Received: 31 July 2023 | Revised: 24 October 2023 | Accepted: 14 November 2023

DOI: 10.1111/ffe.14195

SPECIAL ISSUE ARTICLE

FFEMS Fatigue & Fracture of Engineering Materials & Structures WILEY

Analysis of data-driven models for predicting fatigue strength of steel components with uncertainty quantification

Christian Frie[1,2] | Anton Kolyshkin[1] | Chris Eberl[2,3]

[1]CR, Robert Bosch GmbH, Renningen, Germany

[2]Laboratory for Micro- and Materials Mechanics, University of Freiburg, Freiburg im Breisgau, Germany

[3]Fraunhofer Institute for Mechanics of Materials, Freiburg im Breisgau, Germany

Correspondence
Christian Frie, CR, Robert Bosch GmbH, 71272 Renningen, Germany.
Email: cf300@students.uni-freiburg.de

Funding information
None reported.

[Correction added on 24 January 2024, after first online publication: Section 2.3 and Article Category are updated in this version.].

Abstract

Material informatics has emerged as a valuable research field in material science, providing solutions to previously unsolvable problems or accelerating deliverables. Fatigue failure, as a complex and non-deterministic phenomenon, requires a probabilistic approach to assess the uncertainty of the fatigue strength prediction. This study compares various probabilistic data-driven models for credible fatigue strength predictions for three distinct steel groups. The analysis considers data and model uncertainty, evaluating their impacts on predictive quality from engineering and data science perspectives. Results reveal that deep ensembles outperform other probabilistic models regarding negative log-likelihood (NLL), while random forest exhibits the lowest root mean square error (RMSE). Notably, the prediction accuracy of case-hardened steels is negatively affected by insufficient material properties definitions, while stainless steels demonstrate the best performance compared to other steel types.

KEYWORDS

fatigue strength prediction, material informatics, probabilistic data-driven models, uncertainty quantification

Highlights

- DeepEnsembles is the best ML model for probabilistic fatigue strength predictions.
- Random forest is the best ML model for deterministic fatigue strength predictions.

Abbreviations: BNN, Bayesian neural networks; CSG-MCMC, cyclical stochastic gradient Markov chain Monte Carlo; DaBef, Datenbank und Auswertesystem Betriebsfestigkeit (Database for Fatigue Strength); DeepEns, deep ensembles; DL, deep learning; DNN, deep neural network; EvidPrior, evidential neural network; FKM , Forschungskuratorium Maschinenbau (Research Association for the Mechanical Engineering Industry); FKM-Guideline, Analytical Strength Assessment of Components; GPR, Gaussian process regression; HV20, Vickers hardness; MCDropout, MC dropout; ML, machine learning; NIG, normal inverse-gamma distribution; NIMS, National Institute for Materials Science; NLL, negative loglikelihood; RF, random forest; RFJ, random forest with jackknife-estimator; RMSE, root-mean-squared error; TPE, tree-structured Parzen estimator; UQ, uncertainty quantification; VDMA, Verband Deutscher Maschinen- und Anlagenbau (Mechanical Engineering Industry Association).

- The accuracy and credibility of data-driven fatigue strength predictions depends on material properties.
- Understanding and considering the dominant fatigue damage influencing factors are crucial for prediction improvements.

1 | INTRODUCTION

Fatigue describes the weakening of the material due to a cyclic load, resulting in a structural failure of metallic components. Developing structural components requires consideration of various influencing factors for reliable design. Current physical-motivated methods require tremendous modeling and computation effort to predict the fatigue strength for a broad range of metallic materials due to complex multiscale impact of the component's geometry, applied load, surface quality, microstructure characteristics, and so forth.[1–5] Different guidelines have been developed for assessing the component fatigue strength.[6–11] However, these guidelines are often empirically derived and practically focused.

To overcome these circumstances of high modeling effort and increase material fatigue strength prediction accuracy, data-driven approaches have recently become popular to predict the fatigue strength.[12–18] Agrawal et al[12] compared various deterministic machine learning (ML) and deep learning (DL) for steel fatigue prediction using solely chemical composition and heat treatment parameters. He et al[14] compare deep neural network (DNN) and random forest (RF) for fatigue strength prediction with different steels using mechanical properties. Both Xiong et al[18] and He et al[15] apply symbolic regression models to predict the fatigue strength of steels. Symbolic regression models derive an analytical equation contrasting classical black box ML and DL models. However, symbolic regression only applies to a few features due to the tremendous equation complexity for higher feature numbers. All of the works mentioned above show proof of determining fatigue strength with data-driven approaches.

Including uncertainty quantification (UQ) methods within data-driven approaches is beneficial in multiple ways. A common issue of data-driven approaches is the dependency on the database on which the ML models were trained. Data is usually scarce, heterogeneous, and incomplete, especially in engineering applications. This uncertainty is known as data uncertainty. Since fatigue specimen or component tests are cost-intensive, they are rarely open-source available. Combining scarce databases from different sources induces model uncertainty as a consequence of different testing standards, data quality and steel types, due to feature shifts within the data domain. Despite the issue of data quality and availability, the fatigue problem itself is a non-deterministic phenomenon, that is influenced by microstructural changes and processing-induced defects, resulting in probabilistic failure modeling when evaluating fatigue specimen/component tests.[19] Therefore, probabilistic behavior should also be considered within data-driven methods. The above reasons induce either model or data uncertainty in the data-driven fatigue strength prediction. Considering UQ methods gains trust in data-driven approaches since the certainty of the model's prediction can directly be investigated. Furthermore, distinguishing between the types of uncertainty contributes to understanding the database structure and the lack of specific database entries. It offers valuable insights about identifying uncertain steels by classifying the uncertainty. As a result, one learns about which kind of data and features shall be collected in the future to improve the prediction accuracy further.

A first step towards models with UQ was derived by Weichert et al[17] using a Gaussian process regression (GPR) model to quantify the uncertainty in the material fatigue prediction of stainless steels. GPRs cannot distinguish between model and data uncertainty and are limited to a small amount of data points.[20] All the previously mentioned publications focus on predicting material fatigue strength for one specimen geometry, neglecting different component design parameters, as this information was unavailable in their databases. Kolyshkin et al[16] use the first extended database considering material and component design parameters, including UQ, through a probabilistic RF, without focusing on model comparison or differentiating between uncertainty types.

This work compares various probabilistic ML and DL methods for two distinct feature sets to predict the local fatigue strength of steel components with uncertainty quantification (UQ) and distinguish between model and data uncertainty. The best models are identified and tested concerning reliable uncertainty prediction. Subsequently, their performance is tested on three different steel types. The uncertainty owing to the unconsidered fatigue-influencing factors is considered depending on each steel type in the model validation section. Finally, the amount of data required to make reliable predictions with UQ is studied.

This study contributes to the advancement of material informatics and its application to fatigue strength prediction, thus providing reliable material fatigue and component fatigue prediction and increasing component design flexibility and reducing product development time. This research enhances the ability to optimize material selection and design, improving structural components' reliability and durability in various engineering applications. The results offer valuable insights for engineers and data scientists, highlighting the importance of accounting for uncertainties in modeling complex and multiscale phenomena such as fatigue failure.

2 | FUNDAMENTALS

2.1 | Types of uncertainty

In general, the influence of each uncertainty factor on the overall predictive uncertainty is unknown and can not be modeled separately since the features interact with each other. The design and training of DNNs are affected by the data quality and how well the relation between input and output can be modeled. Thus, typically, probabilistic DNNs try to distinguish between the two types of uncertainties: aleatoric and epistemic. Hüllermeier and Waegeman[21] and Gawlikowski et al[22] provide brief overviews of both uncertainties and discusses concepts, sources and methods to identify them.

- Aleatoric uncertainty, also known as statistical or data uncertainty, refers to the uncertainty caused by the data's randomness. Data uncertainty is induced by the noise in the measurement system, the lack of knowledge and the insufficiency of information available about the functional relation between input and output. Aleatoric uncertainty can be reduced by reducing measurement noise, enhancing domain knowledge or providing more meaningful features. The data uncertainty is unaffected by the amount of data used to train the model.[21,22]
- Epistemic uncertainty, also known as the model or systemic uncertainty, is not only provoked by the choice, the structure, and the training of the model itself. It is also affected by the ability of a model to deal with the variability of input data which can cause data domain shift and out-of-distribution data, leading to interpolation or extrapolation. In contrast to aleatoric uncertainty, the epistemic uncertainty can be reduced with more data by lowering the interpolation and extrapolation range. Further reduction of epistemic uncertainty can be done by the right model choice and by enhancing the training procedure.

2.2 | ML models

Recently, a wide variety of publications have dealt with developing and investigating novel and well-known probabilistic DL models for different domains. Wang and Yeung[23] and Jospin et al[24] provide surveys for Bayesian DL, while Gawlikowski et al[22] summarize and discuss publications concerning UQ in DL models. Gawlikowski et al. [22] categorizes DL models based on the UQ approach. Five different probabilistic deep learning models were chosen based on this grouping. The first four columns in Table 1 are summarized from fig. 3 in Gawlikowski et al.[22] Columns 4 and 5 are added to illustrate how the DL approaches distinguish between epistemic and aleatoric uncertainty. All models except *Evidential Neural Networks* use a normal distribution to model the aleatoric uncertainty. They differ in how the models are trained and the epistemic uncertainty is estimated.

2.2.1 | Deep ensembles

Lakshminarayanan et al[25] proposed deep ensemble neural networks to estimate predictive uncertainty. Multiple DNNs with varying weight initialization are trained independently of each other. Due to the stochastic training process, different local minima in the loss landscape are reached. The different network parameterizations lead to varying predictions due to epistemic uncertainty.

2.2.2 | Bayesian neural networks (BNN)

In contrast to standard NNs, where single weights are learned, BNNs place distributions over weights that will be learned during training. Inferring the true posterior distribution of the NN's weights is intractable due to the high number of network parameters. Thus, variational inference methods are used to approximate the true distribution. Usually, a normal distribution is used for approximation from which the mean and the standard deviation are learned in the training process. During prediction, each forward feed samples a weight out of its weight distribution, leading to different predictions and reflecting the epistemic uncertainty.[24,26]

2.2.3 | MC dropout

Gal and Ghahramani[27] presented dropout as a Bayesian approximation approach. Dropout is a widely known regularization method to avoid overfitting. While dropout is usually turned off during prediction, Gal and

TABLE 1 List of probabilistic deep learning methods modeling epistemic and aleatoric uncertainty.

Approach	Model type	Posterior approximation methods	Model name	Epistemic unc.	Aleatoric unc.
DNN	Ensemble Methods		*Deep Ensembles*	Prediction of each ensemble member	Output layer consists of two neurons - one for the mean and one for a standard deviation of Normal distribution
	Bayesian Methods	Variational Inference	*Bayesian Neural Network*	Prediction of multiple forward feeds	
			MC Dropout		
		Sampling Methods	*Stochastic Gradient Markov Chain Monte Carlo*	Prediction of each sampled ensemble member	
	Single Network		*Evidential Neural Network*	Both uncertainties can be inferred from the Normal Inverse-Gamma (NIG) distribution whose parameters are learned during training	

Note: Columns 1 to 4 were summarized from fig. 3 in Gawlikowski et al.[22]

Ghahramani[27] suggest enabling it to make uncertainty estimations. Enabling dropout during prediction and applying multiple forward feeds leads to different predictions through the dropout, causing epistemic uncertainty since Dropout is a binary decision, whether a neuron is turned on or off, the variational distribution is a Bernoulli distribution, while BNNs typically use a normal distribution.

2.2.4 | Cyclical stochastic gradient Markov chain Monte Carlo (CSG-MCMC)

In contrast to the other NN models trained with the most used Adam optimizer,[28] Zhang et al[29] extended the SG-MCMC optimizer presented by Welling and Teh[30] to propose a cyclical (SG-MCMC) optimizer to explore exceptionally high dimensional and multimodal loss-landscapes. This method cyclically anneals the learning rate to reach and escape from local minima. In contrast to the deep ensemble method, where multiple independent networks are saved, only one network is trained, and the parameter configuration at each minimum is saved and added to the ensemble.

2.2.5 | Evidential deep learning (EvidPrior)

Amini et al[31] proposed a deep evidential neural network for regression tasks. This NN model uses a single network and requires only one forward feed for the prediction and uncertainty estimation. This makes it popular for time-critical and memory-restricted applications. Explaining the network's method is more complicated than the other probabilistic approaches. For more details, we refer to Amini et al.[31]

2.2.6 | Random forest with jackknife-estimator (RFJ)

The RFJ is the only ML approach, while all other models are DL methods. Wager et al[32] proposed the jackknife estimator for predictive uncertainty estimation for the random forest. In contrast to the DL methods, the prediction is a two-step process. First, the RFJ predicts the mean, followed by a confidence estimation around the mean prediction. Therefore, The RFJ cannot distinguish between aleatoric and epistemic uncertainty.

2.3 | Metrics and losses

The goal of supervised learning, such as regression, is to map the input vector $\mathbf{x}_i$ to a corresponding real-valued output vector $\mathbf{y}_i \in \mathbb{R}$ of a dataset $\mathcal{D} = \{\mathrm{x}_i, \mathrm{y}_i\}_{i=1}^{M_{Data}}$. In classical regression tasks, a machine learning model f with parameters θ outputs a single value $\mu_i = \mathrm{f}_\theta(x_i)$ for a given input x_i. The training consists of finding the model parameters θ, which minimize a loss function.

$$\mathrm{L}_{RMSE}(\theta) = \sqrt{\frac{1}{M_{Data}} \sum_{i=1}^{M_{Data}} (\mathrm{y}_i - \mu(\mathrm{x}_i))^2} \tag{1}$$

The root-mean-squared error is typically used as a loss function that does not consider the predictive uncertainty (Equation 1). Adding a second output in the final layer of a neural network can capture heteroscedastic Gaussian variance. Using a *Softmax* activation function ensures that $\sigma^2(\mathrm{x}_i) > 0$.[33] The loss function changes to the negative log-likelihood criterion in Equation (2) when the NN outputs a mean $\mu(\mathrm{x}_i)$ and a standard deviation $\sigma(\mathrm{x}_i)$ of a Gaussian distribution.

$$\mathrm{L}_{NLL}(\theta) = \frac{1}{M_{Data}} \sum_{i=1}^{M_{Data}} \frac{log(\sigma^2(\mathrm{x}_i))}{2} + \frac{(\mathrm{y}_i - \mu(\mathrm{x}_i))^2}{2\sigma^2(\mathrm{x}_i)} + const. \tag{2}$$

$$\mu_*(\mathbf{x}) = \frac{1}{M_{Pred}} \sum_{j=1}^{M_{Pred}} \mu_j(\mathbf{x}_i) \tag{3}$$

$$\sigma_*^2(\mathbf{x}_i) = \frac{1}{M_{Pred}} \sum_{j=1}^{M_{Pred}} \underbrace{(\sigma_j^2(\mathbf{x}_i)}_{\text{Aleatoric unc.}} + \underbrace{\mu_j^2(\mathbf{x}_i)) - \mu_*^2(\mathbf{x}_i)}_{\text{Epistemic unc.}} \tag{4}$$

During prediction mode, Lakshminarayanan et al[25] and Kendall and Gal[34] model the predictive uncertainty by a uniformly weighted Gaussian mixture $\mathcal{N}(\mu_j(\mathbf{x}_i), \sigma_j(\mathbf{x}_i))$ where $\{\mu_j(\mathbf{x}_i), \sigma_j^2(\mathbf{x}_i)\}_{j=1}^{M_{Pred}}$ are M_{Pred} predictions of the probabilistic neural network, where the meaning of M depends on the model. For the *Deep Ensembles*, M_{Pred} stands for each model in the ensemble where *MCDropout* makes M_{Pred} forward feeds with the same model having Dropout enabled. These predictions are assembled into a single mean and a single standard deviation $\mathcal{N}(\mu_*(\mathbf{x}_i), \sigma_*^2 \mathbf{x}_i))$ with Equations (3) and (4). The total variance in Equation (4) can be decomposed into the aleatoric and epistemic uncertainty.[25,34]

2.4 | Bayesian optimization

Bayesian optimization is a sequential strategy for global optimization of black-box function and is often used in deep learning models for hyperparameter tuning.[35–38] Bergstra et al[39] proposed the tree-structured Parzen estimator (TPE), which constructs a surrogate model to obtain the best hyperparameters for the ML model. Out of a predefined search space, the TPE sequentially optimizes the surrogate model for proposing the hyperparameters for the next iteration cycle, the chance of the largest expected improvement of a predefined metric (e.g., mean-squared error)

2.5 | Calibration

Calibration describes the degree of the uncertainty's reliability.[22] A well-calibrated estimator predicts the target value with the corresponding predictive probability. Guo et al[40] observed that many DL models are uncalibrated, leading to underconfident or overconfident predictions. Different calibration methods have been proposed in literature.[40–43] Kuleshov et al[41] proposed a post-processing calibration method based on the isotonic regression applied to a hold-out dataset after training. The miscalibration area criterion determines a scalar value for measuring the goodness of the predictive uncertainty on the hold-out dataset. Based on the miscalibration area, a recalibration factor can be estimated and multiplied by the predictive uncertainty to receive calibrated predictions. The open-source uncertainty toolbox estimates the miscalibration area and the recalibration factor.[44]

2.6 | Fatigue

Fatigue calculations are based on comparing actual loads and load collectives with empirical material data, which are gained by the number of cycles the material can withstand for a given amplitude. Therefore, the fatigue strength is determined by the highest stress the material can resist without breaking under a cyclic load. Typically, various load amplitudes for a given stress ratio are considered to characterize this load. Normed specimen or component geometries are tested under various stress amplitudes until failure, or if the component exceeds a certain number of cycles, it is considered as runout, as the component would not fail within its lifetime in an industrial application. The S-N curve describes the fatigue characteristics of a material by fitting an S-N curve model to the experimental test results and is required for reliable component design. Different S-N curve models have been proposed in the literature.[19,45–49] Several factors, such as the material, stress ratio, component design, load, environmental conditions, and so forth, influence the S-N curve. Analytical empirical models have been derived to reduce cost and time for the design effort, such as the FKM-Guideline.[11] Other guidelines were developed for a specific application, such as designing of offshore wind parks[8] or for vessels design in the petroleum industry[9] and other guidelines were developed for national[6,10] for continental/national interests.[7] The FKM-Guideline is used in this work.

2.6.1 | FKM-Guideline

The FKM-Guideline[11] "Analytical Strength Assessment of Components" is a guideline developed by the Forschungskuratorium Maschinenbau (FKM) and published by the VDMA (Verband Deutscher Maschinen- und Anlagenbau). It describes a general procedure for assessing the strength of components in mechanical engineering made of steel, cast iron and aluminum materials. It comprises a static and fatigue strength assessment for linear material behavior. Additionally, a subdivision between nominal and local stresses is considered to account for notched components. The FKM-Guideline is derived empirically and the factors depend on the materials, loadings, component designs, stress ratios and environmental conditions such as temperature and distinguish between *stainless*, *case-hardened*, and all other steels, solely named *steel* for steel materials. The FKM-Guideline is valid for tensile strength < 1400 MPa only and should carefully be applied to tensile strength between 1400 MPa and 1600 MPa as it assumes surface-induced fatigue damage.[11] It is designed to derive a conservative fatigue strength assessment and showed an RMSE of 90 MPa in Kolyshkin et al[16] applied on the same database used in this work.

2.6.2 | Damage mechanism

The material's fatigue properties depend on the component design and the microstructural characteristics. The latter is responsible for the probabilistic fatigue behavior due to variations within the microstructure. Moreover, variation in process parameters, different testing conditions influences the fatigue strength. The damage mechanisms strongly depend on factors that induce stress concentration. Stress concentration can be caused by notches in the designed component or microstructural characteristics. Fatigue cracks for low-strength materials usually nucleate at intrinsic defects, such as slip bands and grain boundaries. Process defects include scratches, nonmetallic inclusions, segregation, and so forth and are mostly responsible for fatigue failure for high-strength steels. There is also a mixture zone of intrinsic and process defects in the transition of low- to high-strength steels.[50–54]

3 | METHODS AND CONCEPTS

3.1 | Database and feature selection

The database used to train the data-driven approaches includes Bosch's internal and external open access datasets. The three main database sources are the following:

- National Institute for Materials Science (NIMS)[55]
- Datenbank und Auswertesystem Betriebsfestigkeit (DaBef)[56]
- Robert Bosch internal database

The collected database includes 1250 S-N Curve experiments evaluated on approximately 30,000 specimens of 58 steels tested under different conditions. The tabular database includes up to 70 features per S-N curve experiment, containing the chemical composition, processing, forming, heat treatment, finishing, static tests, load, and geometry during testing. The raw data of the experimental outcomes are evaluated with the maximum-likelihood method for a bilinear S-N curve model according to Köder.[47] As a result, four S-N curve parameters, with the slope k in high cycle fatigue area, the kneepoint N_k, the data scatter T_s, and the fatigue strength of 50% failure probability at $2*10^7, Sd_{50\%,2E7}$, are estimated for each database entry. The raw data assumes specimens that did not fail within $2*10^7$ cycles as survivals. Thus, the very-high-cycle fatigue regime is not considered.

In this work, two selected feature combinations are used to predict the local fatigue strength $Sd_{50\%,2E7}$ (Table 3) for different steel design components. The features were chosen based on the availability in the database, selected features in other publications,[12,16,57,58] and the generally known influencing factors on the component fatigue strength.[59] The *Engineer features* comprise the chemical composition, mechanical properties, specimen geometry, loading, and testing frequency features.

The FKM-Guideline[11] applies an analytical prediction of the component fatigue strength based on the corresponding material group shown in Table 2, which is used as an additional feature to incorporate an engineering

TABLE 2 Dataset with in total 853 entries, categorized by the FKM-Guideline material group[11] and the corresponding number of entries.

FKM material group	Steel	Stainless steel	Case-hardened steel
Dataset size, $M_{Data} = 853$	707	82	64

model into the pure data-driven approaches. The local fatigue strength concept is applied in this work to account for the induced local stresses of notched specimens. For *Stainless* and *Steel* steels, the analytical FKM-prediction was calculated according to chapter 4 of the FKM-Guideline for the fatigue strength for local stresses. The calculation follows: First, the material fatigue strength for a stress ratio $R = -1$ is calculated based on the tensile strength with different constants and multiplication factors according to the loading type and material. In the next step, the material strength is adjusted by the design factor calculated based on the component's design parameters, such as loaded volume, stress concentration factor and surface roughness. As the surface roughness is not included in the database, the FKM-Guideline uses a default value. Lastly, the fatigue strength of the component for a specific stress ratio is derived under consideration of a material-dependent mean stress factor multiplied by the component fatigue strength at $R = -1$. In contrast, FKM Chapter 5.5 was applied for surface-treated components and local stresses for the *case-hardened* steels. The calculation differs from Chapter 4, as a surface-treated factor is additionally included.[11]

Figure 1 illustrates the distribution of hardness measured at the surface and tensile strength of the core material concerning its steel group for *Stainless* and *Steels*. For *case-hardened* steels, the surface hardness and the tensile strength of the case-hardened material were used. The database captures a wide range of low- to high-strength steels. The *Process features* extend the *Engineer features* by heat treatment parameters.

Missing values were removed from the database, resulting in a remaining dataset size of 853 entries. The heat treatment features were removed from the dataset for the *Engineer Features* and included for the *Process Features* to train both feature combinations on the same dataset size. All models were implemented using Python with the DL library tensorflow and the probabilistic extension, tensorflow-probability.[60] The library Scikit-Learn[61] was used for the RF model, and the uncertainty toolbox[44] was applied for model calibration. Hyperopt's[62] TPE was used for hyperparameter optimization.

3.2 | General method

Figure 2 illustrates the two branched diagrams of this work. First, the performance of the probabilistic ML and DL models for both feature combinations for all steels are compared in the left branch to determine the best models. Each model's parameters have been optimized in an iterative process to determine the best hyperparameters, followed by a 10-fold cross-validation to examine the model's performance on the entire dataset. The best two models are selected for further analysis in the right branch based on the NLL and RMSE metrics in Section 4.1. Both models are tested for reliable uncertainty prediction by conducting a calibration analysis in the right branch in Section 4.2. Subsequently, Section 4.3 investigates the model's performance and predictive uncertainty on the testing dataset based on the different FKM steel types (*Steel*, *Stainless*, and *case-hardened*) for model validation. Furthermore, the model's performance evolution and the predictive uncertainty on the testing dataset for solely *Steel* depending on the number of database entries are investigated in Section 4.4.

3.3 | Hyperparameter tuning and ML training

Selecting the appropriate hyperparameters for ML models is a crucial task. The TPE from Section 2 is used as a Bayesian optimization algorithm to select the optimal hyperparameter for each model. The search spaces from which the hyperparameters were determined can be found in Appendix A (Tables A1–A6). For each model, the TPE performed 150 optimization cycles, and the best hyperparameter set was fitted in a 10-fold cross-

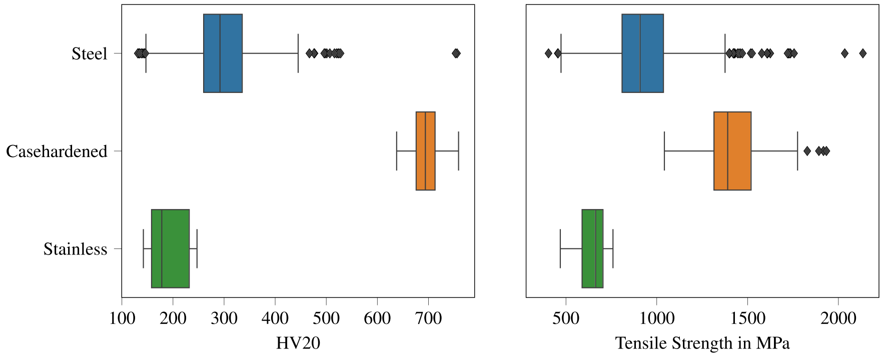

FIGURE 1 Distribution for Vickers hardness (HV20) at the surface and tensile strength of the database for three different steel types according to the FKM-Guideline.[11] [Colour figure can be viewed at wileyonlinelibrary.com]

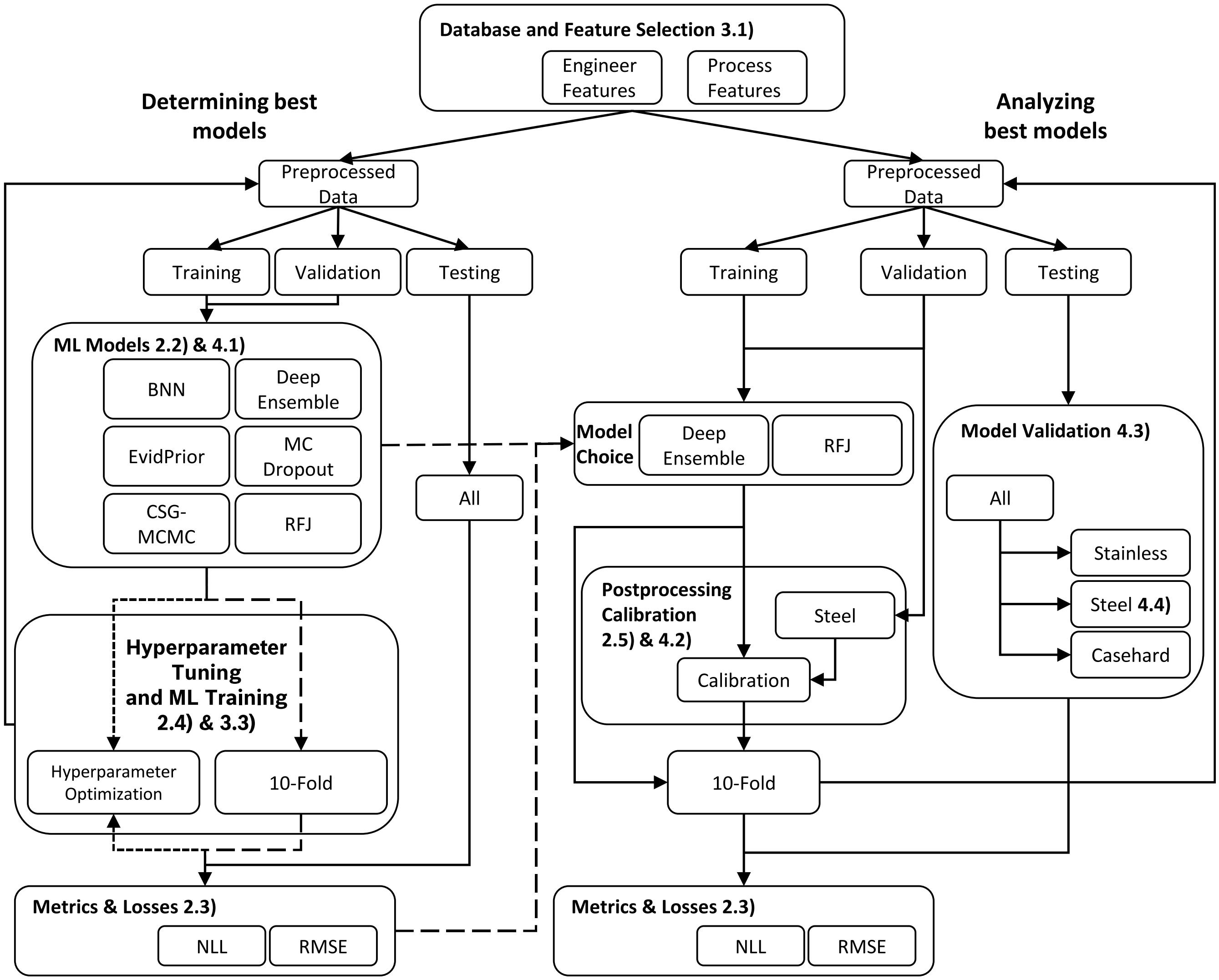

FIGURE 2 Worflow of this work, containing two main branches for determining and analyzing the best models. The numbers symbolize the corresponding chapters. The dotted lines in the left branch illustrate the iterative process of the hyperparameter optimization, while the dashed lines demonstrate the 10-fold cross-validation. [Colour figure can be viewed at wileyonlinelibrary.com]

validation where a train, validation, and test split of 70-15-15 was used. The TPE determines the follow-up hyperparameters based on the predictive performance of the test data set, seeking to reduce the error further and increase accuracy. The hyperparameter for each optimization iteration was visually investigated to ensure that the hyperparameter boundaries in the search space did not affect the TPE algorithm by excluding the optimal value. All DL models were trained with early stopping on the validation dataset to prevent overfitting of the training dataset. The validation dataset was also used as a calibration dataset in the right branch to determine the calibration factor. All models except for *EvidPrior* and *RFJ* use a normal distribution to model the aleatoric uncertainty and are trained against the negative log-likelihood criterion in Equation (2). The mean prediction for the testing dataset and the predictive uncertainty is modeled with Equations (3) and (4). The *RFJ* is trained against the root-mean-squared error in Equation (1), and the predictive uncertainty is calculated with the jackknife-estimator as described in Wager et al.[32] For training and prediction of *EvidPrior*, we refer to their publication.[31]

4 | RESULTS AND DISCUSSION

4.1 | Determining best models

Figure 3 illustrates the performance for six different probabilistic data-driven models for the two feature combinations evaluated for the negative-loglikelihood (NLL) and RMSE metric according to Equations (1) and (2) for all steels types.

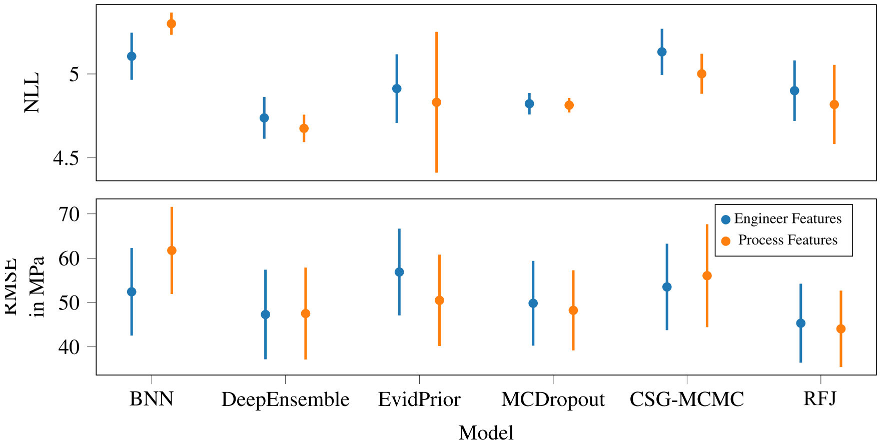

FIGURE 3 Comparison of different probabilistic ML and DL models for two feature combinations for the negative-loglikelihood (NLL) and RMSE metric according to Equations (1) and (2) for the entire database with mean and standard deviation of the 10-fold cross validation. [Colour figure can be viewed at wileyonlinelibrary.com]

The chemical composition combined with the heat treatment determines the microstructure and defect morphology of the material, influencing the mechanical and fatigue properties. Hence, it is expected that the predictive quality will increase. The performance of all models, except the BNN, increases with lower NLL values by using the *Process features* where heat treatment features are included. Including the heat treatment features increases the number of model inputs. The BNN has to learn twice as many parameters as the other DL approaches with the same amount of data available. Increasing the model's input dimension without increasing the available data explains the decrease in BNN performance.

The *DeepEnsemble* outperforms all other models in terms of NLL, except for the RMSE where *RFJ* has the lowest value. Additionally, the RMSE for the *DeepEnsemble* does not decrease as much as the NLL. The fact that NLL and RMSE are not necessarily correlated can also be seen for the *CSG-MCMC* model, where the RMSE increases while the NLL decreases. Lakshminarayanan et al[25] also observed that NLL and RMSE are not always correlated and reasoned that the model is trained with the NLL loss function instead of the RMSE, which incorporates both the mean and standard deviation (see Equation 2). Reducing the NLL does not necessarily lower the RMSE. The *RFJ* is trained against the RMSE and provides the lowest RMSE value for both feature combinations. Applying the Jackknife-Estimator after the training procedure for uncertainty estimation results in a mean NLL comparable to the *EvidPrior* and *MCDropout* but worse than the *DeepEnsembles*.

While Bayesian methods have a strong theoretical base,[23,24] they are less frequently used in industrial applications compared to *DeepEnsembles*. Lakshminarayanan et al[25] compared the performance of *DeepEnsembles*, *BNN*, and *MCDropout* for well-known standard UCI-datasets[63] showing *DeepEnsembles* is superior over the other methods. *DeepEnsembles* can capture multiple modes in the loss landscapes, while Bayesian methods can only observe single optima.[22,23] This characteristic makes *DeepEnsembles* methods robust and reliable.

Gustafsson et al[64] investigated uncertainty quantification methods for real-life applications and found that *DeepEnsembles* are more applicable than *MCDropout*, which was also founded by Beluch et al[65] for active learning tasks. We also compared the performance of *DeepEnsembles* trained against the RMSE instead of NLL (not shown here) and witnessed an improvement of the RMSE value while still being less accurate than the *RFJ*. This survey was also observed by Borisov et al,[66] who investigated the performance of ML and DL methods for supervised learning of heterogeneous tabular data and found that ML methods outperform DL approaches in terms of RMSE.

This work extends the funding from Borisov et al[66] for probabilistic methods, that ML models achieve comparable results for the NLL than DL approaches for heterogeneous tabular data. Since *DeepEnsembles* were found to be robust and reliable, we claim this also holds for an ensemble of ML methods, like the *RFJ*. However, this requires more research on distinct datasets for generalizability. The performance differences depend on the metric of interest since *DeepEnsembles* is trained on the NLL while the *RFJ* uses the RMSE. Choosing the optimal model depends on the applications. If solely the mean prediction is of interest and uncertainty estimation is irrelevant, the *RFJ* is the best model as it provides the smallest root-mean-squared error. If uncertainty quantification is required, the *DeepEnsembles* yields the smallest NLL with the cost of less accurate mean prediction. Both models are considered for further analysis.

4.2 | Calibration of *DeepEnsembles* and *RFJ*

Figure 4 illustrates the change in NLL performance before and after calibration. The post-processing calibration procedure is applied after the training procedure based on isotonic regression, which determines a scalar value multiplied by the standard deviation to achieve a more calibrated prediction. The calibration for the *RFJ* for both feature combinations significantly improves the NLL, while the calibration for *DeepEnsembles* harms the prediction quality.

There is rare research on uncertainty calibration for regression models caused by the increased complexity for continuous output, compared to discrete and bounded outputs as in classification tasks.[41,67] Chung et al[44] tested different evaluation metrics besides the NLL for regression tasks and concluded that calibrated UQ for regression tasks might not always be straightforward. Even for classification tasks in DL, UQ calibration is still an open field of research.[68] Rahaman and Thiery[68] investigated the uncertainty calibration for *DeepEnsembles* for classification tasks and found a non-trivial trade-off between data augmentation which increases robustness and increased calibration accuracy and the resulting data distribution shift leading to a raise of the calibration error. Thus, it remains unclear why the performance of the *RFJ* increases while the predictive quality of *DeepEnsembles* decreases trained on the same data split.

The main drawback of the calibration procedure is that a single scaling factor is determined for the entire dataset. From a data science point of view, the calibration dataset requires a representative subset to determine a single factor for recalibration. This is usually achieved by collecting a random subset. Data is usually scarce in engineering applications and taking a random subset further reduces the training dataset. The overall performance relies on a trade-off between the amount of training data and a representative calibration dataset.

A single calibration factor is calculated for *Steels* since it provides enough data for providing a representative calibration dataset while *Stainless Steels* and *case-hardened Steels* are rarely present in the dataset in Table 2. Thus, their predictive uncertainty of *Stainless Steels* and *case-hardened Steels* are not calibrated. Therefore, the *RFJ* in Figure 4 is partially calibrated as it is a weighted sum of all steel groups. Calculating a single calibration factor for all steel types was tested but neglected due to worsening the NLL performance of *Stainless Steels* and *case-hardened Steels*.

Calibrating all steel types with one calibration is also questionable from an engineering perspective since it would imply that the predictive uncertainty of the fatigue strength of all steels has a homogeneous error, which is adjusted by one calibration factor.

The fatigue properties of *Stainless Steels* correlate with the quasistatic parameters, and slip bands emergence at the surface are the primary damage mechanism. Contrary, the fatigue process of *case-hardened Steels* is not only affected by slip bands emergence but also by the size of inclusions and the steel's purity, resulting in more heterogeneous and complex failure mechanisms.[50–54]

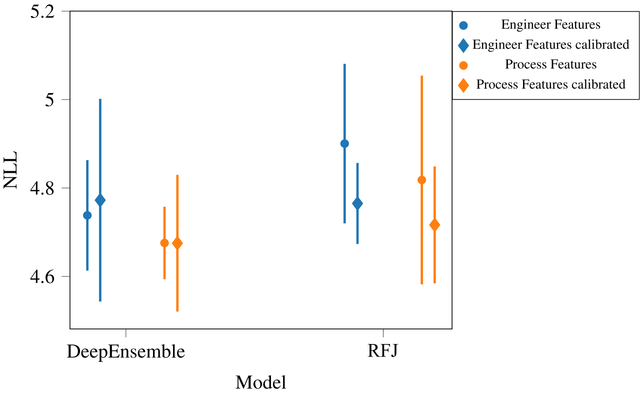

FIGURE 4 Change in NLL performance for the *DeepEnsemble* and random forest (*RFJ*) for the uncalibrated and calibrated model for both feature combinations with a mean and standard deviation of the 10-fold. [Colour figure can be viewed at wileyonlinelibrary.com]

4.3 | Model validation

Figure 5 illustrates the performance of the *DeepEnsembles* and *RFJ* depending on the steel type and the corresponding predictive total uncertainty. The *DeepEnsembles* can further distinguish between aleatoric and epistemic uncertainty according to Equation (4). In contrast, the total uncertainty of the *RFJ* is not capable of further differentiation.

The NLL, RMSE, and total uncertainty of the *DeepEnsembles* and *RFJ* show a similar trend for all steel types and feature combinations where more minor RMSE errors lead to lower total predictive uncertainty because less uncertainty is required to capture the fatigue strength value and thus resulting in lower NLL values. From a physical perspective, it can be assumed that adding heat treatment parameters increases the performance because it determines the microstructure. This assumption holds for both models and all steel types except for the predictive quality from the *DeepEnsembles* for *case-hardened* steels. A reason could be the increased input size for the DL approach while providing the same amount of *case-hardened* steel entries.

Stainless steels provides the lowest NLL and RMSE error, followed by *Steel* and *case-hardened* steels, while the latter has a significantly worse performance. *All* is a weighted average of all steel types. This is mainly influenced by *Steel* but with higher NLL values due to the impact of *case-hardened* steels. The *RFJ* has a slightly better performance for *case-hardened* steels than the

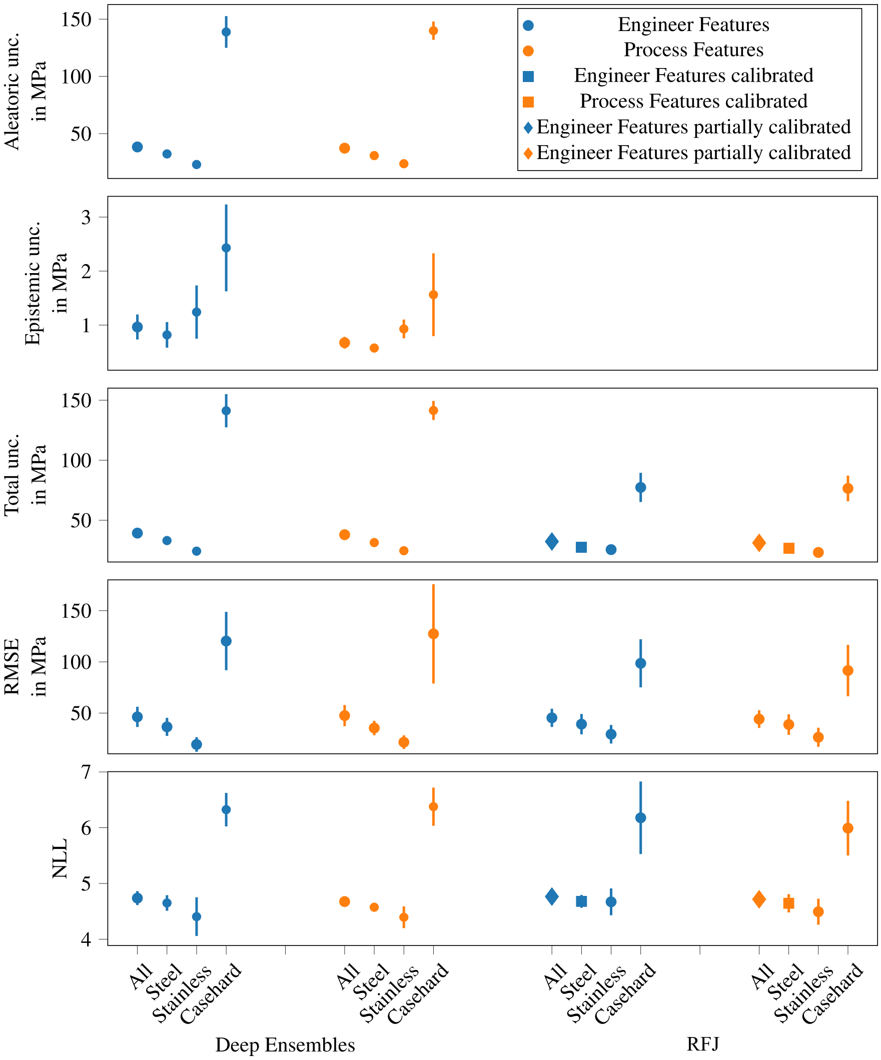

FIGURE 5 Comparison of model performance and different predictive uncertainties (Equation 4) for different steel types according to Wächter et al[11] with mean and standard deviation of the 10-fold. [Colour figure can be viewed at wileyonlinelibrary.com]

DeepEnsembles, indicating that the DL approach is not just incapable of capturing deeper relations for improving the prediction quality. Furthermore, it performs even worse than the ML approach of a *RFJ*. In contrast, the *DeepEnsembles* outperforms the *RFJ* about the *Stainless* steels for the NLL and RMSE, while the performance for *Steels* is similar.

The total predictive uncertainty of the *DeepEnsembles* can be split into an epistemic and an aleatoric part according to Equation (4). The epistemic uncertainty for *Stainless* and *case-hardened* steels is increased compared to *Steel*, indicating that more data will further reduce this uncertainty. *Steel* is the group that is most present in the dataset compared to *Stainless* and *case-hardened* steels (see Table 2). However, the epistemic uncertainty contributes just a small fraction to the total uncertainty while the aleatoric uncertainty provides the highest part. The aleatoric uncertainty can only be reduced by providing more meaningful features to the model. In general, adding known physical influence factors like residual stresses, surface roughness, grain size, defect size, or defect distribution could enhance the predictive quality of both models.

Case-hardened steels have the worst NLL/RMSE performance and highest aleatoric/data uncertainty, indicating that the *DeepEnsemble* model is not capable of learning the fatigue strength with the provided input since information of the main fatigue damage mechanism of high-strength steels are not provided in the dataset. High-strength steel's predominant fatigue damage mechanisms are process defects, where sizes, shapes, and distribution are valuable but missing information.[50–54] Presumably, providing information about inclusions and volume defects could probably increase the predictive performance, especially for high-strength steels. The hardening process significantly changes the surface hardness, the hardening depth, the microstructure, surface roughness, the residual stresses, etc. All these influencing factors, except the surface roughness, are indirectly considered through the carburization and hardness at the surface (see Table 3). The predictive quality for *case-hardened* steels is also limited by the hybrid modeling approach of incorporating an analytical prediction of fatigue strength by the FKM-Guideline. Figure 1 shows the median of *case-hardened* steels for tensile strength around 1400 MPa. The model's input feature might already be uncertain since the FKM-Guideline is not valid above 1600 MPa, resulting in a wrong prediction and increasing the RMSE and aleatoric uncertainty. However, all influencing parameters should be collected separately in the future instead of indirect consideration through the carburization and hardness at the surface. This could increase the data diversity and probably improve performance as the ML methods can learn to distinguish between the separate influencing factors. Furthermore, the data coverage in the feature space and, therefore, the scope and limitations of the ML model are more obvious for users since the value ranges of each

TABLE 3 Feature description for selected feature combinations *Engineer Features* and *Process Features*.

Category	Description	Engineer features	Process features
Chemical composition	*wt*% of carbon	✓	✓
	wt% of silicon	✓	✓
	wt% of manganese	✓	✓
	wt% of phosphorus	✓	✓
	wt% of sulfur	✓	✓
	wt% of chromium	✓	✓
	wt% of nickel	✓	✓
	wt% of molybdenum	✓	✓
Mechanical properties	0.2% proof stress in MPa	✓	✓
	Tensile strength in MPa	✓	✓
	Elongation at fracture in %	✓	✓
	Vicker Hardness HV20 on surface	✓	✓
Component design, load and testing	Effective diameter in mm	✓	✓
	Relative stress gradient at hot spot mm^{-1}	✓	✓
	90% of maximum loaded volume according to Wächter[11] in mm^3	✓	✓
	Stress concentration factor	✓	✓
	Loading type	✓	✓
	Stress ratio	✓	✓
	Testing frequency in Hz	✓	✓
Material science model	Analytic prediction of local fatigue strength based on FKM Guideline 2012[11]	✓	✓
Material group	Categorization of steels based on FKM Guideline 2012[11]	✓	✓
Process parameters	Normalizing temperature in $^\circ$ C		✓
	Carburization temperature in $^\circ$ C		✓
	Carburization time in min		✓
	Through hardening temperature in $^\circ$ C		✓
	Through hardening time in min		✓
	Cooling rate for through hardening $^\circ$ C/min		✓
	Temperature of the cooling medium		✓
	Tempering temperature in $^\circ$ C		✓
	Tempering time in min		✓
	Cooling rate for tempering $^\circ$ C/min		✓

feature are known. This increases credibility and enhances usage in applications.

For low-strength ductile materials, such as *Stainless* steels (see Figure 1), several works have shown that the fatigue strength linear correlates with the quasi-static mechanical properties.[50,52,53,69] The dominant fatigue damage mechanism are slip bands or grain boundaries at the surface, while inclusion size usually does not determine fatigue strength.[53,54,70] Thus *RFJ*, as well as the *DeepEnsembles*, predict the fatigue strength with the highest accuracy compared to the other steel types since one failure mechanism is predominant and the quasi-static mechanical properties already imply a high correlation with the fatigue strength. The surface roughness is vital for the induced failure initiation in bulk materials. Therefore, providing this information could further reduce the error for low-strength steels.

The *Steel* group is a pool of all other steel types that can not be grouped into one of the other two categories. Thus providing a heterogeneous mixture of a variety of steels. The hardness in Figure 1 shows the widest distribution and thus includes fatigue damage mechanisms from intrinsic and processing defects. The overall performance is slightly worse than for *Stainless* steels. Further analysis cannot be done here since all steel types are summarized together. Providing defect information for high-strength steels and surface roughness for low-strength steels could increase the predictive quality similar to the conclusion from *case-hardened* and *Stainless* steels.

Thus, further work should consider alternative categorizations for model validation instead of the three steel groups of the FKM-Guideline. A categorization according to the predominant fatigue damage mechanism could be applied by differentiating between intrinsic defects, process defects and a mixture of both. This distinction could further help to understand what data-driven models can learn and what causes the aleatoric/data uncertainty in the prediction. Notably, the database includes various load and specimen geometries, which increases the complexity of understanding the underlying fatigue damage mechanisms.

4.4 | Influence of dataset size on models performance

Figure 6 illustrates the performance change for *DeepEnsemble* and *RFJ* depending on the dataset size. This investigation was only tested for the *Steel* type since *Stainless* and *case-hardened* steels are rarely present in the dataset (see Table 2). Further reduction of the training data reduces the model's generalizability due to low data density for both steel types, and the applicability in practice is questionable. 100% matches the original dataset with 853 entries. The dataset size was reduced by randomly selecting a subset from the *Steel* material group. The NLL of the *DeepEnsemble* and the RMSE for the *RFJ* decrease with a rising number of data for both feature combinations, seemingly converging beyond 100% of available data. Thus, providing more data for both feature combinations and models will probably lead to further error reduction, but the benefit appears marginal. The process features increase the predictive quality for both models and both metrics.

The NLL and RMSE performance correlates with the database size for the *RFJ* as the prediction is a two-step procedure. First, the prediction mean is calculated, followed by the uncertainty interval. Thus, reducing the prediction accuracy directly impacts the uncertainty prediction, resulting in a lower NLL performance. At the same time, the *DeepEnsembles* shows a different behavior where the RMSE remains constant, and the NLL decreases with lowering the available training data. A decreasing NLL by lowering the dataset size is expected for the *DeepEnsemble* since it is trained on the NLL and not on the *RMSE*. Thus, decreasing the training data increases the total predictive uncertainty, resulting in less certain predictions. At the same time, the RMSE error stays unaffected, indicating that it is easier to train the mean than the total uncertainty with the NLL loss function in Equation (2). It requires more investigations on why the RMSE for *DeepEnsemble* does not increase. The total uncertainty for *DeepEnsemble* is mainly driven by the data uncertainty, similar to Figure 5, where the aleatoric uncertainty slightly increases when lessening data. At first, this seems to conflict with the definition of aleatoric uncertainty given in section 2, that it is unaffected by dataset size. However, the aleatoric uncertainty will also increase if the DL method cannot capture the appropriate underlying function due to the lack of training data. The definition of the aleatoric uncertainty assumes that there is enough data to train a DL model. The epistemic uncertainty also increases when decreasing data for both feature combinations, where the rise is higher when incorporating heat treatment features due to the increased input size for the same amount of data.

The amount of data required to train a model always depends on the relational complexity between input and output. The random subset taken to decrease the training data includes a broad range of low- to high-strength steels (see Figure 1). Thus, not only intrinsic defects but also process defects play an essential role in fatigue failure mechanisms. Excluding high-strength steels could reduce complexity due to focusing on fewer failure mechanisms and less data is required for low errors similar to the predictive quality of *Stainless* steels. Weichert et al[17] used only 114 *Stainless* steels to achieve reliable fatigue strength

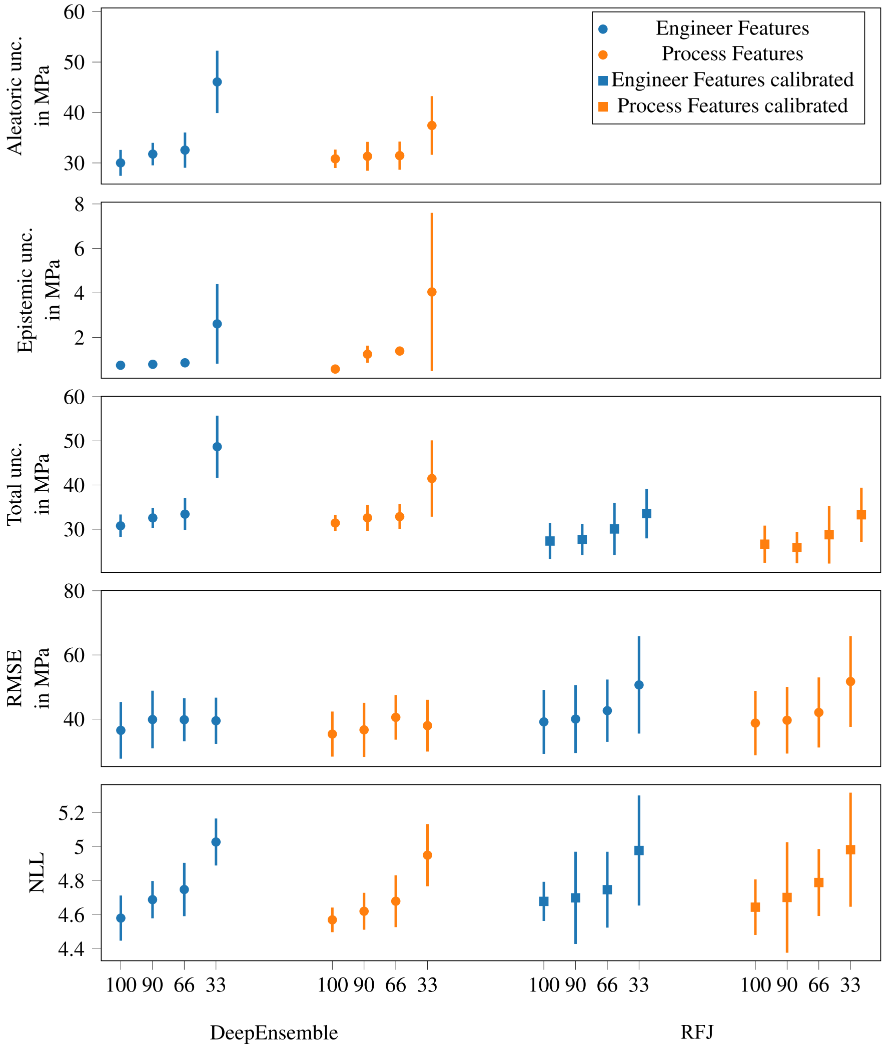

FIGURE 6 Change of different types of predictive uncertainty according to Equation (4) for DeepEnsemble and RFJ depending on the dataset size in % for two feature combinations with mean and standard deviation of the 10-fold. [Colour figure can be viewed at wileyonlinelibrary.com]

prediction using GPR. On the contrary, increasing the amount of data for high-strength steels will decrease the epistemic uncertainty. Still, the RMSE error and the aleatoric uncertainty remain high unless more meaningful features are included. The material digitalization will further increase, making more data available for data-driven methods. Including more design elements will further increase the database diversity and improve the generalization of the ML application to assess the fatigue strength for various design elements. The epistemic uncertainty might support detecting design components that have not been available in the database yet, indicating the reliability of the ML/DL prediction. Consequently, ML/DL models with UQ can increase the trustworthiness of data-driven approaches in engineering applications.

5 | CONCLUSION

Probabilistic ML/DL methods are powerful methods to model complex input-output relations by distinguishing between data and model uncertainty. This work compares various probabilistic Machine and Deep Learning methods to predict the fatigue strength with uncertainty quantification and validate the results for three different steel types categorized by the FKM-Guideline into *Stainless* steels *case-hardened* steels and other *Steels*. Despite its simplicity for probabilistic metallic fatigue strength prediction, we show that *DeepEnsemble* outperforms all other probabilistic DL methods. Even though the Random Forest *RFJ* performs slightly worse than the *DeepEnsemble*, it is easier and faster to train than the *DeepEnsemble* model and thus exhibits a good benchmark model. Calibrating probabilistic models for regression tasks is crucial and needs more research to provide reliable predictions as it remains unclear when calibration is required. Including heat treatment features improves the predictive quality for all three steel groups. We showed that the predictive quality is the best for low-strength ductile *stainless* steels since there is solely one dominant fatigue damage mechanism. *Case-hardened* steels have the worst performance since process defects play an important role in fatigue failure, and the dataset does not supply such information. Providing defect information about shape, size, and distribution while simultaneously enlarging the database with further microstructural features about grain size, core hardness, hardness gradient, and so forth could improve the mean prediction and lower the predictive uncertainty. The last group *Steel* has the most database entries and includes a broad range of hardness's. Thus, the performance is slightly worse than for the *Stainless* steels since high-strength steels are also included, involving a variety of fatigue damage mechanisms resulting in a more heterogeneous dataset and thus in higher predictive uncertainty. Further work should focus on data categorizations based on the hardness or predominant failure mechanisms instead of distinguishing between the FKM-steel types to increase the interpretability of the aleatoric/data uncertainty.

AUTHOR CONTRIBUTIONS

Christian Frie: Conceptualization; methodology; software; formal analysis; investigation; writing—original draft. **Anton Kolyshkin**: Methodology; writing—review and editing. **Chris Eberl**: Supervision; writing—review and editing.

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflict of interest.

ACKNOWLEDGEMENTS

Open access funding enabled and organized by Projekt DEAL.

DATA AVAILABILITY STATEMENT

Research data are not shared.

REFERENCES

1. Hennessey C, Castelluccio GM, McDowell DL. Sensitivity of polycrystal plasticity to slip system kinematic hardening laws for al 7075-t6. *Mater Sci Eng: A*. 2017;687:241-248.
2. McDowell DL, Dunne FPE. Microstructure-sensitive computational modeling of fatigue crack formation. *Int J Fatigue*. 2010; 32(9):1521-1542.
3. Natkowski E, Sonnweber-Ribic P, Münstermann S. Determination of fatigue lifetimes with a micromechanical short crack model for the high-strength steel SAE 4150. *Int J Fatigue*. 2022; 156:106621.
4. Natkowski E, Sonnweber-Ribic P, Münstermann S. Surface roughness influence in micromechanical fatigue lifetime prediction with crystal plasticity models for steel. *Int J Fatigue*. 2022;159:106792.
5. Schäfer BJ, Sonnweber-Ribic P, Ul Hassan H, Hartmaier A. Micromechanical modelling of the influence of strain ratio on fatigue crack initiation in a martensitic steel—a comparison of different fatigue indicator parameters. *Materials*. 2019;12(18):2852.
6. Institution BS. *Guide on Methods for Assessing the Acceptability of Flaws in Metallic Structures*. British Standard Institution: London, UK; 1999.
7. Koçak M. Fitnet fitness-for-service procedure: an overview. *Weld World*. 2007;51:94-105.
8. Norsok S. Design of steel structures. N-004, Rev 2; 2004.
9. RP579 API. Recommended practice for fitness-for-service. American Petroleum Institute. 2000.
10. Shen J, Lu M, Peng H, Liu Y, Chen Z. A Simplified fatigue assessment method for ASME VIII-2. *J Pressure Vessel Technol*. 2020;143(1):11502. https://doi.org/10.1115/1.4047712
11. Wächter M, Müller C, Esderts A. *Angewandter festigkeitsnachweis nach fkm-richtlinie*. Springer. 2017.
12. Agrawal A, Deshpande PD, Cecen A, Basavarsu GP, Choudhary AN, Kalidindi SR. Exploration of data science techniques to predict fatigue strength of steel from composition and processing parameters. *Int Mater Manuf Innov*. 2014;3: 90-108.
13. Chen J, Liu Y. Fatigue modeling using neural networks: a comprehensive review. *Fatigue Fract Eng Mater Struct*. 2022;45(4): 945-979.
14. He L, Wang Z, Akebono H, Sugeta A. Machine learning-based predictions of fatigue life and fatigue limit for steels. *J Mater Sci Technol*. 2021;90:9-19. https://linkinghub.elsevier.com/retrieve/pii/S1005030221002607
15. He N, Ouyang R, Qian Q. Learning interpretable descriptors for the fatigue strength of steels. *AIP Advances*. 2021;11(3):35018.
16. Kolyshkin A, Frie C, Froschmeier T. Datenbasierte lebensdauervorhersage. 48 Tagung des DVM-Arbeitskreises Betriebsfestigkeit. 2022. https://dvm-wissen.de/gb/betriebsfestigkeit-2022/240-BF-2022-017.html
17. Weichert D, Kister A, Houben S, Ernis G, Wrobel S. Robustness in fatigue strength estimation. arXiv preprint arXiv: 221201136; 2022.
18. Xiong J, Zhang T, Shi S. Machine learning of mechanical properties of steels. *Sci China Technol Sci*. 2020;63(7):1247-1255.
19. Castillo E, Fernández-Canteli A. *A Unified Statistical Methodology for Modeling Fatigue Damage*: Springer Science & Business Media. 2009.
20. Bishop CM, Nasrabadi NM. *Pattern Recognition and Machine Learning*, Vol. 4. Springer; 2006.
21. Hüllermeier E, Waegeman W. Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Mach Learn*. 2021;110(3):457-506. https://doi.org/10.1007/s10994-021-05946-3
22. Gawlikowski J, Tassi CRN, Ali M, other. A survey of uncertainty in deep neural networks. 2022.
23. Wang H, Yeung D-Y. A survey on bayesian deep learning. 2021.
24. Jospin LV, Laga H, Boussaid F, Buntine W, Bennamoun M. Hands-on Bayesian neural networks—a tutorial for deep learning users. *IEEE Comput Intell Mag*. 2022;17(2):29-48.
25. Lakshminarayanan B, Pritzel A, Blundell C. Simple and scalable predictive uncertainty estimation using deep ensembles. *Adv Neural Inf Process Syst*. 2017;30.
26. Wang H, Yeung D-Y. Towards Bayesian deep learning: a framework and some existing methods. 2016.
27. Gal Y, Ghahramani Z. Dropout as a Bayesian approximation: representing model uncertainty in deep learning. 2016.
28. Kingma DP, Ba J. Adam: a method for stochastic optimization. 2017.
29. Zhang R, Li C, Zhang J, Chen C, Wilson AG. Cyclical stochastic gradient MCMC for Bayesian deep learning. 2020.
30. Welling M, Teh YW. Bayesian learning via stochastic gradient Langevin dynamics. 2011.

31. Amini A, Schwarting W, Soleimany A, Rus D. Deep evidential regression. 2020.
32. Wager S, Hastie T, Efron B. Confidence intervals for random forests: the jackknife and the infinitesimal jackknife. 2014.
33. Nix DA, Weigend AS. Estimating the mean and variance of the target probability distribution. In: Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94), Vol. 1; 1994:55-60 vol.1.
34. Kendall A, Gal Y. What uncertainties do we need in bayesian deep learning for computer vision? *Adv Neural Inf Process Syst*. 2017;30.
35. Cho H, Kim Y, Lee E, Choi D, Lee Y, Rhee W. Basic enhancement strategies when using Bayesian optimization for hyperparameter tuning of deep neural networks. *IEEE Access*. 2020; 8:52588-52608.
36. Probst P, Bischl B, Boulesteix A-L. Tunability: importance of hyperparameters of machine learning algorithms. 2018.
37. Snoek J, Larochelle H, Adams RP. Practical Bayesian optimization of machine learning algorithms. *Adv Neural Inf Process Syst*. 2012;25.
38. Snoek J, Rippel O, Swersky K, et al. Scalable Bayesian optimization using deep neural networks. In: International Conference on Machine Learning PMLR; 2015:2171-2180.
39. Bergstra J, Bardenet R, Bengio Y, Kégl B. *Algorithms for hyperparameter optimization* Shawe-Taylor J, Zemel R, Bartlett P, Pereira F, Weinberger KQ, eds., Vol. 24: Curran Associates, Inc.; 2011. https://proceedings.neurips.cc/paper_files/paper/2011/file/86e8f7ab32cfd12577bc2619bc635690-Paper.pdf
40. Guo C, Pleiss G, Sun Y, Weinberger KQ. On calibration of modern neural networks. 2017.
41. Kuleshov V, Fenner N, Ermon S. Accurate uncertainties for deep learning using calibrated regression. 2018.
42. Lee K, Lee H, Lee K, Shin J. Training confidence-calibrated classifiers for detecting out-of-distribution samples. 2018.
43. Wilson AG, Izmailov P. Bayesian deep learning and a probabilistic perspective of generalization. 2022.
44. Chung Y, Char I, Guo H, Schneider J, Neiswanger W. Uncertainty toolbox: an open-source library for assessing, visualizing, and improving uncertainty quantification. 2021.
45. Basquin OH. The exponential law of endurance tests. *Proc Am Soc Test Mater*. 1910;10:625-630.
46. Castillo E, Fernández-Canteli A, Hadi AS. On fitting a fatigue model to data. *Int J Fatigue*. 1999;21(1):97-106.
47. Köder M. Schwingfestigkeitsnachweis für innendruckbelastete common-rail-bauteile aus 100cr6 unter berücksichtigung hochzyklischer betriebsbeanspruchungen. 2014.
48. Palmgren A. Die lebensdauer von kugellagern. *Verfahrenstechinik*. 1924;68:339-341.
49. Stromeyer CE. The determination of fatigue limits under alternating stress conditions. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Math Phys Char*. 1914; 90(620):411-425.
50. Garwood MF. Interpretation of tests and correlation with service. ASM, 1; 1951.
51. Meyers MA, Chawla KK. *Mechanical Behavior of Materials*. Cambridge University Press. 2008.
52. Murakami Y. *Metal fatigue: Effects of Small Defects and Nonmetallic Inclusions*. Referex Engineering: Elsevier; 2002. https://books.google.de/books?id=zcJmlAEACAAJ
53. Pang JC, Li SX, Wang ZG, Zhang ZF. General relation between tensile strength and fatigue strength of metallic materials. *Mater Sci Eng: A*. 2013;564:331-341.
54. Suresh S. *Fatigue of Materials*. Cambridge University Press. 1998.
55. Furuya Y, Nishikawa H, Hirukawa H, Nagashima N, Takeuchi E. Catalogue of NIMS fatigue data sheets. *Sci Technol Adv Mater*. 2019;20(1):1055-1072. PMID: 31762842.
56. Ellmer F, Esderts A, Eulitz K-G, Hinkelmann K. Forschungskuratorium maschinenbau. Datenbank und auswertesystem betriebsfestigkeit. FKM-Vorhaben; 2011.
57. He N, Ouyang R, Qian Q. Learning interpretable descriptors for the fatigue strength of steels. *AIP Advances*. 2021;11(3): 35018. https://doi.org/10.1063/5.0045561
58. Xiong J, Zhang T, Shi S. Machine learning of mechanical properties of steels. *Sci China Technol Sci*. 2020;63(7):1247-1255.
59. Sander M. *Sicherheit und betriebsfestigkeit von maschinen und anlagen*. Springer; 2008.
60. Abadi M, Agarwal A, Barham P, other. TensorFlow: large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/, Software available from tensorflow.org; 2015.
61. Pedregosa F, Varoquaux G, Gramfort A, other. Scikit-learn: machine learning in Python. *J Mach Learn Res*. 2011;12:2825-2830.
62. Bergstra J, Komer B, Eliasmith C, Yamins D, Cox DD. Hyperopt: a python library for model selection and hyperparameter optimization. *Comput Sci Disc*. 2015;8(1):14008.
63. Dua D, Graff C. UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences. 2017. http://archive.ics.uci.edu/ml
64. Gustafsson FK, Danelljan M, Schon TB. Evaluating scalable Bayesian deep learning methods for robust computer vision. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2020:318-319.
65. Beluch WH, Genewein T, Nurnberger A, Kohler JM. The power of ensembles for active learning in image classification. 2018.
66. Borisov V, Leemann T, Sessler K, Haug J, Pawelczyk M, Kasneci G. Deep neural networks and tabular data: a survey. IEEE transactions on neural networks and learning systems, PP. 2021.
67. Levi D, Gispan L, Giladi N, Fetaya E. Evaluating and calibrating uncertainty prediction in regression tasks. 2020.
68. Rahaman R, Thiery AH. Uncertainty quantification and deep ensembles. 2021.
69. Fleck NA, Kang KJ, Ashby MF. Overview no. 112: The cyclic properties of engineering materials. *Acta Metallurgica et Materialia*. 1994;42(2):365-381. https://www.sciencedirect.com/science/article/pii/0956715194904936
70. Zhang ZF, Wang ZG. Grain boundary effects on cyclic deformation and fatigue damage. *Progr Mater Sci*. 2008;53(7):1025-1099.

How to cite this article: Frie C, Kolyshkin A, Eberl C. Analysis of data-driven models for predicting fatigue strength of steel components with uncertainty quantification. *Fatigue Fract Eng Mater Struct*. 2024;47(3):1036-1052. doi:10.1111/ffe.14195

APPENDIX A: HYPERPARAMETER SEARCH SPACE FOR ML MODELS

TABLE A1 Hyperparameter space for BNN.

Hyperparameter	Range [min, max, (stepsize)]/ Category
Layer 1	[100, 400, 20]
Layer 2	[100, 400, 20]
Layer 3	[0, 400, 20]
Layer 4	[0, 400, 20]
Batch size	[16, 32, 4]
Optimizer	Adam
Activation function	[“Elu,” “Relu,” “Selu,” “Tanh”]
Learning_rate	[0.00001, 0.1] log scale

TABLE A2 Hyperparameter space for DeepEnsembles.

Hyperparameter	Range [min, max, (stepsize)]/ Category
Layer 1	[100, 400, 20]
Layer 2	[100, 400, 20]
Layer 3	[0, 400, 20]
Layer 4	[0, 400, 20]
Batch size	[16, 32, 4]
Optimizer	Adam
Activation function	[“Elu,” “Relu,” “Selu,” “Tanh”]
Learning_rate	[0.00001, 0.1] log scale
L2 regularizer	[0.00001, 0.1] log scale

TABLE A3 Hyperparameter space for MCDropout.

Hyperparameter	Range [min, max, (stepsize)]/ Category
Dropout 1	[0.1, 0.5]
Layer 1	[100, 400, 20]
Dropout 2	[0.1, 0.5]
Layer 2	[100, 400, 20]
Dropout 3	[0.1, 0.5]
Layer 3	[0, 400, 20]
Dropout 4	[0.1, 0.5]
Layer 4	[0, 400, 20]
Batch size	[16, 32, 4]
Optimizer	Adam
Activation function	[“Elu,” “Relu,” “Selu,” “Tanh”]
Learning_rate	[0.00001, 0.1] log scale

TABLE A4 Hyperparameter space for EvidPrior.

Hyperparameter	Range [min, max, (stepsize)]/ Category
Layer 1	[100, 400, 20]
Layer 2	[100, 400, 20]
Layer 3	[0, 400, 20]
Layer 4	[0, 400, 20]
Batch size	[16, 32, 4]
Optimizer	Adam
Activation function	[“Elu,” “Relu,” “Selu,” “Tanh”]
Learning_rate	[0.00001, 0.1] log scale
L2 regularizer	[0.00001, 0.1] log scale

TABLE A5 Hyperparameter space for CSG-MCMC.

Hyperparameter	Range [min, max, (stepsize)]/ Category
Layer 1	[100, 400, 20]
Layer 2	[100, 400, 20]
Layer 3	[0, 400, 20]
Layer 4	[0, 400, 20]
Batch size	[16, 32, 4]
Optimizer	SG-MCMC
Activation function	[“Elu,” “Relu,” “Selu,” “Tanh”]
Learning_rate	[0.00001, 0.1] log scale

TABLE A6 Hyperparameter space for RFJ.

Hyperparameter	Range [min, max, (stepsize)]/ Category
Number of estimators	[1000, 4000, 200]
Max depth	[4, 40]
Min samples split	[2, 10]
Min samples leaf	[1, 3]
Max features	[16, 32]