Proceedings of Machine Learning Research 204:1–13, 2023 Conformal and Probabilistic Prediction with Applications

Neural Networks based Conformal Prediction for Pipeline Structural Response

Sara El Mekkaoui SARA.EL.MEKKAOUI@DNV.COM
Carla J Ferreira CARLA.FERREIRA@DNV.COM
Juan Camilo Guevara Gómez JUAN.CAMILO.GUEVARA.GOMEZ@DNV.COM
Christian Agrell CHRISTIAN.AGRELL@DNV.COM
Group Research & Development DNV AS, Høvik, Norway

Nicholas James Vaughan NICHOLAS.JAMES.VAUGHAN@DNV.COM
Hans Olav Heggen HANS.OLAV.HEGGEN@DNV.COM
Energy Systems DNV AS, Høvik, Norway

Editors: Harris Papadopoulos, Khuong An Nguyen, Henrik Boström and Lars Carlsson

Abstract

The widespread use of machine learning models has achieved considerable success across various domains. Nevertheless, their deployment in safety-critical systems can result in catastrophic consequences if uncertainties are not handled properly. This study is concerned with the simulation of the physical response of a subsea pipeline when it is hooked by an anchor. Predicting this response is crucial for risk assessment, however, it is computationally unfeasible to run a significant amount of input sets to compute the probability of failure of the system. Therefore, the use of a surrogate model becomes essential. In this context, a surrogate model is a machine learning model trained on data from a physics-based simulation. This is achieved by neural network based surrogate models, as they are capable of modelling complex relationships and provide greater accuracy than other machine learning models in many use cases. However, to ensure the safe use of these models, it is important to understand the uncertainty associated with their predictions. Therefore, we apply the conformal prediction framework to provide valid prediction intervals and improve the uncertainty quantification of the neural network models. In order to create adaptive conformal prediction intervals, we employ multilayer perceptron neural network models that provide uncertainty estimates through both the Monte Carlo dropout technique and treating the output as a Gaussian distribution, with the neural network providing estimates for both mean and variance. The conformal prediction procedure improves the uncertainty estimation of uncalibrated models and guarantees new test samples are within the predicted intervals with the corresponding selected confidence level.

Keywords: Inductive Conformal Prediction, Supervised Learning, Neural Networks, Surrogate Models, Safety-Critical Systems

1. Background and motivation

As modern machine learning (ML) methods continue to demonstrate their efficacy in solving complex engineering problems, ensuring the safe use of ML models in critical decision-making and control strategies has become a top priority, particularly in industries where the value of what is produced is high and operational deviations could be dangerous or have a significant negative impact on service.

When it comes to providing quick insights into complex systems based on observations and experience, data-driven models play a crucial role. However, these models must account for the inherent uncertainties in observations and the limited experience base, to be accurate and relevant. Safety can be defined as "freedom from risk which is not tolerable" (ISO). According to this definition, a safe system is one in which scenarios with unacceptably severe consequences have a sufficiently low probability, or frequency, of occurrence. Therefore, the models efficiency metrics should not be based on the predictive power alone but also include a rigorous uncertainty assessment. As uncertainty is essential for assessing risk, methods that include rigorous treatment of uncertainty are preferred (e.g., Bayesian methods, probabilistic inference, and conformal prediction). Incorporating uncertainty analysis into data-driven models is needed as it allows us to assess the reliability of the results, identify potential sources of error, gain a more comprehensive understanding of the system under study, and promote more informed decision-making.

In the past, the safety of complex systems has been addressed by cautiously engineering their capability and carefully examining potential scenarios to which the system may be exposed. Even though many operations are now automated, people are still overseeing the entire system and bearing the ultimate accountability. However, as high-risk and safety-critical systems increasingly rely on ML methods, greater attention is being paid to ensuring their assurance. C. Agrell (2018) identifies three critical challenges when applying machine learning methods to high-risk and low-probability scenarios:

- **A high-risk scenario reduces the tolerance for erroneous predictions**: We cannot accept a decision that may have catastrophic consequences based on a faulty ML algorithm.

- **Critical consequences are often related to tail events - for which data are naturally scarce**: ML methods require data. If the data are scarce, the uncertainty associated with the predictions will be high, and the predictive accuracy will be significantly reduced.

- **ML models that can fit complex data well are often opaque and impenetrable for human understanding**: This makes the model inscrutable and less falsifiable. For a decision maker in a high-risk context, this increases the uncertainty and thus reduces her ability to trust the model.

This paper addresses two of the above-mentioned challenges in ML related to high-risk cases where erroneous predictions could lead to disastrous consequences: the need for accurate predictions and the issue of scarce data. To address these issues, the paper suggests using conformal prediction with neural networks to incorporate uncertainty estimates into predictions. Furthermore, it can potentially mitigate the negative impact of scarce data on predictive accuracy. By using conformal prediction in neural networks, the authors aim to increase the transparency and trustworthiness of ML models for high-risk decision-making.

This paper is organized as follows: section 2 describes the use case considered in this study. Section 3 provides the neural networks and conformal prediction methods. Section 4 presents and discusses the results, and section 5 concludes the paper.

2. Use Case

In this study, we address the problem of pipeline structural response in the case of anchor hooking. If a ship's anchor is dragged along the seabed where a subsea pipeline is lying, it may hook around the pipe and pull it along. This can lead to damage to the pipeline which has the potential to cause failure. The force that a dragged anchor applies to a pipeline can be estimated in a stochastic sense according to a probability distribution for given limiting mechanisms, for example, fluke breaks, anchor chain breaks, etc. With given parameters, the response of the pipeline to an anchor hooking can be simulated with nonlinear 3D Finite Element Analysis (FEA).

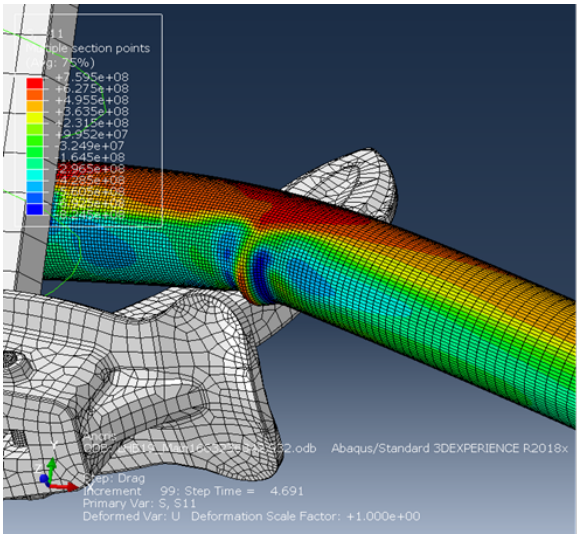

Figure 1: Pipeline being dragged by the anchor as seen in an FEA simulation.

The probability that a dragged anchor will hook a given pipeline can be quantified by considering historic/forecasted ship traffic data and estimated anchoring frequencies along with simple geometric considerations such as fluke length, pipe outer diameter, etc. Generally, there is a distribution of different ship classes and corresponding anchor sizes as well as velocities, contributing to the complexity of the anchor force statistical evaluation. In addition, some of the relevant pipeline parameters such as axial and lateral soil resistance, and depth are typically subject to uncertainty or known variation along a given route.

The Probability of Failure (POF) is the overall governing acceptance criteria for the design. To quantify this, response parameters are compared to established failure criteria. Due to the complexity of the FEA required to predict the pipeline response, it is computationally unfeasible to satisfactorily account for the input statistics and calculate the probability distributions for the response quantities simply by running the FEA enough times with different input sets.

Equinor and GASSCO have identified two goals for a project to make this problem more tractable in practice: (1) calculate the POF for given input statistics of pipeline and ship traffic parameters; (2) calculate the (POF) for a particular anchor hooking incident, given observed lateral displacement and/or dent depth of the pipe's wall.

DNV has addressed these problems with a structural reliability analysis and the Monte Carlo method is used to account for the stochastic nature of the input parameters. To speed up the iterative calculation of the pipeline response within the Monte Carlo method, a surrogate model has been established, in place of the FEA. The surrogate model is a machine learning model which is trained upon a finite set of FEA cases.

The matrix of FEA cases was established by:

- identifying which input dimensions the project required the tool to account for;
- specifying a quantitative range of relevance for each dimension;
- identifying output responses quantities of interest, for the failure criteria;
- after setting a quantity of FEA cases to run, Latin hypercube sampling was used to define the individual cases.

Since the surrogate model is an approximation to the FEA model, there will be some error and this should be included as a 'model uncertainty' in the structural reliability analysis. Relevant outputs include lateral displacement, the bending moment over the pipe cross-section, and the peak axial compressive strain which is the focus of this work. Figure 1 shows the first principal stress distribution in the pipe wall during a hooking incident.

3. Methodology

This study aims to evaluate the uncertainty of neural network based surrogate regression models using tabular data by employing a conformal prediction framework. The dataset used in this study comprises approximately 500 simulation runs, each with 10 data points. Each data point (x_i) consists of eleven input variables, and the corresponding continuous target (y_i) is the quantitative response of interest.

To construct the predictive models, we used a Multilayer Perceptron (MLP) neural network with three hidden layers. The optimal hyperparameters of the MLP model were selected using a three-way holdout validation method (Raschka, 2018) and Bayesian optimization. Two techniques were employed to estimate the prediction uncertainty: Monte Carlo Dropout (MCD) (Gal and Ghahramani, 2016) and an MLP with a probabilistic output layer, henceforth referred to as PMLP. The models' outputs, including point predictions and uncertainty estimates, were used to construct conformal prediction intervals.

To assess the performance of the models, we used a k-folds cross-validation procedure to estimate the test error as depicted in Figure 2. The dataset is divided into k equal-sized folds. The cross-validation procedure iterates k times, where in each iteration, the k^{th} fold is used as a test set. The remaining $k-1$ folds are split randomly into 90% used for training and 10% as conformal prediction calibration set. The model is trained and used to provide predictions of the calibration set examples to construct conformal prediction intervals for the test set. Different performance metrics are used to assess the performance of the model's predictions, uncertainty estimates, and conformal prediction intervals. After completing the k iterations, the performance metrics are summarized, using the mean and variance for instance, to provide an overall performance estimation of the model. In this

study, the cross-validation procedure is performed for each model using 100-folds cross-validation. The data split was based on the simulation runs, ensuring that data points from the same simulation were grouped together in the same set. Subsequent sections detail the MLP predictive models, conformal prediction intervals construction methodology, and performance assessment.

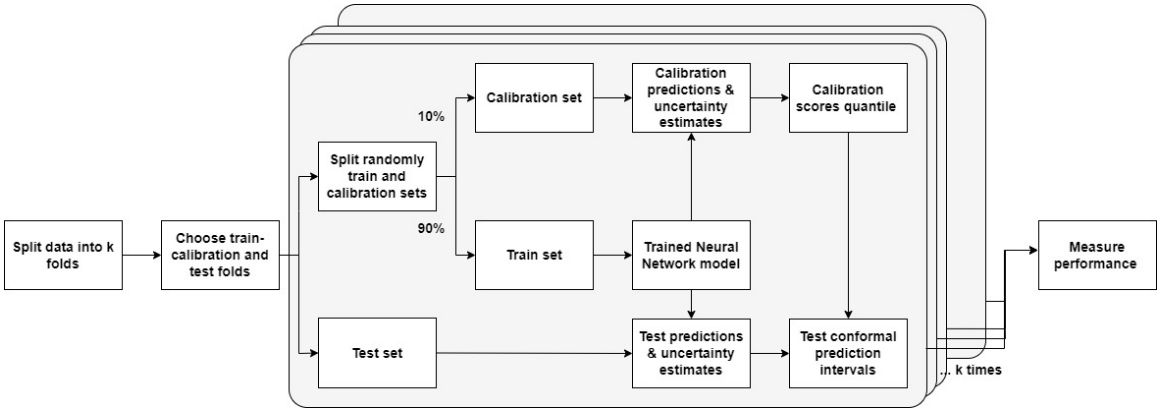

Figure 2: Neural Network based Conformal Prediction procedure

3.1. Predictive models

The dominant paradigm for training neural networks involves the maximization of the likelihood objective, which produces a unique set of model parameters. However, this approach can potentially limit the neural network model's capacity to capture diverse functions that fit the data while also generalizing differently. An alternative approach is to adopt Bayesian inference principles, which allow for the consideration of a range of models with different parameter settings, resulting in improved accuracy and the representation of uncertainty (Murphy (2023)). By leveraging Bayesian inference in neural networks, it becomes possible to account for model uncertainty and produce more reliable predictions. However, Bayesian neural networks are generally more computationally expensive to train due to a large number of parameters and voluminous datasets.

Monte Carlo dropout is a widely used technique to approximate the Bayesian predictive distribution for neural networks (Gal and Ghahramani, 2016). This method involves applying dropout, which is a regularization technique for neural networks, to perform a random sampling of hidden units during test time. This results in the computation of multiple forward pass, generating different models. The mean and the standard deviation of the resulting predictions can then be used as point prediction and uncertainty estimates, respectively. While MCD has shown effectiveness in various areas, such as natural language processing and computer vision, its application should be undertaken with caution. Although intended to represent the learned model's uncertainty (i.e., epistemic uncertainty), it may be an inadequate approximation, as suggested by previous research (see e.g., Osband, 2016; Folgoc et al., 2021). Furthermore, accurate estimates of uncertainty require careful tuning of the dropout rate.

An alternative approach to quantifying the uncertainty in neural network predictions is to model the output as a distribution, for instance, with the assumption that the true output follows a Gaussian distribution with learnable mean and variance parameters. In this study, we consider an MLP with a probabilistic output layer (PMLP). To learn the optimal parameters of the estimated Gaussian distribution, the negative log-likelihood is used as a loss function, which maximizes the likelihood of the target values given the estimated distribution with parameters $\hat{\mu}$ and $\hat{\sigma}$. However, the normal distribution may not always be an appropriate choice for modeling the true output distribution, and the estimated mean and variance values obtained from the model may not necessarily reflect the actual distribution. This could impose a significant inductive bias into the model resulting in a poor performance both for point predictions and uncertainty estimations and leading to wrong interpretations. Therefore, it is crucial to perform further validation to assess the reliability and accuracy of the estimated distribution, which may involve statistical tests and cross-validation techniques.

3.2. Conformal Prediction

Conformal Prediction is a statistical framework that offers a robust method for quantifying uncertainty in predictive models. This methodology can be applied to any predictive model and provides prediction sets or intervals with a guaranteed bound on prediction error at a specified error level. This study specifically considers the inductive conformal regression setting (Papadopoulos et al., 2002), where the underlying model is induced from training data and uses a calibration dataset, to generate conformal prediction intervals for new instances. The procedure followed in this study to construct conformal regressors is described in Algorithm 1.

Constructing a conformal regressor requires choosing the following elements: (i) an underlying model, (ii) a non-conformity score function, and (iii) a significance level. Also, a calibration dataset distinct from the training dataset is needed. This study employs the MLP with Monte Carlo dropout and the PMLP as underlying models. It should be noted that the predictions generated by these models come with variance estimates. In order to measure the non-conformity score, which is used to assess the conformity of new instances that the models have not seen during training, a scoring function is chosen. This scoring function is applied to the calibration set to obtain non-conformity scores. The specified significance level α is used to achieve the desired level of confidence $1 - \alpha$. The non-conformity score corresponding to the $(1 - \alpha)^{th}$ percentile from the calibration dataset is then used to construct prediction intervals on test data. As the non-conformity scoring function is the standardized residuals, the prediction intervals obtained are adaptive.

3.3. Performance Assessment

The performance accuracy of the predictive models is evaluated using the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE). To quantitatively measure the quality of the predictive models' uncertainty estimation, the Mean Absolute Calibration Error (MACE) (Chung et al., 2021; Fenlon et al., 2018), the Average Negative Log Predictive Density(NLPD) (Quinonero-Candela et al., 2006), and the Continuous Ranked Probability Score (CRPS) (Gneiting and Raftery, 2007) are used.

Algorithm 1: Inductive Split Conformal Prediction

Input: Training data $\{(\mathbf{x_i}, y_i)\}_{i \in D_{train}}$, calibration data $\{(\mathbf{x_i}, y_i)\}_{i \in D_{calib}}$, test data $\{(\mathbf{x_i}, y_i)\}_{i \in D_{test}}$, predictive model P, non-conformity score s, significance level $\alpha \in [0, 1]$.

Output: $P_{cp}(\mathbf{x}_{test})$, prediction intervals of test examples.

Fit the MLP network on training data.

Compute non-conformity scores of calibration data $S = s(\mathbf{x_i}, y_i)_{i \in D_{calib}}$ using the score function $s(\mathbf{x_i}, y_i) = \frac{|y_i - \hat{y}_i|}{\hat{\sigma}_i}$ where $\hat{y}$ and $\hat{\sigma}$ are the point prediction and the standard deviation estimate produced by P for the sample x.

Compute the $\hat{q}$ the $\frac{(m+1)(1-\alpha)}{m}$ quantile of S where m is the size of D_{calib}

for x_i, $i \in D_{test}$ **do**

| Get the point prediction $\hat{y}_i$ and the standard deviation estimate $\hat{\sigma}_i$.

end

return $P_{cp}(\mathbf{x}_i) = [\hat{y}_i - \hat{\sigma}_i \hat{q}, \hat{y}_i + \hat{\sigma}_i \hat{q}]$

The MACE estimates the overall level of error during the calibration of n samples by averaging the absolute difference between the predicted probability $p_i^{(b)}$ and observed frequency $o_i^{(b)}$ for a bin b as given equation (1).

$$MACE^{(b)} = \frac{1}{n} \sum_{i=1}^{n} |p_i^{(b)} - o_i^{(b)}| \tag{1}$$

The NLDP is given by equation (2) where $\hat{y}_i$ is the prediction and $p(.)$ is the probability density function. It quantifies the discrepancy between the predicted probabilities and the true probabilities. The NLDP penalizes both over and under-confident predictions but also favors conservative models, that is models tending to be under-confident rather than over-confident.

$$NLPD = -\frac{1}{n} \sum \log p(\hat{y}_i = y_i | \mathbf{x_i})_{i=1}^{n} \tag{2}$$

The CRPS is a measure for probabilistic predictions of scalar observations. It is a quadratic measure of the difference between the cumulative distribution function (CDF) of the predicted probabilities and the empirical CDF of the observed outcomes. The CRPS is given by equation (3) as the integral of the squared difference between the predicted CDF denoted as F, and the empirical CDF of the scalar observation given by the indicator function $\mathbb{1}(x \geq y)$ defined as a Heaviside step function.

$$CRPS(F, y) = \int_{\mathbb{R}} [F(x) - \mathbb{1}(x \geq y)]^2 dx, \tag{3}$$

To evaluate the performance of the conformal prediction intervals, two commonly used metrics were employed: coverage and efficiency. Coverage is defined as the ratio of test targets that were covered by the intervals, while efficiency is defined as the mean length of the test intervals (Angelopoulos and Bates, 2022). These metrics are widely used in the literature to assess the accuracy and effectiveness of conformal prediction intervals and can

provide valuable insights into the reliability and usefulness of the generated intervals in practical applications.

4. Results and Discussion

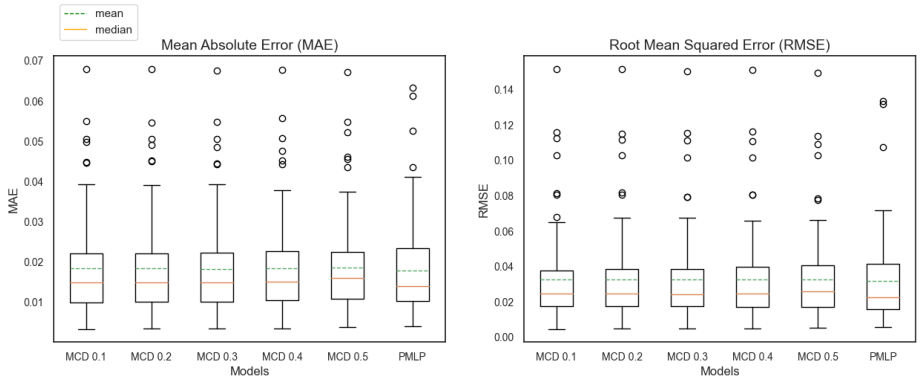

Figure 3: Accuracy metrics of the Monte Carlo Dropout for dropouts rates 0.1 - 0.5 and for the Probabilistic Multilayer Perceptron. The plot shows the distribution of the metrics from the 100-folds cross-validation. In each case, the dashed green line indicates the mean, and the orange line the median.

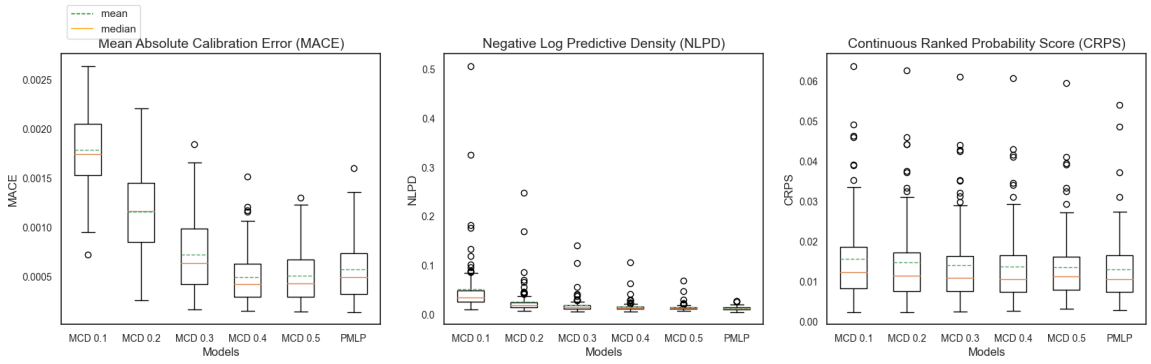

Figure 4: Uncertainty performance metrics of the Monte Carlo Dropout for dropouts rates 0.1 - 0.5 and for the Probabilistic Multilayer Perceptron. The plot shows the distribution of the metrics from the 100-folds cross-validation. In each case, the dashed green line indicates the mean, and the orange line the median.

In order to define what is the best approach to the problem we have used the Uncertainty Toolbox[1] (Chung et al., 2021) to estimate the MAE, RMSE, MACE, NLPD, and CRPS

1. See https://github.com/uncertainty-toolbox/uncertainty-toolbox

for the PMLP and the MCD, the latter with 5 dropout rates from 0.1 to 0.5. Figure 3 and 4 show the distribution of these performance metrics in each case. The distributions are built with the outputs of each 100-folds as depicted in figure 2. From the MAE and RMSE metrics, we can conclude that there is not a substantial difference in performance among the models. The MACE metric shows that MCD with 0.4 and 0.5 dropout rates are the best at calibrating the predicted probabilities with the observed frequencies, while the NLPD metric shows that MCD with a 0.5 dropout rate performs slightly better than the 0.4 dropout rate. Therefore the former is also the model whose predictions are more likely, given the data. The PMLP shows similar results but still, its performance is slightly poorer.

Figure 5 shows the predictions ($\pm$ 1 standard deviation) vs true values and the calibration for the models. The calibration plot refers to the degree to which the targets fall within the uncertainty ranges. It shows the proportion of the test data we expect to lie inside the uncertainty range on the x-axis, and the observed proportion of test data inside the uncertainty ranges on the y-axis. The plot is made by dividing the interval $[0, 1]$ to 100 bins or quantile ranges. For each quantile, we calculate the proportion of test points falling in the interval corresponding to that quantile. This is done by computing the proportion of normalized residuals within the lower and upper bound for that quantile. The model is considered well-calibrated when the blue line falls over the dashed orange line. The shaded area represents the variance of the values from the 100-folds cross-validation procedure. The upper and lower bounds are the 0.9 and 0.1 quantiles, respectively. The blue line corresponds to the mean across the cross-validation values. If the line lies below the diagonal, the model is overconfident with too narrow uncertainty ranges. Otherwise, the model is underconfident with wide uncertainty ranges. As already discussed, the MCD with 0.4 and 0.5 dropout rates seem to be the models that best perform on the data followed by the PMLP, according to the calibration plots.

Figure 6 shows the performance metrics for the conformal prediction intervals and the neural network models' uncertainty. To properly evaluate the performance, it is necessary to establish a trade-off between coverage and efficiency metrics. Efficiency is a measure of how narrow the prediction intervals are. The smaller the interval the more precise the prediction. While coverage represents how often the true outcome falls within the predicted interval. Having a better efficiency may lead to a lower coverage, as it may happen that narrower intervals exclude the true outcome more often. Hence, we need to choose prediction intervals that are sufficiently narrow (efficient) and provide high confidence that the true value is falling within the interval. The plots in Figure 6 show the distribution of the metrics from the 100-folds cross-validation procedure. The first two panels from the left represent the conformal prediction intervals performance after applying the framework on each model for 0.1, and 0.05 significance levels. The third panel shows the metrics related to two standard deviation uncertainty ranges from the neural network models. Herein, the criterion of two standard deviations is applied because it would cover around 95% of the possible values if the prediction were normally distributed. The conformal prediction with a confidence level of 95% (0.05 significance) guarantees a 95% coverage for all the neural network models, especially the badly calibrated ones, as it is noticeable for the MCD models with dropout rates of 0.1, 0.2, 0.3 and 0.4 in the coverage plots in Fig. 6. The conformal prediction framework also improves the efficiency of the PMLP model. In the same figure,

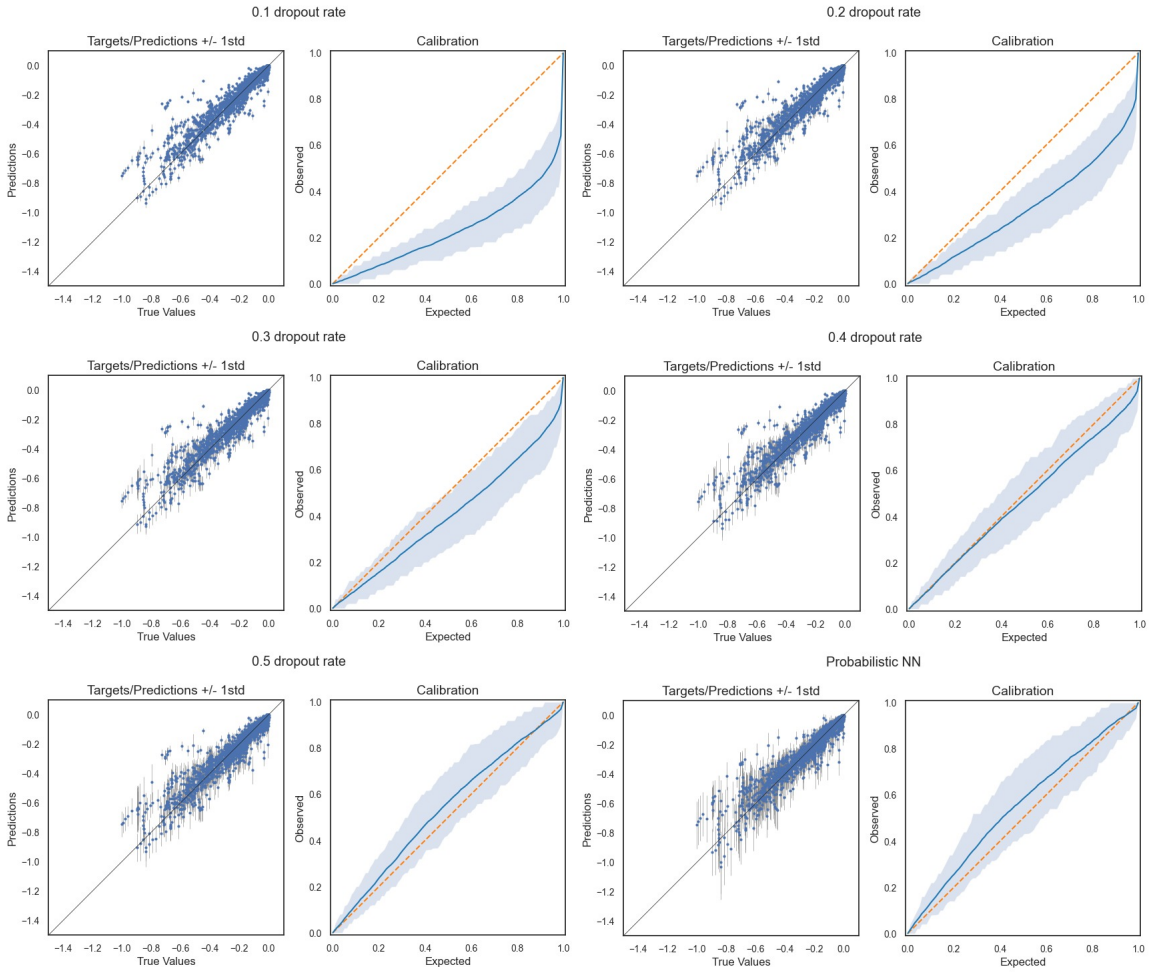

Figure 5: Prediction error (left panels) and calibration plots (right panels) with neural network models uncertainty estimates. The diagonal grey line represents where the samples would fall if the model predictions were perfect. In the calibration plots, the dashed orange line represents how the data should distribute when the model is perfectly calibrated, while the blue line shows the actual distribution of the model's outcomes. The shaded region corresponds to the upper and lower bounds from the 100-folds cross-validation, given by the 0.9 and 0.1 quantiles respectively.

the fluctuation of the conformal prediction coverage and efficiency between the different 100-folds can be attributed to the finite size of the test folds. As the average coverage is approximately equal to $1 - \alpha$ for all cases, we can conclude that the conformal prediction intervals have the correct coverage. For the 0.05 significance level (95% confidence level), both the MCD with 0.4 and 0.5 dropout rates and the PMLP have good coverage. However, the MCD gives better efficiency performance. Moreover, although the coverage and

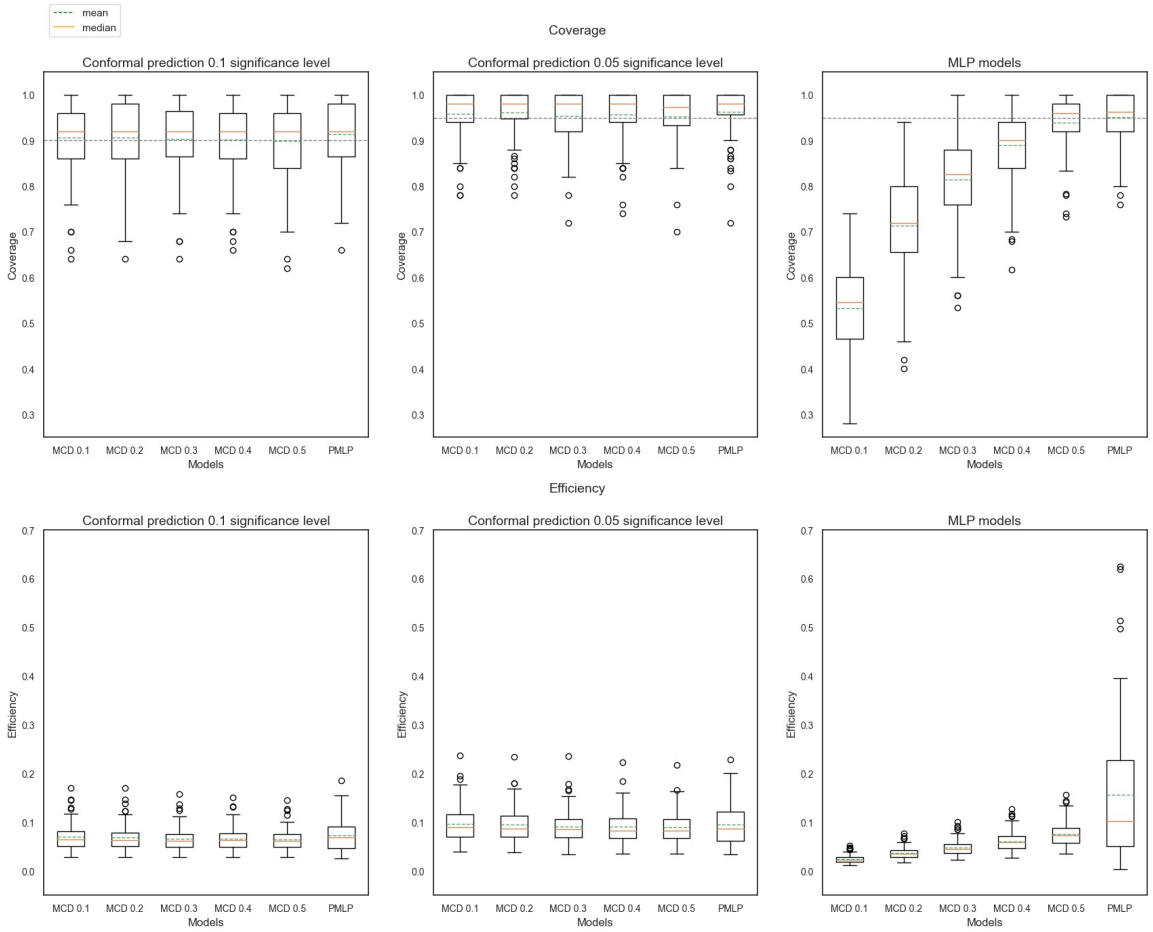

Figure 6: Coverage (top row) and efficiency (bottom row) performance of the conformal prediction intervals and the neural network models uncertainty estimates. The plot shows the distribution of the coverage and efficiency from the 100-fold cross-validation. From left to right: the first two panels correspond to the conformal prediction intervals with significance of 0.1, and 0.05. The last panel corresponds to neural network models with uncertainty ranges of 2 * standard deviations.

efficiency for the MCD 0.4 and 0.5 dropout rates under the conformal prediction frameworks are similar, the coverage for the 0.5 dropout rate in the MLP model, i.e., before applying conformal prediction, is much better than for the 0.4 dropout rate. We, therefore, choose to select the 0.5 dropout rate as the one that best performs given the data within the MCD approach.

The MCD with 0.5 dropout rate provides a coverage of 93.88±0.06 with an efficiency of 11.38±3.67, while the corresponding conformal prediction intervals have a coverage of 95.29±0.06 and an efficiency of 13.40±4.79. In this case, the conformal prediction intervals offer a slight improvement in coverage with larger intervals.

The use of conformal prediction guarantees that new observations will fall within the predicted interval with a selected confidence level. And this provides outstanding information to assess risk and make appropriate decisions. Moreover, the behavior of the intervals when the model is exposed to new data can be used to find changes in the underlying distribution of the data or to detect anomalies and errors in the training data. For instance, if the predicted intervals widen, increasing the uncertainty, or if the observed values consistently fall outside the predicted intervals, it may be that the distribution of data may have shifted due to the presence of noisy inputs or outliers. Although the conformal prediction is unable to explicitly find what is the cause of these behaviors, it does give alerts to the model users to investigate further and make more informed decisions or to apply additional processing in order to reduce the impact of unreliable data on the performance of the model.

5. Conclusions

This study considers the case of the simulation of a subsea pipeline structural response in the situation of anchor hooking. It addresses the challenge of quantifying neural networks based surrogate models uncertainty. In conclusion, the application of conformal prediction on neural networks has shown promising results in improving uncertainty quantification. By providing reliable prediction intervals, conformal prediction can help mitigate the risk of making incorrect decisions based on poor-quality uncertainty estimates from surrogate models.

Overall, this study shows that the combination of neural networks and conformal prediction can significantly enhance the reliability of machine learning based surrogate models, paving the way for their adoption in more complex and critical applications. Further research in this area is required to fully explore the capabilities of this approach by designing more suitable scoring functions and considering other validation strategies. Future work also considers extending this approach to other safety-critical areas.

Acknowledgments

The authors are grateful for the opportunity to showcase the use of conformal prediction for better-informed decision-making on a high-consequence, real-world system provided by Equinor and Gassco. The authors are grateful for the financial support received by the Research Council of Norway and the partners SINTEF and Dr. Techn. Olav Olsen to the project RaPiD–Reciprocal Physics and Data-driven models (grant no. 313909).

References

Anastasios N. Angelopoulos and Stephen Bates. A gentle introduction to conformal prediction and distribution-free uncertainty quantification, 2022.

A. Hafer F.B. Pedersen E. Stensrud C. Agrell, S. Eldevik. Pitfalls of machine learning for tail events in high-risk environments. In *Proceedings of European Safety and Reliability Conference*, Trondheim, Norway, 2018. ESREL.

Youngseog Chung, Ian Char, Han Guo, Jeff Schneider, and Willie Neiswanger. Uncertainty toolbox: an open-source library for assessing, visualizing, and improving uncertainty quantification. *arXiv preprint arXiv:2109.10254*, 2021.

DNV. https://www.dnv.com/. [Online; accessed 10-May-2023].

Equinor. https://www.equinor.com/. [Online; accessed 10-May-2023].

Caroline Fenlon, Luke O'Grady, Michael L. Doherty, and John Dunnion. A discussion of calibration techniques for evaluating binary and categorical predictive models. *Preventive Veterinary Medicine*, 149:107–114, 2018. ISSN 0167-5877. doi: https://doi.org/10.1016/j.prevetmed.2017.11.018. URL https://www.sciencedirect.com/science/article/pii/S0167587717302751.

Loic Le Folgoc, Vasileios Baltatzis, Sujal Desai, Anand Devaraj, Sam Ellis, Octavio E. Martinez Manzanera, Arjun Nair, Huaqi Qiu, Julia Schnabel, and Ben Glocker. Is mc dropout bayesian?, 2021.

Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA, 20–22 Jun 2016. PMLR.

GASSCO. https://www.gassco.no/en/. [Online; accessed 10-May-2023].

Tilmann Gneiting and Adrian E Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

ISO. ISO/IEC Guide 51:2014 Safety aspects — Guidelines for their inclusion in standards. Standard, International Organization for Standardization, 2014.

Kevin P Murphy. *Probabilistic machine learning: Advanced topics*. MIT Press, 2023.

Ian Osband. Risk versus uncertainty in deep learning: Bayes, bootstrap and the dangers of dropout. In *NIPS workshop on bayesian deep learning*, volume 192, 2016.

Harris Papadopoulos, Kostas Proedrou, Volodya Vovk, and Alex Gammerman. Inductive confidence machines for regression. In Tapio Elomaa, Heikki Mannila, and Hannu Toivonen, editors, *Machine Learning: ECML 2002*, pages 345–356, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg. ISBN 978-3-540-36755-0.

Joaquin Quinonero-Candela, Carl Edward Rasmussen, Fabian Sinz, Olivier Bousquet, and Bernhard Scholkopf. Evaluating predictive uncertainty challenge. *Lecture Notes in Computer Science*, 3944:1–27, 2006.

S. Raschka. Model evaluation, model selection, and algorithm selection in machine learning. *ArXiv*, abs/1811.12808, 2018.