, 146:106251. 10.1016/j.triboint.2020.106251.
Wettstein, A. and Matthiesen, S. (2020). Investigation of the thread coefficient of friction when impact tightening bolted joints. *Forschung im Ingenieurwesen-Engineering Research*, 84(1):55-63. 10.1007/s10010-019-00392-z.
Wythoff, B. J. (1993). Backpropagation neural networks: a tutorial. *Chemometrics and Intelligent Laboratory Systems*, 18(2):115-155. 10.1016/0169-7439(93)80052-J.
Yadav, N., Yadav, A., Kumar, M., Yadav, N., Yadav, A., and Kumar, M. (2015). Preliminaries of neural networks. *An introduction to neural network methods for differential equations*, pages 17-42. 10.1007/978-94-017-9816-7.
Yýldýrým, A., Akay, A. A., Gülısýk, H., Çoker, D., Gürses, E., and Kayran, A. (2019). Development of bolted flange design tool based on artificial neural network. *Journal of Pressure Vessel Technology*, 141(5):051203. 10.1115/1.4043915.
Yu, Q., Zhou, H., and Wang, L. (2015). Finite element analysis of relationship between tightening torque and initial load of bolted connections. *Advances in Mechanical Engineering*, 7(5):1687814015588477. 10.1177/1687814015588477.
Zeng, J., Lu, W., and Paavola, J. (2020). Analytical models for predicting the load-bearing strength of beam-tocolumn joints in a composite steel frame. *Thin-Walled Structures*, 157:107100. 10.1016/j.tws.2020.107100.
Zhong, C.-J., Feng, R.-Q., and Zhang, Z.-J. (2021). Prediction of ultimate bearing capacity and structural optimization of aluminum alloy plate joints based on artificial neural network. *International Journal of Steel Structures*, 21(5):1759-1774. 10.1007/978-981-15-1967-3.
Zhou, Z.-H. (2021). Machine learning. *Springer nature*. 10.1007/978-981-15-1967-3

https://doi.org/10.1017/pds.2025.10335 Published online by Cambridge University Press