

Uncertainty Quantification Based on Bayesian Neural Networks for Predictive Quality



Simon Cramer, Meike Huber, and Robert H. Schmitt

Abstract In the context of production metrology, the field *Predictive Quality* develops methods based on statistics and machine learning to predict quality characteristics from process data. In prior work, conventional machine learning methods such as feed-forward neural networks have been successfully applied. Yet, an uncertainty quantification for the prediction is not provided. Therefore, it is not possible to prove the suitability of the applied predictive quality methods for quality inspections. However, we can estimate the uncertainty by taking a Bayesian perspective and utilizing suitable algorithms.

Here we define *Prediction of Quality Characteristics* (PQC), which is the foundation for every Predictive Quality application. We extend our definition of PQC into a general Bayesian framework to interpret predicted quality characteristics. As an example, we show how Bayesian neural networks are applied to PQC to estimate the uncertainty of every prediction. We interpret the results in the industrial context and determine the suitability of the PQC method.

Our results demonstrate that the application of Bayesian methods is highly promising to get Predictive Quality recognized in industry as an accredited method for quality inspections.

Keywords Predictive quality · Prediction of quality characteristics · Bayesian neural networks · Uncertainty quantification

S. Cramer (✉) · M. Huber
WZL, RWTH Aachen University, Aachen, Germany
e-mail: s.cramer@wzl.rwth-aachen.de; m.huber@wzl.rwth-aachen.de

R. H. Schmitt
WZL, RWTH Aachen University and Fraunhofer IPT, Aachen, Germany
Fraunhofer IPT, Aachen, Germany
e-mail: r.schmitt@wzl.rwth-aachen.de

1 Prediction of Quality Characteristics

As Industry 4.0 strategies are rolled out progressively, process data is becoming accessible in large amounts. The available data offers engineers and scientist innumerable opportunities to analyze and improve production processes. Some exemplary applications are predictive maintenance and process mining [23]. The research field *Predictive Quality* describes the user's ability to optimize product and process-related quality characteristics by using data-driven forecasts as a basis for actions to be taken [5]. The foundation for all predictive quality applications is the *prediction of quality characteristics* (PQC).

The prediction those characteristics can be regarded as a virtual inspection process, as it replaces a physical inspection.

In conventional physical inspection processes for determining product quality, a specific operation (e.g., measuring or gauging) is used to decide whether a quality characteristic meets a pre-defined requirement. In order to make this decision, it is checked whether the considered quality characteristic lies within previously defined specification limits.

Since every inspection process is subject to uncertainties (e.g., due to the uncertainty of the underlying measurement process), the decision whether the characteristic meets the requirement is also uncertain. Due to the uncertainty of inspection results, an erroneous decision is possible. Characteristics that are within the specification limits are rejected (α -error), and characteristics that are outside the specification limits are accepted (β -error). Both errors entail technical, economic, and legal consequences. To reduce the risk of a wrong decisions, the limits of conformity are narrower than the specification limits to account for the uncertainty of the inspection process (e.g., the measurement uncertainty). To guarantee a product within the specification limits, the process variance, the variance of the test process, and the specification limits must be aligned according to DIN EN ISO 14253-1 (see Fig. 1) [41].

In order to consider an inspection process as *suitable*, it must be ensured that the quotient of uncertainty of the inspection process U and tolerance of the considered quality characteristic T does not exceed a certain threshold. This threshold value is defined differently in various standards and guidelines (see MSA [18], VDA5 [40], ISO 22514-7 [42]). As a rule of thumb, the *golden rule of metrology* states that the ratio U/T should not be greater than one-tenth to one-fifth [28, 39]. To deploy PQC in industry, the suitability of the (virtual) inspection process must be guaranteed. Hence, the uncertainty of the underlying model must be quantified. The determination of the uncertainty of a model is a typical example from the mathematical field of Uncertainty Quantification [37].

Uncertainty Quantification (UQ) focuses on the quantitative characterization of uncertainties in both real and computer-based applications. UQ methods are used to quantify the probability of certain results if some or all input variables are uncertain. A mathematical model is used to describe the system's behavior extracted from the measured data. UQ problems are divided into two classes: forward uncertainty

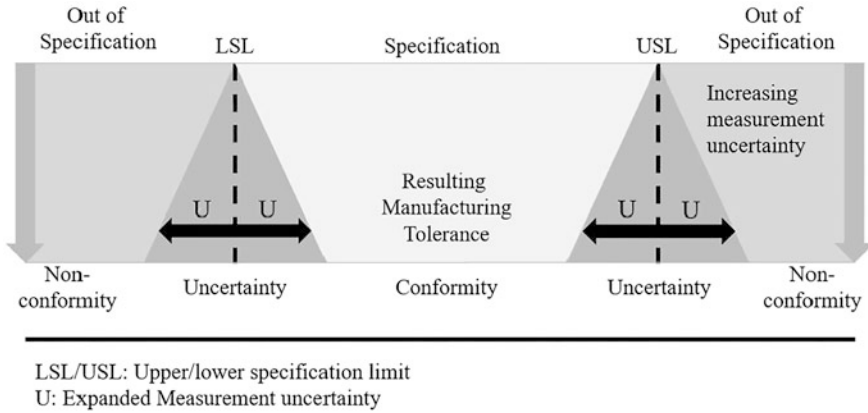


Fig. 1 Limitation of the specification range due to measurement uncertainty according to ISO 14253-1 (see [28, 41])

propagation and inverse uncertainty quantification. Forward uncertainty propagation aims to estimate the different sources of uncertainty, acting on a model to predict an overall uncertainty of the system response. Inverse uncertainty quantification involves estimating the so-called bias correction (i.e., the discrepancy between the measured value and the model) and unknown parameters of the model [6, 37].

In PQC, we estimate the parameters for a given model structure from data. The data used for parameter estimation are usually measurement data and, therefore, affected by uncertainty [28]. For a given model structure and some data, the objective is to minimize the model prediction’s uncertainty by setting the parameters appropriately. The determination of uncertainty in the field of predictive quality can, therefore, be considered an inverse uncertainty quantification problem by definition [37].

2 Definition of Prediction of Quality Characteristics

We first define PQC in a deterministic way before introducing a Bayesian perspective. The definition is provided for a single product in discrete manufacturing. Thus, the index $i \in \mathbb{N}$ identifies a unique part of one product type. With minor modifications, the definition of PQC can be extended to the process industry. The foundation for any machine learning (ML) application is a sufficient database. In the case of PQC it contains the *quality characteristics* and the *process data* on a per-part basis. PQC is an inverse problem, as we want to infer a function H from some infinite-dimensional function space predicting the quality characteristics from process data [37].

We define process data and quality characteristics before constructing a database and deriving the resulting inverse problem.

Definition 1 The *process data* x_i for part i is generated by $m \in \mathbb{N}$ sensors, where the readings of every sensor s_j $0 \leq j < m$ are given as a function of time $s_j : T \rightarrow S$ with $t \in T \subset \mathbb{R}^+$. Accordingly the process data is modelled by $x_i : T \rightarrow S^m$ with $x_i(t) := [s_0(t), \dots, s_m(t)]^T$.

Definition 2 The measurements of the *quality characteristics* $y_i \in \mathbb{R}^n$ for part i are given by $n \in \mathbb{N}$ measurements, where every measurement v_l $0 \leq l < n$ is a fixed value $y_i := [v_l]^T$.

In comparison to the process data x_i we assume that the quality characteristics are time-invariant—or measured only once. Based on Definitions 1 and 2 the data for a unique part i is given by the tuple (x_i, y_i) . Hence, we denote $\mathcal{D} := \{(x_i, y_i)\}$ ($0 \leq i < k$) the database for a given PQC application with $k \in \mathbb{N}$ entries.

Given the database \mathcal{D} we want to determine the parameters $\mathbf{w} \in \mathbf{W}$ of the mapping $H\mathbf{w}$ with

$$y_i = H\mathbf{w}(x_i) \quad \forall (x_i, y_i) \in \mathcal{D}. \quad (1)$$

Thus, the inverse problem has become a *parameter estimation* problem, which is usually ill-posed [37]. A common approach is the computation of a least-squares solution:

$$\arg \min_{\mathbf{w}} \|y_i - H\mathbf{w}(x_i)\|_{\mathcal{D}}^2. \quad (2)$$

Note here that some kind of regularization usually improves the solution as noise in the data is considered [37]. The presence of noise in the data motivates the expansion of this deterministic interpretation of the parameter estimation using a Bayesian perspective.

The measurement of a quality characteristic is subject to measurement uncertainty; thus, it is better represented by a random variable. All sensor readings are also subject to measurement uncertainty and hence—to preserve the time dependency—interpreted as a stochastic process, which we define as follows:

Definition 3 Let $u(t, \omega) : T \times \Omega \rightarrow S$ be a *stochastic process*, where $t \in T \subset \mathbb{R}^+$ and $\omega \in \Omega$. Here Ω is the sample space of the probability space (Ω, \mathcal{F}, P) with \mathcal{F} being a σ -algebra and P a probability measure.

Accordingly we give the definitions of process data and quality characteristics in the Bayesian sense:

Definition 4 The *process data* X is generated by $m \in \mathbb{N}$ sensors, where the sensor readings u_j $0 \leq j < m$ are given by a stochastic process. Accordingly the process

data is modelled by $X : T \times \Omega^m \rightarrow S^m$ with $X(t, \bar{\omega}) := [u_j(t, \omega_j)]^T$ where $\bar{\omega} := [\omega_j]^T$.

Definition 5 The measurements of the *quality characteristics* $Y : \Omega^n \rightarrow \mathbb{R}^n$ are given by $n \in \mathbb{N}$ measurements, where every measurement v_l $0 \leq l < n$ is a random variable $Y(\bar{\omega}) := [v_l(\omega_l)]^T$ where $\bar{\omega} := [\omega_l]^T$.

Based on Definition 4 and 5 the data of a single part i is given by (x_i, y_i) , where $(x_i = X(\cdot, \bar{\omega}_i), y_i = Y(\cdot, \bar{\omega}_i))$ is a realization of (X, Y) . Taking a Bayesian point of view, Eq. (1) introduces the conditioned random variable $Y|X, \mathbf{w}$ and the solution to the inverse problem is the conditioned random variable $\mathbf{w}|\mathcal{D}$ [37]. The parameters can be determined with maximum likelihood estimation (MLE) as

$$\mathbf{w}^{MLE} = \arg \max_{\mathbf{w}} \log P(\mathcal{D}|\mathbf{w}) = \arg \max_{\mathbf{w}} \sum_i \log P(y_i|x_i, \mathbf{w}) \tag{3}$$

or by introducing a prior $P(\mathbf{w})$ on the parameters and finding the maximum a posteriori (MAP) parameters

$$\mathbf{w}^{MAP} = \arg \max_{\mathbf{w}} \log P(\mathbf{w}|\mathcal{D}) = \arg \max_{\mathbf{w}} \log P(\mathcal{D}|\mathbf{w}) + \log P(\mathbf{w}). \tag{4}$$

Example 1 Let the product have $n = 2$ quality characteristics, and the total amount of sensors on the involved machinery be $m = 3$. Then the database \mathcal{D} is constructed from Table 1. For sensor $j = 0$ there are two readings, for sensor $j = 1$ there is one reading and for sensor $j = 2$ there are three readings. We append all sensor readings into a single vector $x \in \mathbb{R}^6$. The same procedure applies to the quality characteristics, which form the vector $y \in \mathbb{R}^2$.

Assume that $H_{\mathbf{w}}(x) := \mathbf{w}x = y$ is a linear operator with $\mathbf{w} \in \mathbb{R}^{2 \times 6}$, then the least-squares solution $\hat{\mathbf{w}}$ according to Eq. (2) is

$$\hat{\mathbf{w}} \approx \begin{pmatrix} -12.92 & -89.1 & 1.69 & 3.35 & -3.17 & 24.28 \\ 0.93 & 0.12 & -0.002 & 0.04 & 0.002 & -0.16 \end{pmatrix}. \tag{5}$$

Table 1 Database entries for an exemplary predictive quality application

Part i	x_i			y_i	
	0	1	2	0	1
0	[0.49, 0.4]	[29]	[3.7, 3.7, 3.8]	100.1	0.02
1	[0.52, 0.39]	[27]	[3.6, 3.8, 3.9]	98.99	0.03
2	[0.5, 0.42]	[30]	[3.7, 3.6, 3.8]	100.2	0.03
3	[0.49, 0.37]	[29]	[3.6, 3.7, 3.7]	100.01	0.028
4	[0.52, 0.4]	[27]	[3.6, 3.2, 3.9]	100	0.03
5	[0.51, 0.42]	[30]	[3.5, 3.6, 3.8]	99.4	0.031

3 State of Uncertainty Quantification for Predictive Quality

The formal proof of suitability requires a determination of the measurement uncertainty. We present the results of our literature review regarding the prediction of quality characteristics and on uncertainty quantification in (deep) machine learning. The UQ methods are designated keystones to provide a measurement uncertainty for PQC applications.

Current State of Predictive Quality The quality of product depends on the interaction of the individual production steps and the condition of the components/machinery and material characteristics. Due to the increasing complexity in production processes, the number of interactions between individual processes is rising. Further, the increasing individualization of products leads to a significant increase in process variance [5].

To improve the understanding of products and processes in production engineering, data analytics methods are used to extract information from data and derive actions based on this information [15, 35]. In this sense, data analytics describes the steps of data investigation, data understanding, and knowledge acquisition, which aim to uncover new relationships within the production process [11]. There are many different methods for the implementation of this decision support, starting with statistical methods up to complex machine learning models, which differ in their application and depend on various factors such as purpose, expertise, and available resources. Data analytics methods can be categorized as descriptive analytics, diagnostic analytics, predictive analytics, or prescriptive analytics. The categories can be seen as steps in the data analysis, which partly rely on each other [26].

Considering the categories, PQ focuses on the application of *predictive analytics* to determine product quality based on process data [5]. Besides considering data from different process steps, existing information on intermediates and the individual assembly can also be taken into account. This enables a comprehensive optimization of the production process. By including data from product usage, the fulfillment of customer requirements can be increased [16, 36].

In recent years, the use of ML algorithms for PQC has been investigated in a manifold of applications. Especially the use of neural networks has shown potential for predicting quality characteristics, as they are capable of mapping and detecting complex cause-effect dependencies while the user is not required to contribute a high amount of expert knowledge [28, 34]. For example, Chen et al. used a back-propagation neural network algorithm and the Taguchi method for quality prediction in plasma-enhanced chemical vapor deposition for semiconductor manufacturing already in 2007 [12]. Ogordnyk et al. introduce a neural networks approach for PQ in the injection molding process. The task here was to classify the product quality based on 18 machine and process parameters [30]. Baturynska et al. describe a prediction model for selective laser sintering. They use neural networks to predict the deviation of manufactured parts in three dimensions depending on their orientation and positioning in the 3D printer [3].

The examples have in common that a model is set up to predict quality characteristics without quantifying the model's uncertainty. Thus, no proof of suitability is obtained, making the use as an inspection tool in an industrial environment challenging. There are, however, machine learning methods which can be used to quantify the uncertainty of the model. These are introduced in the following.

Uncertainty Quantification in (Deep) Machine Learning In the rise of (deep) machine learning since the 2010s, the importance of UQ has been underestimated in the scientific community. As adoption of ML progresses in industrial and consumer applications, safety and security regulations make some types of UQ necessary: verification, robustness, and interpretability [13]. Verification of a ML system provides formal guarantees about its behavior [8, 33, 44]. The robustness (i.e., the reaction to novel/noisy data) is highly relevant for industrial applications, as self-learning robots, and consumer applications, as autonomous vehicles [10, 27, 32]. Interpretability is another active field, where researchers try to understand why an ML system behaves a certain way [31]. We argue that verification and robustness are a form of UQ and that at least a subset of interpretability can be classified as UQ. In all cases, uncertainty in the model or the data are investigated.

Uncertainty in the data and the model are studied using Bayesian approaches since 1989. Early examples of Bayesian learning and Bayesian approaches to neural networks are [25] and [22]. In the 1980s, data sets were significantly smaller than today, and computational power was expensive. Since, the definition of UQ has been significantly expanded. Sullivan et al. consider the treatment of all uncertainties in real and computer-based applications [37]. Especially in the simulation community, where finite element and finite volume methods and their variants are commonly used, UQ did not gain traction until the early 2000s [43]. This was mainly due to the curse of dimensionality and the lack of computational power to perform the simulations for all parameter sets to be investigated [4]. The development of improved methods (e.g., sparse collocation) opened novel possibilities to overcome the curse of dimensionality and explore large parameter spaces efficiently [37].

In deep learning, there are three main movements for UQ [9]. There is Concrete Dropout [14]. The dropout rate becomes a learnable parameter, and nodes are dropped during the evaluation. Thus, a sample from a posterior distribution is generated from a single neural network by randomly omitting a certain percentage of neurons in each layer at each evaluation. This method is an extension to Dropout, which is used as a regularization method to prevent overfitting during model training [19]. Secondly, Deep Ensembles, as introduced in [24], are more sophisticated than Concrete Dropout. Depending on the algorithm's variant, multiple neural networks are trained with different initializations and on different data subsets. At the evaluation, the outputs of all the neural networks are interpreted as samples from a posterior distribution. If we expand the number of models to infinity, we converge to Bayesian Neural Networks (BNN). For a BNN, the weights of each layer are represented by a probability distributions [17]. These networks are evaluated by sampling multiple times from the posterior distributions. In [20] a

different classification is discussed, which takes other approaches into account that do not apply to PQC.

BNN are capable of representing aleatoric uncertainty (e.g., variability in the data) and epistemic uncertainty (e.g., model neglecting effects or missing data) via the posterior distribution [7]. This is a crucial feature for PQC applications as by Definitions 4 and 5 we have (commonly) unknown uncertainty in our data and no indication whether an employed model structure is sufficiently expressive. Even though we have seen successful applications of neural networks to PQC (cmp. [3, 12, 30] and more), assumptions regarding the structure or the hyperparameters of the models may be inherently flawed. BNN are successfully applied to various disciplines as physics [38], civil engineering [1], and others [2, 21, 45]. The BNN have shown excellent results, not only on theoretical toy problems (cmp. [7]) but in real world applications. Thus, we focus on BNN given their benefits and apply them to production engineering, and in particular to PQC. We demonstrate briefly how we apply BNNs to PQC, when predicting a quality characteristic \hat{y} from process data \hat{x} .

The (posterior) predictive distribution of the unknown value \hat{y} for the test item \hat{x} is given by $P(\hat{y}|\hat{x}) = \mathbb{E}_{P(\mathbf{w}|\mathcal{D})} [P(\hat{y}|\hat{x}, \mathbf{w})]$. The unknown distribution $P(\mathbf{w}|\mathcal{D})$ can be rewritten using Bayes' theorem:

$$P(\mathbf{w}|\mathcal{D}) = \frac{P(\mathcal{D}|\mathbf{w})P(\mathbf{w})}{P(\mathcal{D})}, \quad (6)$$

where $P(\mathbf{w})$ is the prior on the weights, $P(\mathcal{D})$ is a normalizing constant, and $P(\mathcal{D}|\mathbf{w})$ is the likelihood of observation. To enable PQC in industrial settings, the predicted distribution $P(\hat{y}|\hat{x})$ requires a small variance σ^2 . However, this is not a specific goal of training a BNN since this method aims to approximate the distribution based on the given data. Hence the ambitions of quality engineers and mathematicians are not necessarily aligned.

There is not yet a consensus on how to quantify the quality of uncertainty quantification. Standard measures for a good fit of the posterior are the average marginal-log-likelihood, the prediction interval coverage probability, or the mean prediction interval width. However, Yao et al. show that these measures depend on the inference method used to determine the posterior distribution; we refer to [46] for a discussion of this matter.

Interim Conclusion As detailed above, ML algorithms are successfully applied to PQC applications. In special use cases, we even see deployments in industrial applications even though uncertainties are not considered. Further, we established that UQ essential part for PQC and almost all other ML applications outside of laboratories.

To accomplish the overall goal to certify PQC methods as an inspections process, the application of UQ on PQC methods is imminent. We focus our upcoming research on BNN, as we see them as the most comprehensive and expressive method.

4 Application of Bayesian Neural Networks to the Prediction of Quality Characteristics

We apply a BNN to an injection molding process of a thin-walled thermoplastic part. In expert interviews, 14 process parameters (e.g., tool temperature, cycle time, pressure) were identified, each of which is recorded with one sensor. Hence, the machine provides $m = 14$ sensors for process data. We focus on $n = 1$ quality characteristic, i.e., a length of the exemplary part with a nominal value of 72.6 mm. The database \mathcal{D} was generated using a full-factorial design of experiments (DoE), where machine settings are explicitly varied, with $k = 600$ experiments. The measurements of the quality characteristic were performed on a coordinate-measuring machine, whose suitability was proven by a Gage R&R Study (MSA) in advance [29].

The data quality is excellent, as it was manually verified during the recording and before model training. All sensors and the quality characteristic are scaled to the interval $[0, 1]$ to facilitate efficient model training. The original scaling is used for the interpretation in the industrial context in Sect. 4.1.

We use a feed-forward neural network with two hidden layers and leaky ReLU activation functions. The first hidden layer has four nodes, while the second hidden layer has two nodes. The second layer's output is used to parametrize a normal distribution $\mathcal{N}(\mu, \sigma)$: the first node is interpreted as the mean μ , while the second node is understood as the variance σ .

Comparably to [7], we use a prior $P(\mathbf{w})$ on our weights \mathbf{w} and fit a posterior $P(\mathbf{w}|\mathcal{D})$. A prior is placed on the weights $P_t(\mathbf{w}) = \prod_j \mathcal{N}(\mathbf{w}_j|t_j, \sigma_p)$ where $\mathcal{N}(x|\mu_p, \sigma_p)$ is the Gaussian density evaluated at x with mean μ_p and variance σ_p . The prior is learnable as the means t_j are fitted during training, while $\sigma_p = 1$ is fixed. We use a Gaussian variational posterior with trainable mean and variance.

The network is trained for 1250 epochs with a learning rate of 0.001 using the Adam optimizer. The other hyperparameters of the optimizer are the default values.¹ For the loss L we use the sum of the Kullback–Leibler divergence from both hidden layers and add the negative log-likelihood:

$$L = KL_1 + KL_2 + \mathbb{E}_{q_1(\mathbf{w}_1|\theta_1), q_2(\mathbf{w}_2|\theta_2)} [\log P(\mathbf{w}|\mathcal{D})]. \quad (7)$$

Here $KL_i = KL[q_i(\mathbf{w}_i|\theta_i)||P(\mathbf{w}_i)]$ where $i = 1, 2$ indexes the hidden layers and θ_i are the parameters of a distribution on the weights. We keep the notation according to [7] and refer the interested reader for details. The loss L over the 1250 epochs is given in Fig. 2. After plateauing for about 1000 epochs, a final drop occurs over another 200 epochs before optimal performance is reached.

We train the BNN on 540 data points ($\approx 90\%$) and randomly select 60 ($\approx 10\%$) points for the evaluation. We sample the trained BNN 5000 times for each evaluation

¹ https://www.tensorflow.org/api_docs/python/tf/keras/optimizers/Adam.

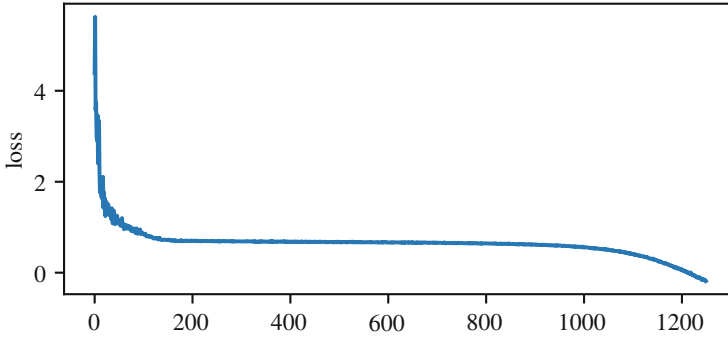


Fig. 2 Loss L during the training with 1250 epochs

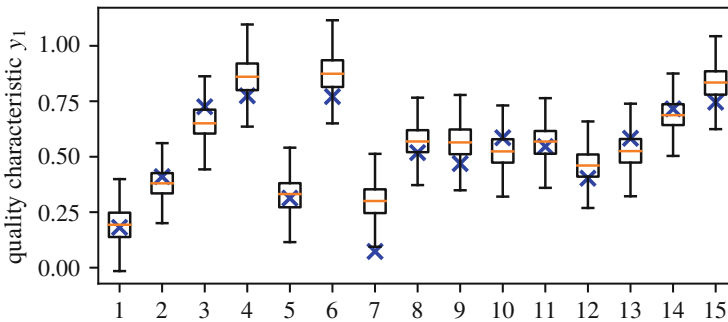


Fig. 3 Quality characteristic y_1 (blue) with a box plot of the prediction $H_{\mathbf{w}}(x)$ based on 5000 model evaluations for 15 samples from the test data set

point to generate as many pairs (μ_i, σ_i) for the parametrized normal distribution. Figure 3 depicts the means μ_i in a box plot for the first 15 evaluation points and give the results for the first 10 as tabular data in Table 2. The actual quality characteristics y_1 are given in blue in the box plot for comparison. The mean absolute error (MAE) between the mean of means $\frac{1}{5000} \sum_{i=0}^{5000} \mu_i$ and the actual value y_1 is ≈ 0.1814 . In relation to the size of the data set, this is a reasonably low MAE. In Fig. 3 only sample $i = 7$ is an outlier regarding the mean of means. A more extensive data set would allow more rigorous training of the BNN and yield a better MAE. We provide code and the scaled data set in our GitHub repository.²

² <https://github.com/predictive-quality/bnn-example>.

Table 2 Quality characteristic y_1 with the mean prediction $\mathbb{E}[H_{\mathbf{w}}(x)]$ and the variance of the prediction $\mathbb{V}[H_{\mathbf{w}}(x)]$ after the training

y_1	$\mathbb{E}[H_{\mathbf{w}}(x)]$	$\mathbb{V}[H_{\mathbf{w}}(x)]$
0.30239546	0.2830744	0.00976317
0.48470376	0.43315677	0.00700909
0.30063383	0.27861683	0.00965479
0.15363564	0.1343955	0.00571805
0.49191045	0.5456889	0.00547123
0.35253981	0.356259	0.00638528
0.30983144	0.296187	0.00978199
0.32386012	0.26562449	0.00686061
0.51326475	0.51090966	0.00551332
0.35963779	0.36319379	0.0046631

Table 3 Quality characteristic y_1 with the mean prediction $\mathbb{E}[H_{\theta}(x)]$ and the variance of the prediction $\mathbb{V}[H_{\theta}(x)]$ after transformation to the original scale

y_1	$\mathbb{E}[H_{\theta}(x)]$	$\mathbb{V}[H_{\theta}(x)]$
72.15644055	72.1800606	0.01459118
72.33991666	72.402933	0.01047518
72.15099116	72.177907	0.0144292
71.97468018	71.9982013	0.0085457
72.47748751	72.4117432	0.00817683
72.24590892	72.2413622	0.00954288
72.17247074	72.1891511	0.01461932
72.13510798	72.2063012	0.01025328
72.43496978	72.4378489	0.00823973
72.25438673	72.2500395	0.00696906

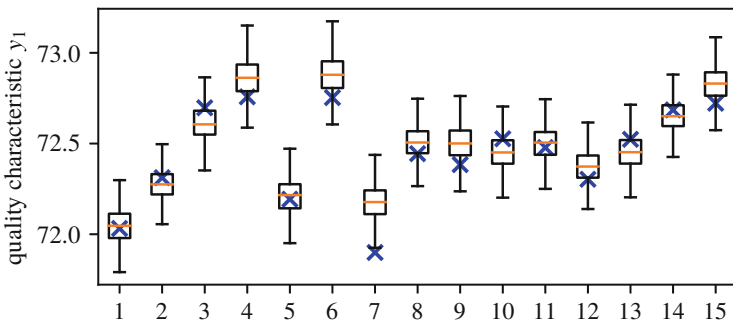


Fig. 4 Quality characteristic y_1 (blue) in its original scale with a box plot of the prediction $H_{\mathbf{w}}(x)$ based on 5000 model evaluations for 15 samples from the test data set

4.1 Interpretation in the Industrial Context

For the industrial practitioner, the raw results of the BNN need further interpretation. Primarily, we have to restore the original scaling to evaluate the PQC in context. In Table 3 and Fig. 4 the predicted values are restored to their original scaling. It

is notable how the variance decreases after the rescaling. This does not indicate a better model performance but is rather due to the dependency of variance on the mean. Similarly, the MAE decreases to ≈ 0.0641 .

To prove the suitability for this virtual inspections process, the *golden rule of metrology* according to which the ratio $\frac{U}{T}$ of the uncertainty of measurements U to tolerance T shall not be greater to one-tenth to one-fifth [39]. For our example, we can interpret the 2σ -interval γ of $H_{\mathbf{w}}(x)$ as the uncertainty of measurement. Then with $\mathbb{V}[H_{\mathbf{w}}(x)] < 0.0167$:

$$\gamma = 2\sqrt{\mathbb{V}[H_{\mathbf{w}}(x)]} \leq 0.258. \quad (8)$$

Given $T = 0.6$ and choosing $U = \lceil \gamma \rceil$, we derive

$$\frac{U}{T} \leq \frac{0.258}{0.6} = 0.43 \stackrel{!}{\leq} 0.2. \quad (9)$$

Thus, based on this conservative estimate of the uncertainty of measurement, this BNN is not suitable as an inspection process. However, the following aspects need further consideration:

- Using a more advanced inference method (e.g., Hamiltonian Monte Carlo) can better approximate the posterior and generate more favorable results regarding the suitability.
- As the database was generated by a DOE, the process variation is deliberately high. This is in stark contrast to a real production environment, where the variation is usually low, and process capability is ensured.
- The size of the database is relatively small compared to the number of trainable parameters (≈ 210) in the BNN.
- The hyperparameters have a significant influence on the performance of the BNN. Deliberate, application-specific manual tuning or the use of AutoML-methods could guarantee proof of suitability.

Overall, we are certain that BNN are a well-suited method for PQC, but we openly acknowledge that more research is necessary before adopting industrial applications.

Furthermore, for a formal evaluation of the suitability, the measurement uncertainty must be determined by an approved procedure as the GUM or the VDA 5 (see [39] for details). However, none of these procedures considers algorithms based on process data. Many aspects from physical inspection procedures are transferable to PQC, yet some error sources (e.g., numerical concerns) are not addressed. As the adoption and development of PQC methods progress, the process to determine suitability will be extended as well.

5 Concluding Remarks

We identified the prediction of quality characteristics as the fundamental foundation of every predictive quality method. To give a framework for future research, we provided a formal definition of *prediction of quality characteristics*. Further, we established PQC as a virtual inspection process, which can complement and/or reduce costly physical inspections. For every inspection process, a proof of suitability is necessary, which requires the determination of the measurement uncertainty of the underlying method. Hence we added a Bayesian perspective to our definition to PQC, to consider model- and data-inherent uncertainties.

Based on our literature review, we reason that existing machine learning methods, as BNN, can provide an adequate uncertainty estimation. The uncertainty estimates are a decisive keystone to establish PQC as a virtual inspection process and permit proof of suitability. As a showcase, we applied a BNN to an injection molding process and give several hints on how to improve the uncertainty estimate for future applications. To facilitate adoption in industry, we advocate for a revision of standards as the VDA5 or the ISO 22514-7 to accommodate for virtual inspection processes.

Acknowledgments Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy—EXC-2023 Internet of Production—390621612.

The data set was created in cooperation with Festo Polymer GmbH.

References

1. Arangio, S., Beck, J.L.: Bayesian neural networks for bridge integrity assessment. *Struct. Control. Health Monit.* **19**(1), 3–21 (2012). <https://doi.org/10.1002/stc.420>
2. Auld, T., Moore, A.W., Gull, S.F.: Bayesian neural networks for internet traffic classification. *IEEE Trans. Neural Netw.* **18**(1), 223–239 (2007). <https://doi.org/10.1109/TNN.2006.883010>
3. Baturynska, I., Semeniuta, O., Wang, K.: Application of machine learning methods to improve dimensional accuracy in additive manufacturing. In: Wang, K., Wang, Y., Strandhagen, J.O., Yu, T. (eds.) *Advanced Manufacturing and Automation VIII*, Lecture Notes in Electrical Engineering, pp. 245–252. Springer, Singapore (2019). https://doi.org/10.1007/978-981-13-2375-1_31
4. Bellman, R.: *Dynamic Programming*. Princeton University Press, Princeton, NJ (1984)
5. Bergs, Thomas: *Internet of Production—Turning Data into Value* (2020). <https://doi.org/10.24406/IPT-N-589615>
6. Biegler, L.T. (ed.): *Large-scale inverse problems and quantification of uncertainty*. In: Wiley series in computational statistics. Wiley, Chichester, West Sussex (2011)
7. Blundell, C., Cornebise, J., Kavukcuoglu, K., Wierstra, D.: *Weight Uncertainty in Neural Networks* (2015). ArXiv: 1505.05424
8. Borg, M., Englund, C., Wnuk, K., Duran, B., Levandowski, C., Gao, S., Tan, Y., Kaijser, H., Lönn, H., Törnqvist, J.: *Safely Entering the Deep: A Review of Verification and Validation for Machine Learning and a Challenge Elicitation in the Automotive Industry* (2018). ArXiv: 1812.05389

9. Caldeira, J., Nord, B.: Deeply uncertain: comparing methods of uncertainty quantification in deep learning algorithms. *Machine Learning: Science and Technology* **2**(1), 015002 (2020). <https://doi.org/10.1088/2632-2153/aba6f3>. ArXiv: 2004.10710
10. Carlini, N., Wagner, D.: Towards Evaluating the Robustness of Neural Networks. arXiv:1608.04644 [cs] (2017). ArXiv: 1608.04644
11. Cattaneo, L., Fumagalli, L., Macchi, M., Negri, E.: Clarifying data analytics concepts for industrial engineering. *IFAC-PapersOnLine* **51**(11), 820–825 (2018). <https://doi.org/10.1016/j.ifacol.2018.08.440>
12. Chen, W.C., Lee, A.H.I., Deng, W.J., Liu, K.Y.: The implementation of neural network for semiconductor PECVD process. *Expert Systems with Applications* **32**(4), 1148–1153 (2007). <https://doi.org/10.1016/j.eswa.2006.02.013>
13. Döbel, I., Leis, M., Molina Vogelsang, M., Welz, J., Neustroev, D., Petzka, H., Riemer, A., Püping, S., Voss, A., Wegele, M.: *Maschinelles Lernen. Eine Analyse zu Kompetenzen, Forschung und Anwendung*. Study, Fraunhofer-Gesellschaft, München (2018)
14. Gal, Y., Hron, J., Kendall, A.: Concrete dropout. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*, vol. 30, pp. 3581–3590. Curran Associates, Inc., Red Hook (2017)
15. Ge, Z., Song, Z., Ding, S.X., Huang, B.: Data mining and analytics in the process industry: the role of machine learning. *IEEE Access* **5**, 20590–20616 (2017). <https://doi.org/10.1109/ACCESS.2017.2756872>
16. GQW-Jahrestagung: Qualitätsmanagement 4.0—Status quo! Quo vadis? Bericht zur GQW-Jahrestagung 2016 in Kassel. No. Band 6 in Kasseler Schriftenreihe Qualitätsmanagement. Kassel University Press, Kassel (2016)
17. Graves, A.: Practical variational inference for neural networks. In: Shawe-Taylor, J., Zemel, R.S., Bartlett, P.L., Pereira, F., Weinberger, K.Q. (eds.) *Advances in neural information processing systems* 24, pp. 2348–2356. Curran Associates, Inc., Red Hook (2011)
18. Group, A.I.A.: *Measurement systems analysis: [MSA] ; reference manual*, 4th edn. Automotive Industry Action Group, Southfield, Mich (2010)
19. Hinton, G.E., Srivastava, N., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.R.: Improving neural networks by preventing co-adaptation of feature detectors (2012). ArXiv: 1207.0580
20. Hüllermeier, E., Waegeman, W.: *Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods* (2020). ArXiv: 1910.09457
21. Khan, M.S., Coulibaly, P.: Bayesian neural network for rainfall-runoff modeling. *Water Resour. Res.* **42**(7) (2006). <https://doi.org/10.1029/2005WR003971>
22. Kononenko, I.: Bayesian neural networks. *Biol. Cybern.* **61**(5), 361–370 (1989). <https://doi.org/10.1007/BF00200801>
23. Krauß, J., Dorißen, J., Mende, H., Frye, M., Schmitt, R.H.: Machine learning and artificial intelligence in production: application areas and publicly available data sets. In: Wulfsberg, J.P., Hintze, W., Behrens, B.A. (eds.) *Production at the leading edge of technology*, pp. 493–501. Springer, Berlin (2019). https://doi.org/10.1007/978-3-662-60417-5_49
24. Lakshminarayanan, B., Pritzel, A., Blundell, C.: Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles (2017). ArXiv: 1612.01474
25. Lansner, A., Ekeberg, O.: A one-layer feedback artificial neural network with a Bayesian learning rule. *Int. J. Neural Syst.* **01**(01), 77–87 (1989). <https://doi.org/10.1142/S0129065789000499>
26. Lin, N.: *Applied Business Analytics: Integrating Business Process, Big Data, and Advanced Analytics*. Pearson Education, Upper Saddle River (2014)
27. Mangal, R., Nori, A.V., Orso, A.: Robustness of Neural Networks: A Probabilistic and Practical Approach (2019). ArXiv: 1902.05983
28. Mueller, T., Huber, M., Schmitt, R.: Modelling complex measurement processes for measurement uncertainty determination. *International Journal of Quality and Reliability Management* **37**(3), 494–516 (2020). <https://doi.org/10.1108/IJQRM-07-2019-0232>

29. Mueller, T., Kiesel, R., Schmitt, R.H.: Automated and predictive risk assessment in modern manufacturing based on machine learning. In: *Advances in Production Research*, pp. 91–100. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-03451-1_10
30. Ogorodnyk, O., Lyngstad, O.V., Larsen, M., Wang, K., Martinsen, K.: Application of machine learning methods for prediction of parts quality in thermoplastics injection molding. In: Wang, K., Wang, Y., Strandhagen, J.O., Yu, T. (eds.) *Advanced Manufacturing and Automation VIII*, Lecture Notes in Electrical Engineering, pp. 237–244. Springer, Singapore (2019). https://doi.org/10.1007/978-981-13-2375-1_30
31. Otte, C.: Safe and interpretable machine learning: a methodological review. In: Moewes, C., Nürnberger, A. (eds.) *Computational Intelligence in Intelligent Data Analysis*, vol. 445, pp. 111–122. Springer, Berlin (2013). https://doi.org/10.1007/978-3-642-32378-2_8
32. Patel, D., Hazan, H., Saunders, D.J., Siegelmann, H.T., Kozma, R.: Improved robustness of reinforcement learning policies upon conversion to spiking neuronal network platforms applied to Atari Breakout game. *Neural Netw.* **120**, 108–115 (2019). <https://doi.org/10.1016/j.neunet.2019.08.009>
33. Pei, K., Cao, Y., Yang, J., Jana, S.: Towards Practical Verification of Machine Learning: The Case of Computer Vision Systems (2017). ArXiv: 1712.01785
34. Schmitt, J., Böning, J., Borggräfe, T., Beiting, G., Deuse, J.: Predictive model-based quality inspection using Machine Learning and Edge Cloud Computing. *Adv. Eng. Inform.* **45**, 101101 (2020). <https://doi.org/10.1016/j.aei.2020.101101>
35. Schmitt, R.H., Ellerich, M., Schlegel, P., Ngo, Q.H., Emonts, D., Montavon, B., Buschmann, D., Lauther, R.: Datenbasiertes Qualitätsmanagement im Internet of Production. In: Frenz, W. (ed.) *Handbuch Industrie 4.0: Recht, Technik, Gesellschaft*, pp. 489–516. Springer, Berlin (2020). https://doi.org/10.1007/978-3-662-58474-3_25
36. Schuh, G., Riesener, M., Prote, J.P., Dölle, C., Molitor, M., Schloesser, S., Liu, Y., Tittel, J.: Industrie 4.0: Agile Entwicklung und Produktion im Internet of Production. In: Frenz, W. (ed.) *Handbuch Industrie 4.0: Recht, Technik, Gesellschaft*, pp. 467–488. Springer, Berlin (2020). https://doi.org/10.1007/978-3-662-58474-3_24
37. Sullivan, T.: Introduction to uncertainty quantification. *Texts in Applied Mathematics*, vol. 63. Springer, Cham (2015). <https://doi.org/10.1007/978-3-319-23395-6>
38. Utama, R., Piekarewicz, J.: Refining mass formulas for astrophysical applications: A Bayesian neural network approach. *Phys. Rev. C* **96**(4), 044308 (2017). <https://doi.org/10.1103/PhysRevC.96.044308>. Publisher: American Physical Society
39. e. V., V.D.I.: VDI/VDE-Richtlinie 2600 Blatt 1: 2013—10 Prüfprozessmanagement—Identifizierung, Klassifizierung und Eignungsnachweise von Prüfprozessen (VDI/VDE-Guideline 2600 Part 1: 2013—10 Inspection process management—Identification, classification and proof of suitability for inspection processes). (2013)
40. VDA (ed.): VDA 5 -Prüfprozesseignung, Eignung von Messsystemen, Mess- und Prüfprozessen, Erweiterte Messunsicherheit, Konformitätsbewertung, 2 edn., vol. 5 (2011)
41. Verlag, B.: Geometrical product specifications (GPS)—Inspection by measurement of workpieces and measuring equipment—Part 1: Decision rules for verifying conformity or nonconformity with specifications (ISO 14253-1:2017); German version EN ISO 14253-1:2017. Tech. rep., Beuth Verlag GmbH (2017). <https://doi.org/10.31030/2693140>
42. Verlag, B.: DIN ISO 22514-7:2020-06, Statistische Verfahren im Prozessmanagement_ -Fähigkeit und Leistung_ - Teil 7: Fähigkeit von Messprozessen (ISO/DIS 22514-7:2020); Text Deutsch und Englisch. Tech. rep., Beuth Verlag GmbH (2020). <https://doi.org/10.31030/3160215>
43. Wojtkiewicz, S., Eldred, M., Field Jr., R., Urbina, A., Red-Horse, J.: Uncertainty quantification in large computational engineering models. In: 19th AIAA Applied Aerodynamics Conference. American Institute of Aeronautics and Astronautics, Anaheim (2001). <https://doi.org/10.2514/6.2001-1455>
44. Xiang, W., Musau, P., Wild, A.A., Lopez, D.M., Hamilton, N., Yang, X., Rosenfeld, J., Johnson, T.T.: Verification for Machine Learning, Autonomy, and Neural Networks Survey (2018). ArXiv: 1810.01989

45. Xie, Y., Lord, D., Zhang, Y.: Predicting motor vehicle collisions using Bayesian neural network models: An empirical analysis. *Accid. Anal. Prev.* **39**(5), 922–933 (2007). <https://doi.org/10.1016/j.aap.2006.12.014>
46. Yao, J., Pan, W., Ghosh, S., Doshi-Velez, F.: Quality of Uncertainty Quantification for Bayesian Neural Network Inference (2019). ArXiv: 1906.09686