

Estimating Sufficient Amount of Training Data for Data-Driven Fault Diagnosis: Learning Curve Extrapolation and Entropy-Based Stopping Criteria

Moritz Neu^{1§}, Marcel Braig^{1§}, Fabian Mauthe^{1§}, and Peter Zeiler¹

¹ Institute for Technical Reliability and Prognostics IZP, Esslingen University of Applied Sciences, Göppingen, 73037, Germany
neu.moritz@outlook.de, marcel.braig@hs-esslingen.de, fabian.mauthe@hs-esslingen.de, peter.zeiler@hs-esslingen.de

ABSTRACT

Data-driven approaches for diagnosing and predicting the health of engineering systems require sufficient training data to achieve reliable performance. However, despite its high practical relevance, the question of how much data are actually required has not yet been systematically investigated. This paper presents and evaluates two complementary approaches to answering this question. Approach I fits parametric functions to the initial segments of learning curves and extrapolates them to estimate the amount of training data required to achieve a target classification accuracy. Approach II introduces a training-free, entropy-based stopping criterion that uses the gradient of normalized entropy (GNE). The information gain from the data collected over time is monitored to determine whether further data acquisition is necessary. Both approaches are evaluated using a dataset for fault classification in rolling bearings, employing a multi-layer perceptron classifier and randomly drawn, in-distribution test sets. The results show that learning curve extrapolation enables predictions of the required amount of data and that GNE can serve as an effective online stopping rule during data acquisition due to the strong correlation between GNE and classification accuracy. The two approaches complement each other: entropy analysis can monitor the progress of data acquisition in real time without model training, while learning curve extrapolation provides quantitative estimates of the remaining data requirements for specific accuracy targets.

1. INTRODUCTION

Data-driven approaches are widely used and empirically validated for diagnosing and predicting the health of engineering systems. However, these approaches require sufficient train-

ing data to cover the relevant state space and labels (Fink et al., 2026; Zio, 2022; Arias Chao, Kulkarni, Goebel, & Fink, 2022). Consequently, the question arises: What exactly constitutes *sufficient*? As demonstrated in Section 2, this question has not been systematically examined in the scientific literature on diagnosing and predicting the health of engineering systems. However, it is crucial for industrial practice to assess whether the current dataset is sufficient and, if not, how much additional data have to be generated. This need is especially acute for fault diagnosis and health prediction, where data originate from degradation runs that deliberately damage systems, making data generation highly resource-intensive (Guo, Dai, & Zhang, 2025; Li et al., 2024; Hagmeyer, Mauthe, & Zeiler, 2021). Therefore, it is crucial to determine how many samples are required to achieve the performance targets set for data-driven diagnostics and prognostics.

For this purpose, this paper investigates methods that aim to answer precisely the aforementioned question. Specifically, we ask whether—given the currently available data—the desired fault classification accuracy can be achieved and, if not, how many additional samples are needed. The study addresses an essential issue in the field of diagnostics and prognostics—one that has been only sporadically investigated across application domains and not at all for diagnostics and prognostics, to the best of the authors’ knowledge. We focus on fault classification for rolling bearings as the diagnostic task. Our approaches comprise (i) modeling learning curves to predict accuracy as a function of dataset size and (ii) entropy analysis to assess the representativeness of the state space.

Before we proceed, the concept of *sufficiency* must be clarified, as it can refer to two different issues. *In-distribution sufficiency* addresses the question of how many samples are required before the model’s performance reaches its saturation point on data drawn from the same distribution as the available data. *Operational sufficiency* addresses the question of whether the training data cover the relevant operational state

[§]These authors contributed equally.
Moritz Neu et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.
<https://doi.org/10.36001/IJPHM.2026.v17i2.4834>

space, so that the required performance is achieved even under conditions not represented in the available data—such as different loads, rotational speeds, bearing types, machines, or failure severities. Both questions are relevant in practice and are not synonymous: A dataset can be sufficient in the first sense and insufficient in the second, since the saturation of a learning curve merely indicates that no further information can be gleaned from the sample distribution. The approaches presented in this paper are formulated independently of this distinction, whereas the evaluation conducted in Section 4 focuses on in-distribution sufficiency. Where data requirements are quantified in the following, they therefore refer to this context.

Therefore, this paper contains the following contributions:

1. Two approaches are presented for determining whether sufficient data are available in machine learning (ML) and for estimating how much additional data are needed. As the literature presented in this paper shows, ML research deals with these issues only very sporadically.
2. The approaches are comprehensively evaluated and discussed using the example of fault diagnosis—specifically fault classification—and based on a wide variety of criteria. The authors are not aware of any other work in the field of fault diagnosis that addresses this issue.
3. In the learning curve analysis approach, five different fit functions are analyzed for extrapolating the learning curves and their suitability is compared.
4. A novel approach is presented that uses entropy analysis to determine, without any training, whether sufficient data are already available for training ML models.

The structure of this paper is as follows: Section 2 introduces related work on predicting data requirements for ML methods. Section 3 describes the approaches used in this paper. The evaluation use case and results are presented and discussed in Section 4. Finally, Section 5 provides a conclusion and outlook.

2. RELATED WORK

Research on estimating the necessary amount of training data n_{req} to achieve a required performance V_{req} in ML methods is still in its infancy. Only a handful of thematically relevant studies could be identified. Based on these studies, the approaches can be divided into three categories that address complementary questions: (1) learning curve extrapolation for direct prediction of n_{req} or prediction of the course of the learning curve $V(n)$ over the amount of training data n , (2) distribution or stability criteria as stop rules when n_{req} is reached without explicit extrapolation, and (3) sample-value estimation to identify high-value samples or regions (no direct estimate of n_{req}).

List of Symbols

Symbol	Description
$\text{Acc}(p)$	$V(p)$ with classification accuracy as performance metric
$\text{GNE}(p)$	Gradient normalized entropy for p
$H(n), H(p)$	Shannon entropy for n or p
i_{sat}	Index of entropy saturation onset
L	Number of bins of the empirical probability mass function (PMF)
$M = \mathcal{D} $	Number of samples in $\mathcal{D}$
n	Number of samples in $\mathcal{D}_{\text{train}}^{(n)}$
$N = \mathcal{D}_{\text{train}} $	Number of samples in $\mathcal{D}_{\text{train}}$
$n_{\text{req}}, V_{\text{req}}$	Necessary training data samples n_{req} for a required performance V_{req}
p	Relative training subset size of $\mathcal{D}_{\text{train}}^{(p)}$
$\hat{p}_{\text{Acc}_{\text{req}}}$	Predicted required p for a classification accuracy of Acc_{req}
p_{fit}	p of the initial fitting segment used to fit $\hat{V}(p)$
$p_{i_{\text{sat}}}$	p with entropy saturation
$r \in \{1, \dots, k\}$	Repetition index—for statistical significance the evaluations are repeated k -times
V	Performance metric of a machine learning model (various can be used)
$V(n), V(p)$	Learning curve over n or p
$\hat{V}(n), \hat{V}(p)$	Learning curve estimate over n or p
$\mathbf{x} \in \mathcal{X}$	Feature vector
$\mathcal{D}$	Dataset
$\mathcal{D}_{\text{test}} = \mathcal{D} \setminus \mathcal{D}_{\text{train}}$	Test set
$\mathcal{D}_{\text{train}} \subset \mathcal{D}$	Full training set
$\mathcal{D}_{\text{train}}^{(\cdot)}$	Training subset
$\mathcal{I}_{\text{fit}}^{(p_{\text{fit}})}, \mathcal{I}_{\text{pred}}^{(p_{\text{fit}})}$	Fitting index set, evaluation index set
$\mathcal{P}_{\text{fit}}^{(p_{\text{fit}})}, \mathcal{P}_{\text{pred}}^{(p_{\text{fit}})}$	Fitting set, evaluation set
$\mathcal{X} \subseteq \mathbb{R}^d$	Feature space with ambient dimension $d \in \mathbb{N}$
$\mathcal{Y} = \{1, \dots, C\}$	Label space of a C -class classification problem with mutually exclusive classes ($C \in \mathbb{N}$)
$\theta_{p_{\text{fit}}}$	Parameters of the learning curve fit function

(1) Learning curve extrapolation: Learning curves $V(n)$ refer to the test performance of an ML model over the available training dataset size n , whereby various performance metrics V can be used. The goal of learning curve extrapolation is to fit the actual learning curve $V(n)$ with a monotonically increasing or decreasing function from given learning curve points, resulting in a learning curve estimate $\hat{V}(n)$ that can be used for extrapolation. The functions used are typically saturating; non-saturating functions often exhibit quasi-saturating behavior (marginal gain $\rightarrow 0$) in the practical n range. $\hat{V}(n)$ can be used to estimate the required number of

samples n_{req} for a target value V_{req} (e.g., 80 % accuracy). Crowther and Cox (2006) examine a differential saturation model with exponential approximation to an asymptotic performance V_{max} :

$$\frac{d\hat{V}(n)}{dn} = k \left(V_{\text{max}} - \hat{V}(n) \right), \quad (1)$$

with the solution

$$\hat{V}(n) = V_{\text{max}} - (V_{\text{max}} - V(0)) \cdot e^{-kn}. \quad (2)$$

It provides reliable fits for large datasets or simple classification tasks (a simulation dataset and the real-world datasets Abalone, Wisconsin Breast Cancer, and Dermatology), but it is difficult to parameterize robustly for small and difficult-to-classify datasets (Weed Seeds) (Crowther & Cox, 2006). Figueroa, Zeng-Treitler, Kandula, and Ngo (2012) show for classification of two clinical datasets and a synthetic waveform dataset that inverse power law models

$$\hat{V}(n) = (1 - a) - bn^c \quad (\text{with } c < 0) \quad (3)$$

approximate learning curves well. The selected weighted non-linear regression improves the prediction compared to unweighted curve-fitting and allows confidence intervals of the extrapolated performance metric. Mahmood, Lucas, Acuna, et al. (2022) evaluate several concave, monotonically increasing functions (power law, logarithm, algebraic root, arctan) for computer vision tasks, such as classification, detection, and segmentation. It is shown that even small fitting errors in $\hat{V}(n)$ can lead to large errors in the n_{req} estimate. Especially when fitting to initial learning curves with small n , the extrapolations diverge. Fitting functions frequently show systematic optimistic or pessimistic bias; therefore, the authors recommend multiple estimation iterations with a correction factor τ , and using an ensemble of optimistic and pessimistic fits to bound the predicted n_{req} . Arctan acts most pessimistically across classification tasks, often yielding the largest safety margin. In many cases, an ensemble of optimistic and pessimistic functions provides practical bounds. Whereas Mahmood, Lucas, Acuna, et al. (2022) extrapolate the learning curve $V(n)$ to produce a point estimate of n_{req} , Mahmood, Lucas, Alvarez, Fidler, and Law (2022) repeatedly fit $V(n)$ to approximate a distribution over the minimum required data. The final choice of n_{req} is then posed as a cost-risk optimization—minimizing expected acquisition cost with a penalty controlling failure risk.

(2) Distribution or stability criteria (stopping rules): Several approaches examine whether a given data distribution is already sufficiently representative or stable—as a stopping rule for further data acquisition. Thus, n_{req} is determined by distribution coverage without explicit extrapolation of $V(n)$. Approaches can be purely data-based without model training or model-based, where a (preliminarily) trained model serves

as a stability or uncertainty sensor (Estepa, Diaz-Verdejo, Estepa, & Madinabeitia, 2020; Jo, Yun, & Choi, 2022). Data-based approaches pursue the idea that data acquisition can be terminated when the core characteristics of the distribution, e.g., vocabulary or entropy, no longer change. Estepa et al. (2020) develop four distribution coverage indicators for HTTP anomaly detection with unigram dictionaries: vocabulary size i_1 , χ^2 -based distribution similarity i_2 , entropy i_3 , and core word count i_4 . These indicators are smoothed using exponentially weighted moving averages. Stability is defined by the coefficient of variation of the smoothed time series in a defined window relative to a threshold value and a minimum duration. A stop occurs when (i_1, i_2) or (i_3, i_4) are stable. Empirically, this allows for significant reductions in unnecessary data acquisition without resulting in performance decreases. Jo et al. (2022) investigate how much passive learning is useful before starting active learning. They introduce the metric Entropy of Sample-Uncertainty (EoU), which quantifies how uniformly a model’s predictive uncertainty is distributed across a dataset. An untrained model is uncertain about nearly all samples (high, near-uniform EoU), whereas a better-trained model concentrates uncertainty on a smaller subset (lower EoU). A threshold τ is set such that, when $\text{EoU} \leq \tau$, the procedure switches from passive to active learning. The method requires a preliminarily trained model to obtain predictive uncertainties, provides a stop rule without extrapolating $V(n)$, and does not directly estimate n_{req} . The approach is evaluated on the MNIST, CINIC-10, and CIFAR-10/100 datasets.

(3) Sample-value estimation: This category aims to assess the marginal value of samples or regions for model performance, rather than estimating the required data size n_{req} . Bachoc, Gamboa, Halford, Loubes, and Risser (2023) construct targeted deformations (referred to as “stresses”) of the empirical test distribution Q_n to obtain a stressed distribution $\hat{Q}_n$ that perturbs means, variances, or correlations. Closeness to Q_n is enforced by minimizing the Kullback–Leibler divergence $D_{\text{KL}}(\hat{Q}_n \| Q_n)$. Instead of synthesizing new points, the existing observations are reweighted, thereby preventing the occurrence of out-of-distribution artifacts. The method identifies where the model is sensitive—and thus where additional data would be most effective—but does not provide a direct n_{req} estimate. Xie, Ding, Tomizuka, and Zhan (2023) propose a subset selection approach that embeds images with a general-purpose pretrained vision model and uses feature-space similarity to select a small, diverse, and representative subset. The method efficiently surfaces high-value regions of the data distribution, but does not estimate n_{req} . Related questions about sample-value are also studied in active learning (Chen et al., 2019; Gwon, Hwang, Kim, Ok, & Kwak, 2025).

In conclusion, it can be noted that all three categories examine the scale of datasets, but with slightly different ob-

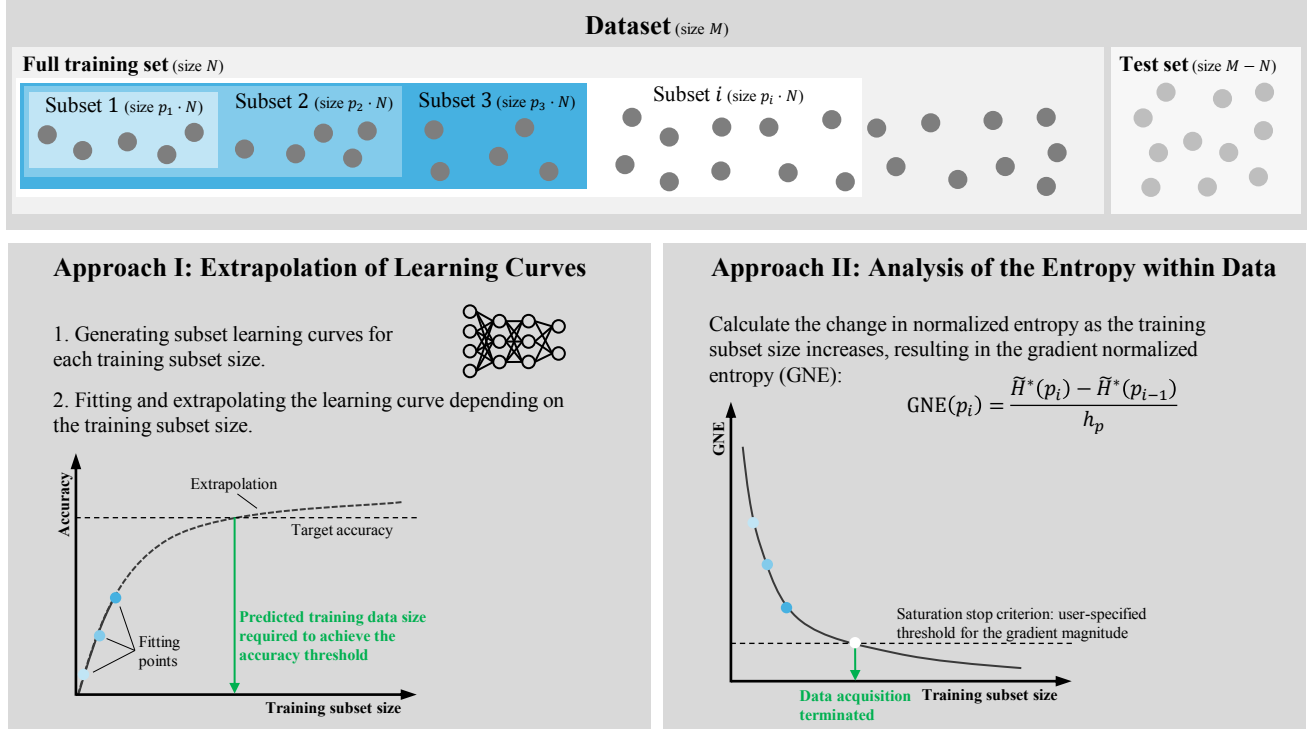


Figure 1. Visualization of the two approaches investigated in this paper. Both approaches are applied to the same training subsets and evaluated with the same test sets, to enable comparability. While Approach I aims to predict the future course of learning curves in order to estimate how much additional data are needed, Approach II evaluates whether sufficient data are available or further data need to be generated.

jectives and approaches that require varying amounts of resources. (1) Learning curve extrapolation provides an explicit estimate of n_{req} for a given V_{req} but requires repeated training or retraining of ML models at multiple n values and is therefore computationally intensive. In addition, there is specification uncertainty due to the choice of the functional form for learning curve approximation, and there are also fit-related estimation errors. (2) Distribution or stability criteria define n_{req} through sufficient coverage or stability of input data or a trained model, but do not provide an extrapolation of $V(n)$. Purely data-based approaches do not require model training. (3) Sample-value estimation methods typically rely on forward passes—plus reweighting in stress tests—and aim to locate high-value regions or samples. They do not directly estimate n_{req} .

The aim of this paper is to determine whether an existing dataset is sufficient for training and, if not, how much additional data are required. Accordingly, we focus on one approach of category (1) and category (2) each. The approach of category (2) evaluates the adequacy of the current data situation, whereas the approach of category (1) estimates how much additional data are needed. Note that (2) does not allow specifying V_{req} ; the methods from (1) remain necessary for this. Category (3) is of secondary importance for the present study—it does not estimate n_{req} .

3. METHODOLOGY

As explained in the previous section, this paper adapts approaches from categories (1) and (2) to assess whether a given dataset is sufficient to train an ML fault classification model. Category (1) is used to predict $V(n)$ via learning curve extrapolation with several parametric fit functions, while an entropy analysis of the available data is employed from category (2). The remainder of this section is organized as follows. Section 3.1 describes the process of obtaining learning curves by repeatedly training classification models on nested training subsets. These learning curves represent the ground truth for the evaluations of the applied approaches explained in Section 3.2 and Section 3.3. Section 3.2 details the used methodology for learning curve extrapolation, and in Section 3.3 the entropy-based analysis of the available data is presented. In Figure 1 both approaches are schematically illustrated.

3.1. Preparation: Generation of Learning Curves

To apply or evaluate the approaches described in Section 3.2 and Section 3.3, learning curves—defined as the true values of a performance metric on a test set as a function of the training sample size—are required. Therefore, in the first step, such learning curves are obtained by repeatedly training an

ML model from scratch while increasing the amount of training data.

Let $\mathcal{X} \subseteq \mathbb{R}^d$ denote the feature space with ambient dimension $d \in \mathbb{N}$. So, each instance is represented by a d -dimensional feature vector $\mathbf{x} \in \mathcal{X}$. Let $\mathcal{Y} = \{1, \dots, C\}$ be the label space of a C -class classification problem with mutually exclusive classes ($C \in \mathbb{N}$). A dataset is a finite set of input-label pairs

$$\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^M \subseteq \mathcal{X} \times \mathcal{Y}, \quad (4)$$

with $M = |\mathcal{D}|$ as the number of samples. We partition $\mathcal{D}$ into a full training set $\mathcal{D}_{\text{train}} \subset \mathcal{D}$ and a test set $\mathcal{D}_{\text{test}} = \mathcal{D} \setminus \mathcal{D}_{\text{train}}$, and write $N = |\mathcal{D}_{\text{train}}|$ for the size of the full training set. To generate learning curves, we construct nested training subsets of $\mathcal{D}_{\text{train}}$:

$$\{\mathcal{D}_{\text{train}}^{(n)}\}_{n \in \mathcal{N}}, \quad \mathcal{D}_{\text{train}}^{(n)} \subseteq \mathcal{D}_{\text{train}}, \quad |\mathcal{D}_{\text{train}}^{(n)}| = n, \quad (5)$$

$$\mathcal{N} \subseteq \{1, \dots, N\}$$

such that $n \leq n' \Rightarrow \mathcal{D}_{\text{train}}^{(n)} \subseteq \mathcal{D}_{\text{train}}^{(n')}$. Concretely, a permutation π of $\{1, \dots, N\}$ is fixed and

$$\mathcal{D}_{\text{train}}^{(n)} = \{(\mathbf{x}_{\pi(i)}, y_{\pi(i)})\}_{i=1}^n, \quad (6)$$

is defined, which ensures nesting and reproducibility.

The performance of the model on $\mathcal{D}_{\text{test}}$ is reported as a function of the size of the training subset, either $V(n)$ for $n \in \mathcal{N}$ or $V(p)$ for the relative size of the training subset $p = n/N \in (0, 1]$, referred to as learning curve. In our use case, namely ML fault diagnosis formulated as multi-class classification, we use classification accuracy as the performance metric as a function of p :

$$V(p) = \text{Acc}(p) = \frac{1}{|\mathcal{D}_{\text{test}}|} \sum_{(\mathbf{x}, y) \in \mathcal{D}_{\text{test}}} \mathbf{1}\{\hat{g}_p(\mathbf{x}) = y\}, \quad (7)$$

where $\hat{g}_p$ is trained on $\mathcal{D}_{\text{train}}^{(n)}$ with $n = \lfloor pN \rfloor$.

To average out stochastic variability, we repeat the split into $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$ k times with independent resampling. For each repetition $r \in \{1, \dots, k\}$ we fix an independent permutation $\pi^{(r)}$ to construct the nested $\mathcal{D}_{\text{train}}^{(r, n)}$. This results in a family of k learning curves $\{\text{Acc}^{(r)}(p)\}_{r=1}^k$, where $\text{Acc}^{(r)}(p)$ is the accuracy obtained on split r when training on a fraction p of $\mathcal{D}_{\text{train}}^{(r)}$ and evaluating on the corresponding $\mathcal{D}_{\text{test}}^{(r)}$. For notational simplicity, we suppress the resampling index r in Section 3.2 and 3.3. All procedures are executed independently for each r .

3.2. Approach I: Extrapolation of Learning Curves

This approach is based on the learning curve extrapolation described in Section 2. Given initial points of a learning curve $\text{Acc}(p)$, a parametric function $\widehat{\text{Acc}}(p; \theta)$ is fitted to the initial

Algorithm 1: Procedure for fitting to the initial learning curves and predicting the further course of the learning curves via extrapolation. (For the purposes of this evaluation, the analysis starts at an i -value of 5.)

Input: k learning curves $\{\text{Acc}^{(r)}(p)\}_{r=1}^k$; $\mathcal{P} = \{p_i\}_{i=1}^m$ with the corresponding $\mathcal{I}_{\text{fit}}$ and $\mathcal{I}_{\text{pred}}$; a fit function from Eqs. (9) to (13)

Output: Fitted curves $\left\{\widehat{\text{Acc}}^{(i, r)}(p)\right\}_{i=5}^{99}\bigg\}_{r=1}^k$

```

1 for all repetitions  $r$  with  $1 \leq r \leq k$  do
2   for all observed fractions  $p_i$  with  $5 \leq i \leq 99$  do
3      $p_{\text{fit}} \leftarrow p_i$ 
4     Determine  $\theta$  by minimizing the weighted
       least-squares loss Eq. (8) using  $\mathcal{I}_{\text{fit}}^{(p_{\text{fit}})}$ 
5      $\widehat{\text{Acc}}^{(i, r)}(p) \leftarrow \widehat{\text{Acc}}_{p_{\text{fit}}}(p, \theta)$ 
6 return  $\left\{\left\{\widehat{\text{Acc}}^{(i, r)}(p)\right\}_{i=5}^{99}\right\}_{r=1}^k = \left\{\widehat{\text{Acc}}^{(r)}(p)\right\}_{r=1}^k$ 

```

segment only, and used to extrapolate to larger p . Therefore, a threshold $p_{\text{fit}} \in (0, 1]$ is fixed. The observed fractions are collected as $\mathcal{P} = \{p_i\}_{i=1}^m \subset (0, 1]$ sorted increasingly, i.e., $p_1 \leq p_2 \leq \dots \leq p_m \leq 1$.

This results in the initial fitting index sets

$$\mathcal{I}_{\text{fit}}^{(p_{\text{fit}})} = \{i \in \{1, \dots, m\} : p_i \leq p_{\text{fit}}\} = \{1, \dots, i_{\text{fit}}\},$$

and the evaluation index sets

$$\mathcal{I}_{\text{pred}}^{(p_{\text{fit}})} = \{i \in \{1, \dots, m\} : p_i > p_{\text{fit}}\} = \{i_{\text{fit}} + 1, \dots, m\},$$

with $i_{\text{fit}} := \max\{i : p_i \leq p_{\text{fit}}\}$.

The corresponding subsets of observed fractions are

$$\mathcal{P}_{\text{fit}}^{(p_{\text{fit}})} = \{p_i : i \in \mathcal{I}_{\text{fit}}^{(p_{\text{fit}})}\} = \{p_1, \dots, p_{i_{\text{fit}}}\},$$

$$\mathcal{P}_{\text{pred}}^{(p_{\text{fit}})} = \{p_i : i \in \mathcal{I}_{\text{pred}}^{(p_{\text{fit}})}\} = \{p_{i_{\text{fit}}+1}, \dots, p_m\}.$$

To ensure that θ is determinable in the weighted least squares problem in Eq. (8), the following must hold: $i_{\text{fit}} \geq q$, where q is the number of free parameters of the chosen fitting function. With $\mathcal{I}_{\text{fit}}^{(p_{\text{fit}})}$, the parameters θ of the fit function $\widehat{\text{Acc}}(p; \theta)$ are estimated by minimizing the weighted least-squares loss:

$$\theta_{p_{\text{fit}}} \in \arg \min_{\theta \in \mathbb{R}^q} \sum_{i \in \mathcal{I}_{\text{fit}}^{(p_{\text{fit}})}} \left(\frac{\widehat{\text{Acc}}(p_i; \theta) - \text{Acc}(p_i)}{\sigma_i} \right)^2, \quad (8)$$

where $\sigma_i > 0$ are the per-observation weights.¹ The fitted model is then used to extrapolate to larger training fractions $p > p_{\text{fit}}$ with the index set $\mathcal{I}_{\text{pred}}^{(p_{\text{fit}})}$. This allows, for example,

¹Implemented with `scipy.optimize.curve_fit` using its default unconstrained setting. Parameter constraints were not imposed during fitting in order to preserve comparability across all five functions under uniform optimization conditions.

the required amount of training data $n_{\text{req}} = \lceil p_{\text{req}} N \rceil$ to be estimated in order to achieve a desired accuracy Acc_{req} by solving $\widehat{\text{Acc}}(p_{\text{req}}; \theta_{p_{\text{fit}}}) = \text{Acc}_{\text{req}}$. The following fit functions are used, as they occur frequently in the literature:

- Algebraic root

$$\widehat{\text{Acc}}(p) = a + b \cdot \sqrt{p} \quad (9)$$

- Arctan

$$\widehat{\text{Acc}}(p) = a + b \cdot \arctan(c \cdot p) \quad (10)$$

- Exponential approximation to asymptotic performance (referred to as “exponential”)

$$\widehat{\text{Acc}}(p) = a - (a - b) \cdot e^{-c \cdot p} \quad (11)$$

- Logarithmic

$$\widehat{\text{Acc}}(p) = a + b \cdot \log_2(p) \quad (12)$$

- Power law

$$\widehat{\text{Acc}}(p) = (1 - a) - b \cdot p^c \quad (13)$$

The algebraic root and logarithmic functions each have $q = 2$ free parameters, while $q = 3$ for the arctan, exponential, and power law functions. Each of these functions exhibits distinct mathematical properties. One important difference is whether the function is upper-bounded. The exponential and arctan functions are upper-bounded and saturate toward an asymptote, whereas the logarithmic and algebraic root functions grow without bound. The power law function can be either unbounded or bounded from above depending on the signs and magnitudes of b and c . Beyond boundedness, the functions also differ in flexibility, growth rate, curvature, and behavior near $p = 0$, which affects how well they can fit the learning curves.

In the evaluation in Section 4, we vary p_{fit} incrementally—thus enlarging $\mathcal{I}_{\text{fit}}^{(p_{\text{fit}})}$ —to assess how increasing access to initial points of $\text{Acc}(p)$ affects extrapolated predictions $\widehat{\text{Acc}}(p)$. So, the prediction is updated with each step in which the network training subset is enlarged. One can summarize the procedure described with the pseudocode shown in Algorithm 1.

3.3. Approach II: Analysis of the Entropy within Data

Entropy quantifies the uncertainty of a probability distribution. Higher entropy indicates greater uncertainty, and equiprobable outcomes attain maximal entropy. In the presented approach, entropy is used to assess whether a given dataset is already sufficiently comprehensive so that further data acquisition can be terminated, similar to the ideas of category (2) in Section 2. Let $\{\mathcal{D}_{\text{train}}^{(n)}\}_n$ denote the nested training subsets used to generate a learning curve (see Section 3.1). We simulate progressive data acquisition by enlarging the train-

Algorithm 2: Entropy-based data saturation detection for progressive data acquisition.

Input: k datasets $\mathcal{D}_{\text{train}}$ of size N ; monitored feature x_{sel} ; step size h_p ; smoothing order o ; threshold t .

Output: Saturation fractions $\{p_{\text{sat}}^{(r)}\}_{r=1}^k$.

```

1 for all repetitions  $r$  with  $1 \leq r \leq k$  do
2   Initialize  $i \leftarrow 1, p_i \leftarrow h_p$ 
3   while the current observed fraction  $p_i \leq 1$  do
4      $n_i \leftarrow \lfloor p_i N \rfloor$ 
5     Select training subset  $\mathcal{D}_{\text{train}}^{(r, n_i)}$ 
6     Discretize  $x_{\text{sel}}$  into  $L = \lfloor \sqrt{n_i} \rfloor$  bins
7     Compute empirical PMF  $\{f_l\}_{l=1}^L$ 
8     Compute entropy  $H(n_i)$  using Eq. (14)
9     Compute normalized entropy  $\tilde{H}(n_i)$  using Eqs. (15)
        and (16)
10    Compute smoothed entropy  $\tilde{H}^*(p_i)$  using Eq. (17)
11    if  $i \geq 2$  then
12      Compute gradient  $\text{GNE}(p_i)$  using Eq. (18)
13      Update running maximum  $\text{Max}_i$ 
14      if  $i \geq 3$  and  $|\text{GNE}(p_i)| \leq t \cdot \text{Max}_i$  then
15        break
16     $i \leftarrow i + 1$ 
17     $p_i \leftarrow i h_p$ 
18   $p_{\text{sat}}^{(r)} \leftarrow \min(p_i, 1)$ 
19 return  $\{p_{\text{sat}}^{(r)}\}_{r=1}^k$ 

```

ing fraction p in steps of size $h_p > 0$, with $p_i = i h_p \in (0, 1]$ for $i \geq 1$ and $n_i = \lfloor p_i N \rfloor$; the index $i = 0$ would correspond to an empty training subset and is therefore excluded. This is the same grid $\mathcal{P} = \{p_i\}_{i=1}^m$, $m = \lfloor 1/h_p \rfloor$, that is used in Section 3.2.

For the investigation in this paper, we compute entropy from the empirical distribution of a selected feature x_{sel} of the feature vector $\mathbf{x}$. The feature x_{sel} is chosen as the most discriminative feature according to a one-way ANOVA F -test (Hogg & Ledolter, 1989). Note that in practical applications this test can be computed from the available training data only; the deviation required for the evaluation in this paper is described in Section 4.4. Selecting the most discriminative feature is expected to capture the dominant variation in the class-conditional distributions and thus that part of the data whose coverage is most relevant for the subsequent classification task. For a given subset size n , we discretize x_{sel} into $L = \lfloor \sqrt{n} \rfloor$ equal-width bins and define the empirical probability mass function (PMF) by the relative frequencies $f_l \geq 0$ with $\sum_{l=1}^L f_l = 1$. The Shannon entropy is

$$H(n) = - \sum_{l=1}^L f_l \log_2 f_l, \quad (0 \log 0 := 0), \quad (14)$$

with maximum

$$H_{\max}(n) = -\sum_{l=1}^L \frac{1}{L} \log_2\left(\frac{1}{L}\right) = \log_2 L. \quad (15)$$

The normalized entropy is

$$\tilde{H}(n) = \frac{H(n)}{H_{\max}(n)} \in [0, 1]. \quad (16)$$

Note that this design deliberately restricts the entropy calculation to a single dimension. Empirical joint entropy estimation over multi-dimensional spaces requires a number of bins that grows exponentially with the dimension and therefore becomes unreliable. This can be quantified as follows. Estimating the Shannon entropy in Eq. (14) from the relative frequencies f_i constitutes a plug-in estimator, which is biased towards lower values, with a bias of order $-(L-1)/(2n)$ nats, i.e. $-(L-1)/(2n \ln 2)$ bits, for L bins and n samples in the well-sampled regime $n \gg L$; outside that regime the bias can be considerably larger in magnitude (Paninski, 2003). What matters is thus the ratio between the number of bins and the number of samples. Retaining the resolution $L = \lfloor \sqrt{n} \rfloor$ per axis, a d -dimensional histogram contains $L^d \approx n^{d/2}$ bins, so that this ratio scales as $n^{d/2-1}$. In the one-dimensional case it therefore decreases with increasing n , whereas already in two dimensions it is independent of n : the histogram contains as many bins as there are samples for every subset size, and the leading-order bias remains at approximately half a nat instead of vanishing; in practice it is larger still, since the well-sampled regime is no longer satisfied. For a joint estimate over the full feature space considered in this paper, the estimate would therefore reflect the sparsity of the histogram rather than the information content of the acquired data. A joint estimate over a small subset of features would remain feasible at a coarser resolution; the corresponding variants and their trade-offs are discussed in Section 4.5.3. Normalizing by $H_{\max}$ according to Eq. (16) does not remedy the bias, since it grows with the same bin count that enters the normalization.

We smooth $\tilde{H}$ by a moving average of order $o \geq 1$ with a truncated window:

$$\begin{aligned} \tilde{H}^*(p_i) &= \frac{1}{o_i} \sum_{l=0}^{o_i-1} \tilde{H}(p_{i-l}), \\ o_i &:= \min\{o, i\}, \quad i \geq 1. \end{aligned} \quad (17)$$

To detect saturation, we track the discrete gradient of $\tilde{H}^*$ across successive steps, resulting in the gradient of normalized entropy (GNE):

$$\text{GNE}(p_i) = \frac{\tilde{H}^*(p_i) - \tilde{H}^*(p_{i-1})}{h_p}, \quad i \geq 2. \quad (18)$$

Let

$$\text{Max}_i := \max_{l=2, \dots, i} |\text{GNE}(p_l)|, \quad i \geq 2,$$

represent the running maximum. Then the saturation onset index is defined as

$$i_{\text{sat}} := \min \left\{ i \geq 3 : |\text{GNE}(p_i)| \leq t \text{Max}_i \right\}, \quad (19)$$

where $0 < t < 1$ is a user-specified threshold for the smoothed gradient magnitude. Equation (19) is evaluated from $i = 3$ onwards, since at $i = 2$ the running maximum consists of $\text{GNE}(p_2)$ itself.

We declare entropy saturation at $p_{i_{\text{sat}}}$. At that point, further data acquisition may be terminated, and the current subset used to train the ML classifier. One can summarize the procedure described with the pseudocode shown in Algorithm 2.

4. EVALUATION

First, this section presents the dataset used to evaluate the two approaches described in Section 3. In Section 4.2 the procedure for generating the learning curves, which represent the ground truth for the following evaluations, is described. Sections 4.3 and 4.4 present the detailed evaluation procedures and the results of the two approaches used in this paper. In Section 4.5, these results are discussed in conjunction with the limitations of the study.

4.1. Dataset

Rolling bearings are one of the most important machine elements in the literature on fault diagnosis of engineering systems (Wang, Lei, Li, & Li, 2020; Lei et al., 2020; Raj, Kumar, & Kumar, 2025). Consequently, fault diagnosis of rolling bearings is utilized as a use case for the evaluations in this paper. This dataset comprises data derived from measurements conducted on two rolling bearing test rigs and under different operating conditions—speed and radial load. Bearings of various types and with different defects are installed on the test rigs, where the corresponding vibration and rotational speed are measured. For each measurement, the signals are acquired over one second with a sampling frequency of 15.625 kHz. The two test rigs differ only in their bearing support; all other aspects are identical. Figure A1 of Appendix A presents the test rigs and the bearing types employed. The specific configurations used for data acquisition are listed in Table A1 of Appendix A. The dataset comprises four distinct classes: healthy bearing (OK), inner race fault (IR), outer race fault (OR), and rolling element fault (RE). The 15 combinations of bearing type and fault state listed there, each measured at two speeds and two load levels, correspond to 60 measurement configurations of ten recordings each, yielding a total of $M = 600$ recorded samples.

Table 1. Structure of the MLP architecture used for training to get the learning curves in Figure 2.

Hyperparameter	Value
Input layer size	10
Hidden layers	3
Layer size 1	50
Layer size 2	50
Layer size 3	40
Dropout (1–3)	0.05
Output layer size	4
Activation function	ReLU
Output activation function	Softmax

The feature vector built from one recorded measurement is 10-dimensional and contains the peak-to-peak, absolute mean, standard deviation, and impulse factor of the vibration signal, as well as the radial load level, rotational speed (features 1–6), and bearing type one-hot encoded (features 7–10). The respective labels contain the fault states of the bearings. The dataset is published on GitHub together with the source code of the evaluation².

4.2. Preparation: Generation of Learning Curves

The accuracy learning curves $\text{Acc}(p)$ are examined with a multi-layer perceptron (MLP) classifier. Table 1 summarizes the architecture of the MLP used. The architecture is chosen based on findings of the investigations in (Braig & Zeiler, 2025). For the generation of the learning curves, 25 % of the dataset is defined as the test set $\mathcal{D}_{\text{test}}$. The remaining 75 % is the full training set $\mathcal{D}_{\text{train}}$ and is used to form the training subsets $\mathcal{D}_{\text{train}}^{(p)}$ for MLP training, with $p \in \{0.01i \mid i = 1, 2, \dots, 100\}$. The split is performed randomly at the level of individual recordings, i.e., without grouping by measurement configuration, so that recordings belonging to the same configuration may be assigned to $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$. The rationale for this and its implications are discussed in Section 4.5.3. For each training subset size p the MLP is initialized with new random model weights. As a result, the accuracy $\text{Acc}(p)$ achieved per relative training subset size p is obtained. This process is repeated $k = 12,000$ times with the different random permutations $\pi^{(r)}$ of the dataset. The resulting 5th, 50th (median), and 95th percentiles of the accuracies are plotted for each relative training subset size in Figure 2.

As can be seen in Figure 2, the median accuracy starts at approximately 35 % and then asymptotically reaches a value of 80 %. The increase in accuracy is particularly large for small values of p . From a p value of approximately 50 %, the final

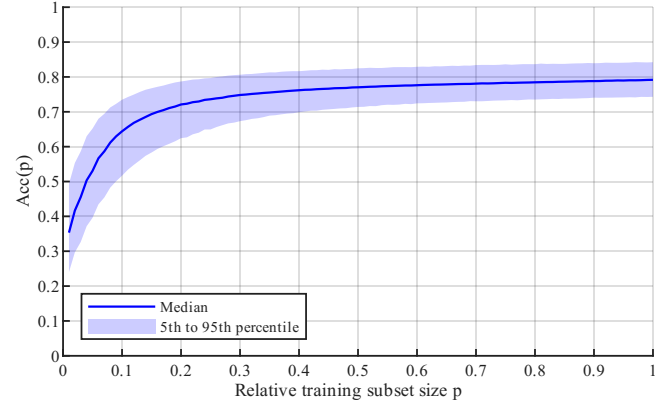


Figure 2. Median and the 5th to 95th percentile across all 12,000 learning curves, which were used as ground truth for the evaluations in this paper. The learning curves represent the accuracy of the MLP classifier over the relative training subset size.

achievable accuracy is almost reached. The interval from the 5th to the 95th percentile starts at approximately $\pm 10\%$ and reaches a width of approximately $\pm 5\%$ as the value of p increases. Here, also, it can be seen that the interval does not become significantly narrower above a value of p of approximately 50 %. At least 95 % of the learning curves achieve an accuracy of more than 70 %. Based on these findings, it can be concluded that the given data allow learning curves to be well represented up to saturation, thus providing a good basis for evaluation in the following sections. Section 4.3 uses the learning curves obtained for Approach I, the extrapolation of the learning curves. Even though Approach II, the entropy analysis, does not require learning curves, the learning curves help in evaluating the results. Therefore, the same permutations $\pi^{(r)}$ of the dataset and training subsets are deliberately used for calculating the learning curves in this section as well as for the entropy analysis in Section 4.4.

4.3. Approach I: Extrapolation of Learning Curves

For the evaluation of *Approach I: Extrapolation of Learning Curves*, Algorithm 1 is executed with the same $k = 12,000$ random splits into $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$ and the same training subsets as in Section 4.2. Two types of evaluation are used. Evaluation I-1 compares the accuracy of the real learning curves $\text{Acc}(p)$ and the predicted learning curves $\widehat{\text{Acc}}(p)$ to assess the quality of the predictions. Evaluation I-2 predicts the relative training size required to achieve a certain accuracy. This value is compared with the actual relative training subset size required. The evaluations are carried out in the same way for all fit functions presented in Section 3.2.

²https://github.com/PHM-Hochschule-Esslingen/Training_Data_Amount_Evaluation

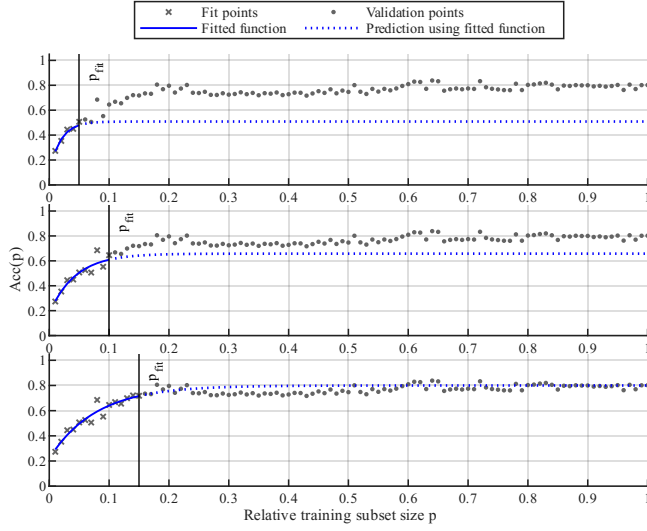


Figure 3. Learning curve accuracy prediction procedure, exemplified by the exponential function and illustrated for three different values of $p_{\text{fit}} \in \{0.05, 0.10, 0.15\}$.

Evaluation I-1: Accuracy

For the evaluation, the fitting threshold is varied over

$$p_{\text{fit}} \in \{0.01i \mid i = 5, 6, \dots, 99\}.$$

The fitting functions are fitted to all points $p \leq p_{\text{fit}}$ and predictions are made for all $p > p_{\text{fit}}$. The evaluation procedure is illustrated in Figure 3 using three different p_{fit} values as examples. To evaluate the results, the root mean square error (RMSE) metric is calculated in two ways. First, the error is considered in the range of the initial fitting set $\mathcal{P}_{\text{fit}}^{(p_{\text{fit}})}$, which refers to the error between the fit function and the true accuracy values fitted (solid line and crosses in Figure 3), leading to

$$\text{RMSE}_{\text{fit}}^{(r, p_{\text{fit}})} = \sqrt{\frac{1}{|\mathcal{P}_{\text{fit}}^{(p_{\text{fit}})}|} \sum_{p_i \in \mathcal{P}_{\text{fit}}^{(p_{\text{fit}})}} \left(\text{Err}^{(r)}(p_i) \right)^2}, \quad (20)$$

with

$$\text{Err}^{(r)}(p_i) = \text{Acc}^{(r)}(p_i) - \widehat{\text{Acc}}^{(r)}(p_i). \quad (21)$$

Second, the RMSE in the range of the evaluation set $\mathcal{P}_{\text{pred}}^{(p_{\text{fit}})}$, which refers to the error between the fit function and the true accuracy values that must be predicted (dashed line and dots in Figure 3), is defined as

$$\text{RMSE}_{\text{pred}}^{(r, p_{\text{fit}})} = \sqrt{\frac{1}{|\mathcal{P}_{\text{pred}}^{(p_{\text{fit}})}|} \sum_{p_i \in \mathcal{P}_{\text{pred}}^{(p_{\text{fit}})}} \left(\text{Err}^{(r)}(p_i) \right)^2}. \quad (22)$$

That is to examine how good the fit functions are when fitting to the given points of the learning curve and, on the other

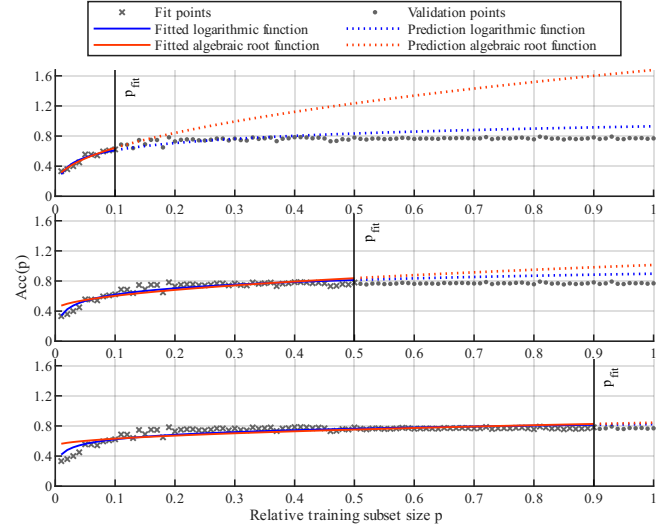


Figure 4. Comparison of fitted logarithmic functions and algebraic root functions for three different values of $p_{\text{fit}} \in \{0.10, 0.50, 0.90\}$.

hand, in predicting the following course of $\text{Acc}(p)$. Figure 4 illustrates exemplary fits of the logarithmic and the algebraic root functions, and the results of the RMSE evaluation are shown in Figures 5 and 6.

In Figure 5 the RMSE_{fit} between the actual accuracy points used for the fit and the resulting fitting functions is given over all $k = 12,000$ repetitions. This means that it is possible to observe how well the functions can be adjusted to the points given for the fit. In addition to the median RMSE, the interval from the 5th to the 95th percentile is also given. At first, for small p , the RMSE values of all fit functions show a local minimum. There are few fit points, so the functions can be adjusted well. The RMSE values of all functions except the algebraic root function then initially increase, reach a maximum, and then decrease. This behavior can be attributed to the fact that the arctan and exponential functions are upper-bounded, and the power law function may also be bounded above depending on the parameters. Therefore, they can accurately represent the flattening of the accuracy for larger p values. The logarithmic function can also better approximate the accuracy as the p value increases, as it can map relative changes such as doublings, but then also flattens sharply early on and can thus well replicate the rapid saturation. The RMSE of the algebraic root function, on the other hand, does not show any decline for larger p values. This function can be used to simulate decreasing gradients, but not those that decrease too drastically. Either the large initial gradient can be modeled, but with a large error in the saturation region towards the end of the learning curve or the flatter gradient at the end of the learning curve can be modeled, but with an increasing error for smaller p values. This behavior is illustrated in Figure 4. As can be seen, the fit of the root function

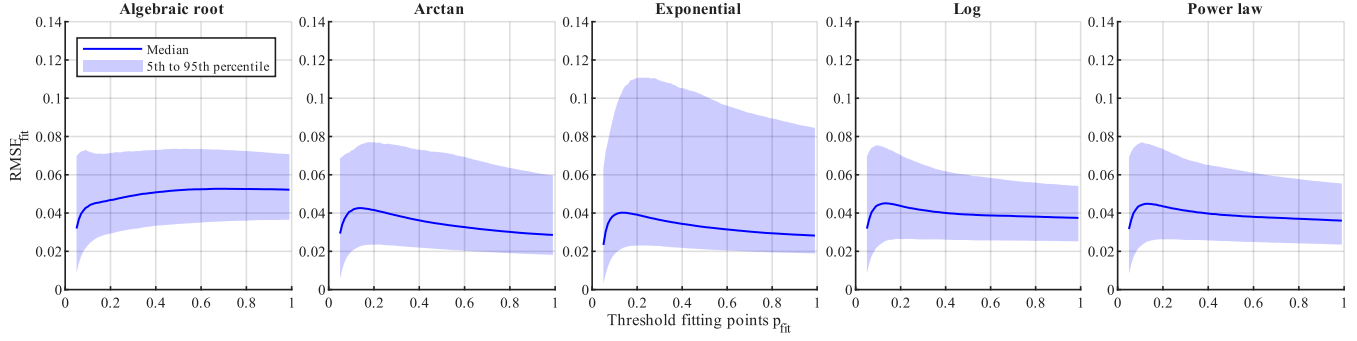


Figure 5. The median and the 5th to 95th percentile interval of the RMSE_{fit} on the fitted points $p \leq p_{\text{fit}}$ over all $k = 12,000$ repetitions.

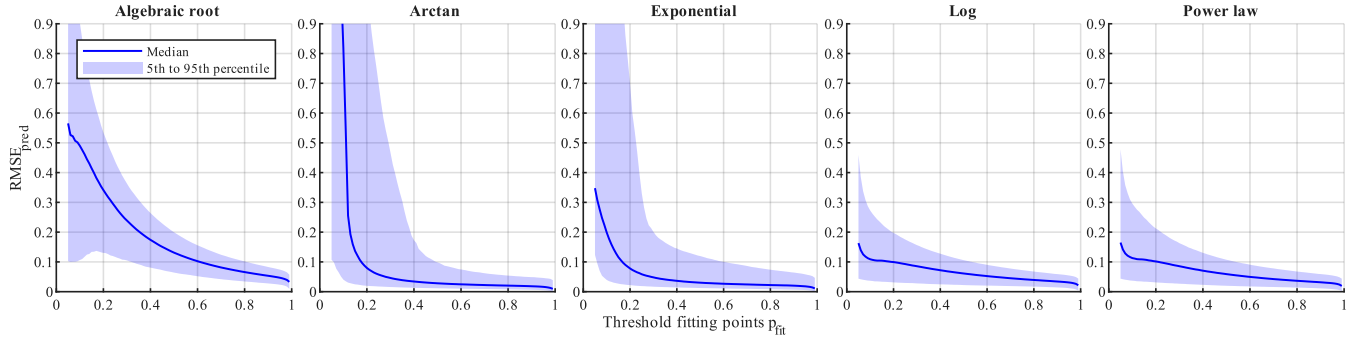


Figure 6. The median and the 5th to 95th percentile interval of the $\text{RMSE}_{\text{pred}}$ on the prediction points $p > p_{\text{fit}}$ over all $k = 12,000$ repetitions.

is relatively good for small p_{fit} in $\mathcal{P}_{\text{fit}}^{(p_{\text{fit}})}$ compared to larger p_{fit} . However, as soon as p_{fit} increases and the flatter part of the learning curve is to be modeled in addition to the steep rise, significant deviations occur on $\mathcal{P}_{\text{fit}}^{(p_{\text{fit}})}$. The logarithmic function, also not upper-bounded, can reproduce the curve much better here. A further noticeable aspect of RMSE_{fit} is that, for the exponential function, although the majority of fits is very good, there are upward outliers that stretch the 5th to 95th percentile interval upwards. This can also be seen in the $p_{\text{fit}} = 0.4$ plots in Figure 7. Here, the percentile interval is slightly wider than for the other functions.

Figure 6 visualizes the $\text{RMSE}_{\text{pred}}$ between the resulting fitting function and the actual accuracy points beyond the fit range, i.e., $\mathcal{P}_{\text{pred}}^{(p_{\text{fit}})}$, which allows conclusions to be drawn about the prediction performance. The RMSE values decrease for all fitting functions as p_{fit} increases, i.e., the predictions of the achievable accuracies improve as more fitting points are used. However, the functions differ in terms of the initial RMSE for small p_{fit} and the rate of decline as p_{fit} increases. The logarithmic and power law functions show very similar behavior. Even with just a few fitting points, they provide good predictions about the further progression of the learning curves. Both the RMSE median and the interval from the 5th to the 95th percentile are relatively small. The

two upper-bounded functions, arctan and exponential, also exhibit very similar behavior to each other. At first for small p_{fit} values, the RMSE median and the 5th to 95th percentile interval are relatively large and then drop significantly.

To explain this phenomenon, Figure 7 depicts the fitted functions of all $k = 12,000$ runs, showing the median and the interval from the 5th to the 95th percentile. As can be seen in the top row for $p_{\text{fit}} = 0.05$, the use of the arctan function for the prediction range $\mathcal{P}_{\text{pred}}^{(p_{\text{fit}})}$ usually results in a strong overestimation of accuracy with values that are mostly well above the maximum accuracy value of one. For the exponential function, the upper limit of the percentile interval also reaches values well above one. In addition, it is noticeable that at the same time, the median has its saturation range too low, at an accuracy of approximately 60 %, although the true values, as can be seen in Figure 2, asymptotically converge mostly to 80 %. However, if p_{fit} is increased to 0.4 (see second row of Figure 7), both functions show a good prediction, with a median saturation in the range of just under 80 %. The occurrence of this saturation can be attributed to the upper bound of the two functions. This leads to the median of the RMSE in Figure 6 becoming even better for larger p_{fit} than for the logarithmic and power law functions, thus achieving the best results across all functions. It should be noted that the power law function can also be limited depending on the

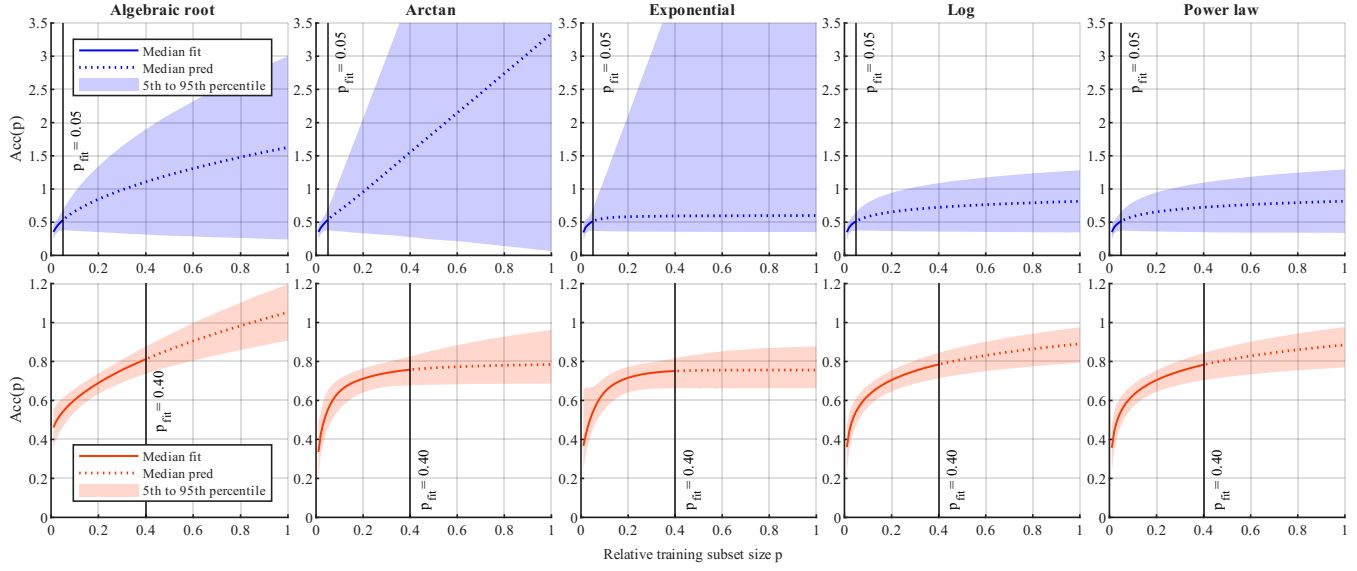


Figure 7. The median and the 5th to 95th percentile interval of the fit functions fitted on the learning curves for $p_{\text{fit}} = 0.05$ and $p_{\text{fit}} = 0.40$.

choice of parameters, but Figure 7 indicates that at $p = 1$, a slight increase can still be observed.

The algebraic root function in Figure 6 also starts with a relatively large RMSE median and a large interval between the 5th and 95th percentiles, followed by a slow decline for larger p_{fit} . However, it does not reach the low RMSE values achieved by the other functions. This can be attributed to the problems mentioned above, namely the simultaneous modeling of the steep rise and the saturation range. Figure 7 illustrates this for $p_{\text{fit}} = 0.4$, as a strong increase can still be observed for p values close to one.

Evaluation I-2: Required Training Size

We also try to predict the required size of the training subset to reach a desired target accuracy. This target accuracy has to be set according to the required use case and the desired accuracies that should be reached. Based on the accuracy curves in Figure 2, we set multiple target accuracies for the evaluations at the values $\text{Acc}_{\text{req}} \in \{60\%, 65\%, 70\%, 75\%, 80\%\}$. The deviation of the predicted training subset size where the accuracy exceeds the target accuracy and the actual training subset size where the obtained accuracy value first exceeds the target accuracy is calculated by

$$\Delta p_{\text{Acc}_{\text{req}}} = p_{\text{Acc}_{\text{req}}} - \hat{p}_{\text{Acc}_{\text{req}}}, \quad (23)$$

where $p_{\text{Acc}_{\text{req}}}$ is the size where the learning curve first exceeds the target value of Acc_{req} and $\hat{p}_{\text{Acc}_{\text{req}}}$ for the fitted function of the learning curve. In some cases, the evaluation of the predicted required training size is not possible. These cases include the fit function not reaching the target value, the recorded learning curve not reaching the target value, both

not reaching the target value, or the recorded accuracy having already reached the target value. In all other cases, the evaluation can be carried out as described. The results are shown in Figure 8 and in Table B1 of Appendix B. The error $\Delta p_{\text{Acc}_{\text{req}}}$ is given for the five different target accuracies, and three fitting thresholds $p_{\text{fit}} \in \{0.05, 0.10, 0.15\}$ are evaluated. For each case, the 5th, 25th, 50th, 75th, and 95th percentiles are given. As can be seen from Table B1 of Appendix B, the number of learning curves evaluated varies due to the reasons mentioned above.

As shown in Figure 8, the error $\Delta p_{\text{Acc}_{\text{req}}}$ increases the greater the target accuracy Acc_{req} for which a prediction of $p_{\text{Acc}_{\text{req}}}$ is to be made. This is plausible, since greater target accuracy requires more far-reaching predictions. In addition, the learning curves flatten out as p increases, so small deviations in the estimated accuracy, e.g., due to outliers, lead to large $\Delta p_{\text{Acc}_{\text{req}}}$. It can also be seen that, especially with high target accuracies, the prediction results improve the larger p_{fit} is selected. This is because the larger the p_{fit} , the larger the initial known course of the learning curve that can be used for the fit. With high levels of target accuracy, the errors $\Delta p_{\text{Acc}_{\text{req}}}$ are almost always greater than zero, which means that the prediction using the function fits is usually too high, resulting in an optimistic estimate with less data than is actually necessary. The fact that this effect occurs primarily with high levels of target accuracy is consistent with the fact that these high accuracy values are only reached by functional fits that achieve these levels of accuracy at all and therefore tend to have higher values, which makes overestimation more likely. For small and medium target accuracies up to 70%, the medians of $\Delta p_{\text{Acc}_{\text{req}}}$ are close to zero and the percentile intervals are also relatively narrow, which means that the amount of

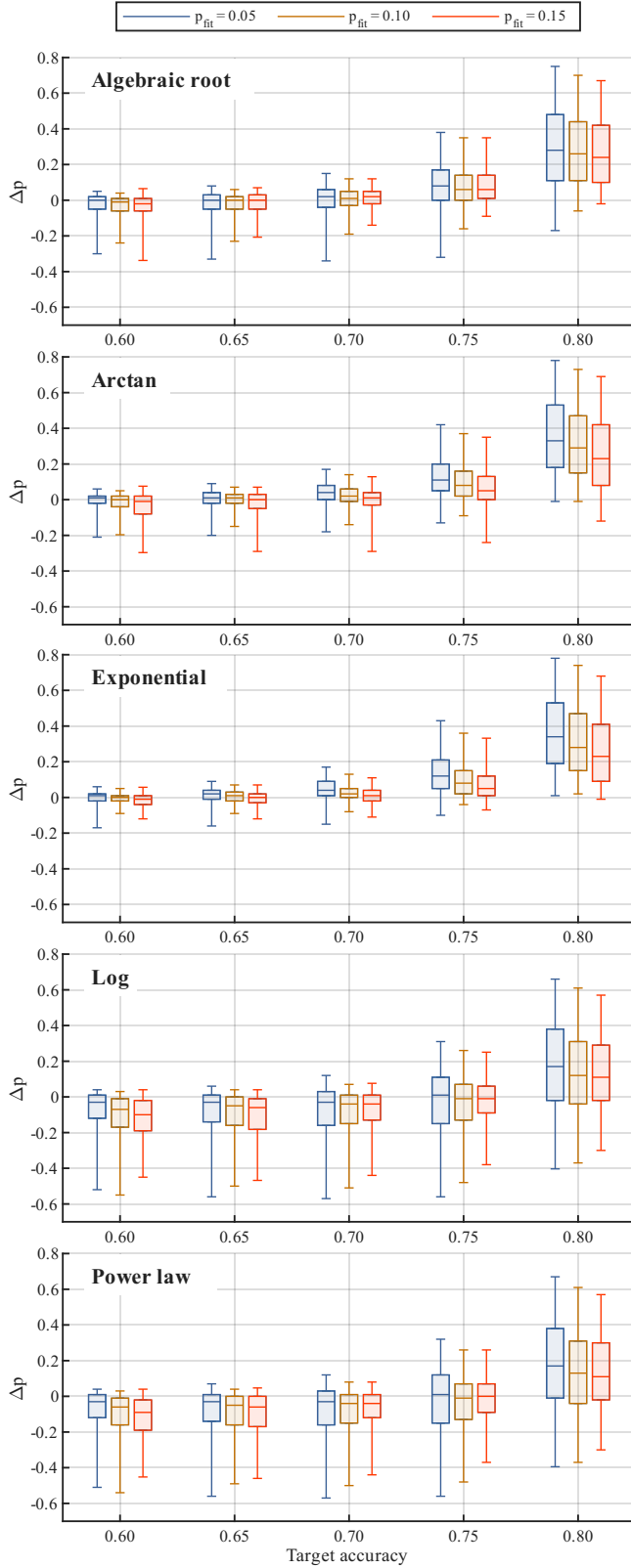


Figure 8. Deviation $\Delta p_{\text{Acc}_{\text{req}}}$ of the predicted and the actual training subset size for five given target accuracies Acc_{req} and three fitting thresholds p_{fit} over all $k = 12,000$ repetitions. Given are the 5th, 25th, 50th, 75th, and 95th percentiles.

data required to achieve these target accuracies can be accurately estimated. The algebraic root, arctan, and exponential functions perform best here, while logarithmic and power law have negative medians, i.e., they overestimate the amount of data required and also scatter more widely in the percentile intervals.

As can be seen in Table B1 of Appendix B, the number of evaluated learning curves varies significantly. A more detailed analysis of the exclusion criteria mentioned above shows that, especially for small p_{fit} , many fitted learning curves do not reach Acc_{req} . The fitted functions are too flat. This effect is particularly noticeable with the exponential function. This also becomes clear when examining the fits in Figure 7 for $p_{\text{fit}} = 0.05$. As can be seen, the median for the exponential function lies at an accuracy of approximately 60 %, so that even target accuracies of 60 % are frequently not achieved. For larger Acc_{req} , this effect also occurs with the logarithmic and power law functions, as can also be seen in the figure. The situation is different for the algebraic root and arctan functions, which, as can be seen in Figure 7 for $p_{\text{fit}} = 0.05$, rise well above an accuracy of 100 %. Another major factor contributing to the reduction in the number of evaluable learning curves is that, especially for small Acc_{req} and larger p_{fit} , p_{fit} is often already above the p required to achieve Acc_{req} . Therefore, no further prediction is possible, and the associated learning curve must be excluded from the evaluation.

4.4. Approach II: Analysis of the Entropy within Data

For the evaluation of the GNE procedure, Algorithm 2 is executed with the same $k = 12,000$ random splits into D_{train} and D_{test} and the same training subsets as in Section 4.2 and the evaluation of Approach I. This allows reference to be made to the accuracy curves for evaluation purposes. The number of bins of the empirical PMF is not a free parameter but is tied to the subset size by the rule $L = \lfloor \sqrt{n} \rfloor$ for the reasons given in Section 3.3.

For feature x_{sel} , the impulse factor is selected, as it is the most discriminative feature in the given dataset according to the ANOVA F -test (see Section 3.3). It should be noted that, in practice, the feature can only be selected from the data available at the time of selection, i.e., from the current training subset and not from the full dataset. For the evaluation, the F -test is therefore carried out separately for every repetition and every training subset size, using only the respective training subset. Using the feature selected in each individual case, however, would result in different features being monitored across repetitions and training subset sizes, so that the entropy sequence would refer to different marginal distributions at different steps of the same acquisition process and the stopping points would no longer be comparable. The feature selected most frequently is therefore adopted consistently for the evaluation. As shown in Figure C1 in Appendix C, the

impulse factor is already the most frequently selected feature for training subset sizes from $p = 0.02$ onward and is selected in nearly all repetitions from $p \approx 0.15$ onward³. Since the selection frequencies are aggregated over all repetitions, whose training sets jointly cover the entire dataset, this choice is strictly speaking not independent of the individual test sets; given the clear dominance of the impulse factor, its influence on the evaluation is negligible.

The simulation of progressive data acquisition is performed with the step size $h_p = 0.01$, since this matches the resolution of the recorded learning curves. Because the observed fractions are given by $p_i = i h_p$, this value affects the GNE approach in two ways: it determines the resolution at which the entropy curve is sampled, and it fixes the first evaluated fraction $p_1 = h_p$, at which monitoring starts. Matching the resolution ensures that an accuracy value is available for every $p_{i_{\text{sat}}}$ determined by the GNE in Evaluation II-1 and that the GNE and accuracy for the correlation calculated in Evaluation II-2 are sampled on the same grid, whereas a larger h_p would compute the difference quotient over a wider interval and reduce the resolution with which $p_{i_{\text{sat}}}$ can be localized. The first evaluated fraction, in turn, represents the amount of data already available at the beginning of monitoring and must be small enough to capture the steep initial section of the entropy curve, because the running maximum Max_i in Eq. (19) is formed from the sequence starting at p_1 . If monitoring started at a substantially larger fraction, the gradients would no longer differ substantially from one another and the criterion would terminate almost immediately. For the small threshold values, which constitute the operating region relevant for achieving practically high accuracies, termination occurs at fractions clearly above $p_1 = 0.01$ (see Table 2), so that the stopping point is governed by the saturation behavior of the entropy curve rather than by where the sequence begins. For large thresholds, in contrast, the criterion already terminates within the first few evaluation steps and the stopping point is governed by the low sensitivity of the threshold itself rather than by the choice of p_1 .

For the two remaining hyperparameters—the threshold t and the smoothing order o —there is no obvious choice of values, which is why the effects of different values on the GNE method are investigated. In the authors' view, t has the greatest influence on the method, as it directly controls the sensitivity of the stopping decision. Therefore, it is not fixed but is systematically varied during the evaluation: to determine the saturation onset index (see Eq. (19)), the values $t \in \{0.5, 0.4, 0.3, 0.2, 0.1, 0.05, 0.01\}$ are examined.

The moving average in Eq. (17) replaces each difference between adjacent entropy values by an average of the preceding differences. Smoothing therefore reduces large differences,

such as the steep initial steps and noise-induced peaks, so that the resulting GNE values decrease. Since Max_i is formed from the past GNE values, the absolute termination threshold $t \text{Max}_i$ is reduced as well. At the same time, averaging attenuates random dips of the gradient, i.e., individual small differences that could otherwise trigger a premature termination. These effects act in opposite directions, so that no systematic shift of the termination point can be expected a priori. Furthermore, as the size of the training subset increases, the entropy changes get smaller, since newly added samples have a smaller influence on the entropy. The entropy curve thus becomes smoother as p_i increases, and the smoothing itself has less and less influence. Overall, the influence of o is expected to be small compared to that of t . Therefore, the main evaluation is limited to $o = 5$, and $o \in \{1, 10\}$ are included as an excursus to show that the value of o does not alter the conclusions drawn for the use case considered here.

To evaluate *Approach II: Analysis of the Entropy within Data*, two procedures are used. In contrast to Approach I, Approach II does not aim to predict the amount of data required for a given target accuracy but rather to enable statements to be made as to whether it makes sense to generate additional data beyond the currently available dataset or to stop data acquisition. Evaluation II-1 therefore examines whether the same GNE value across all repetitions leads to similar accuracies. For robust statements, a specific GNE value should lead to a certain consistent accuracy across all repetitions. For Evaluation II-2, the correlation between the accuracy achieved and the GNE is considered in order to assess the extent to which a decreasing GNE value correlates with increasing accuracy. However, note that for Approach II the accuracy values are not part of the approach but are used for the evaluation only.

Evaluation II-1: Accuracy at GNE Stopping Criterion

In this evaluation, $p_{i_{\text{sat}}}$ is determined for each of the k repetitions for a specified threshold value t . Subsequently, the accuracy at this p value is determined for the corresponding learning curve from Section 4.2. The results over the k repetitions are shown as a box plot in the upper half of Figure 9. The threshold values $t \in \{0.5, 0.4, 0.3, 0.2, 0.1, 0.05, 0.01\}$ are evaluated. The 5th, 25th, 50th, 75th, and 95th percentiles are shown. In addition, the lower half compares the accuracy achieved with $p_{i_{\text{sat}}}$ to the maximum accuracy achievable with the full training set of the respective repetition. To prevent outliers from distorting the result, the median of the last 20 % accuracy values of the respective accuracy curve is considered as the maximum achievable accuracy.

As shown in Figure 9, the accuracy achieved increases as the GNE threshold value decreases and gets closer and closer to what would be achieved by training with all training data. This finding also coincides with the findings from the sub-

³For the smallest subset ($p = 0.01$, $n = 4$), the F -statistic is not defined if every class is represented by at most one sample; these cases were skipped.

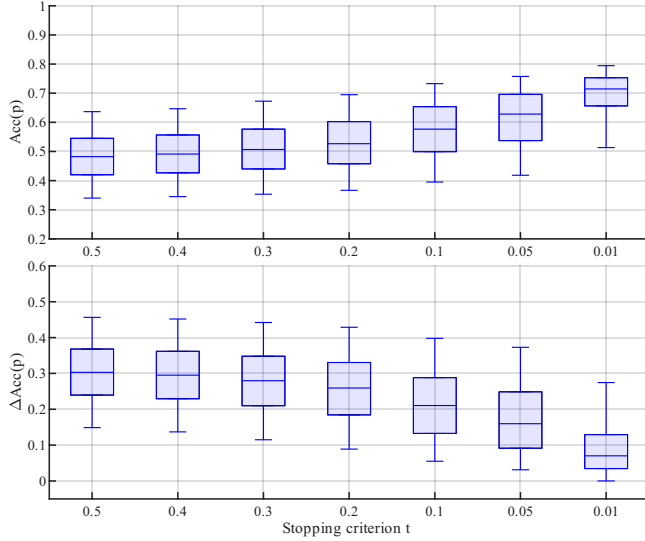


Figure 9. The learning curve accuracies $\text{Acc}(p)$ at the $p_{i_{\text{sat}}}$ determined by the GNE using the threshold value t and for a smoothing order of $o = 5$ over all $k = 12,000$ repetitions, as well as the corresponding accuracy deviation $\Delta\text{Acc}(p)$ to the maximum accuracy achievable with the full training set. The 5th, 25th, 50th, 75th, and 95th percentiles are shown.

sequent Evaluation II-2. As expected, the GNE value decreases as the amount of data increases and can therefore be used as an indicator to evaluate the availability of sufficient training data. In addition, the range of accuracy values decreases sharply when $p_{i_{\text{sat}}}$ of the termination training subset lies within the saturation range of the learning curves. This phenomenon is clearly visible in the evaluation when moving from $t = 0.05$ to $t = 0.01$, with the 25th to 75th percentile interval decreasing from over 15% accuracy to around 10% accuracy. Therefore, the values of $p_{i_{\text{sat}}}$ may fluctuate yet still yield similar accuracy values. This is also confirmed by Figure 10, which shows the distribution of $p_{i_{\text{sat}}}$ across all repetitions. For $t = 0.01$, the dispersion of $p_{i_{\text{sat}}}$ is actually strongest, although the corresponding box plot in Figure 9 is narrowest. Across all t , it can be seen that the resulting values for $p_{i_{\text{sat}}}$ mostly cluster around one value. While this value is below $p_{i_{\text{sat}}} = 0.05$ for the larger $t \in \{0.5, 0.4, 0.3, 0.2\}$, the center of gravity shifts to values between 0.05 and 0.1 for $t \in \{0.1, 0.05\}$. For $t = 0.01$, $p_{i_{\text{sat}}}$ approaches the value 0.2, for which the median of the accuracies in Figure 2 lies between the values of 70% and 80%.

Evaluation II-2: Correlation Between Accuracy and GNE

In Evaluation II-2 the correlation between the achieved accuracy and the GNE is calculated to assess whether a decreasing GNE value correlates with increasing accuracy. For comparison, the differences between the distribution of the increasingly larger training subset and the distribution of the full training dataset $\mathcal{D}_{\text{train}}$ (mean square error of the correspond-

ing histograms, MSE) are also considered. These histograms are formed from the same marginal distribution of x_{sel} that underlies the GNE, so that both quantities refer to the same one-dimensional projection of the data. The correlations of the resulting MSE values with the accuracy and with the GNE are then calculated. Note that in practical application, these MSE distribution differences cannot be calculated, unlike the GNE, but they do provide a further evaluation option in this paper. Figure 11 shows distributions of the different correlation values over all evaluation repetitions. The blue distribution illustrates the correlation between the MSE and the accuracy achieved through training. As can be seen, the curves correlate strongly negatively with each other. This means that as the sample size increases, the samples drawn increasingly approach the distribution of the full training dataset, and the accuracy increases with increasing training sample size. The correlation between the progression of the MSE and the progression of the GNE with increasing sample size is shown in orange. As can be seen, the two correlate strongly with each other. This means that the GNE, which maps the similarity of successive sample sizes, correlates strongly with the MSE, the similarity of these samples to the full training dataset. Therefore, the GNE has a similar significance to the MSE, with the advantage that the distribution of the full training dataset does not need to be known to calculate the GNE, which means that the GNE can be easily calculated in practice. The strong positive correlation between MSE and GNE suggests that GNE should also correlate strongly negatively with accuracy, analogous to the strong negative correlation between MSE and accuracy. This is exactly what is shown in green. The shape of the distribution looks very similar to the blue one. However, the distribution has shifted toward smaller correlation values in terms of magnitude, since the MSE is always compared to the full training dataset, whereas GNE is only compared between two steps in the growing sample sequence without knowing the overall distribution. It can be concluded that the GNE proposed in this paper correlates strongly with the achievable accuracy and is therefore suitable for terminating the acquisition of further data when a specified GNE threshold value is reached. However, it is difficult to assign a GNE value to an accuracy value, which can vary greatly depending on the application and, therefore, requires practical experience.

It is worth emphasizing once again which quantities are included in these correlations. Both the GNE and the MSE are calculated from the marginal distribution of the individual feature x_{sel} , while the accuracy $\text{Acc}(p)$, with which they correlate, is obtained from the MLP trained on the complete ten-dimensional feature vector. The strong negative correlations in Figure 11 therefore indicate that the stabilization of the marginal distribution coincides with the point at which the classifier no longer benefits significantly from additional data drawn from the entire feature space. For the dataset exam-

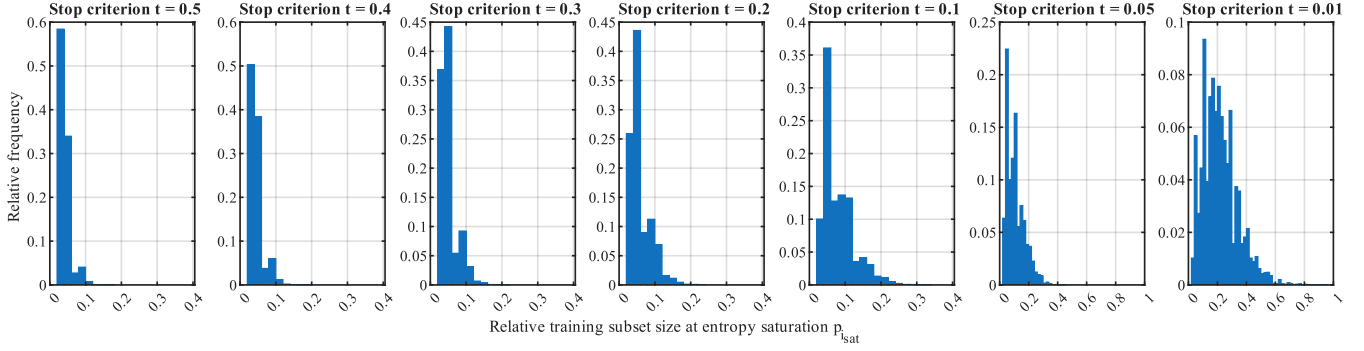


Figure 10. Distribution of $p_{i_{\text{sat}}}$ across all $k = 12,000$ repetitions for different threshold values t and for a smoothing order of $o = 5$.

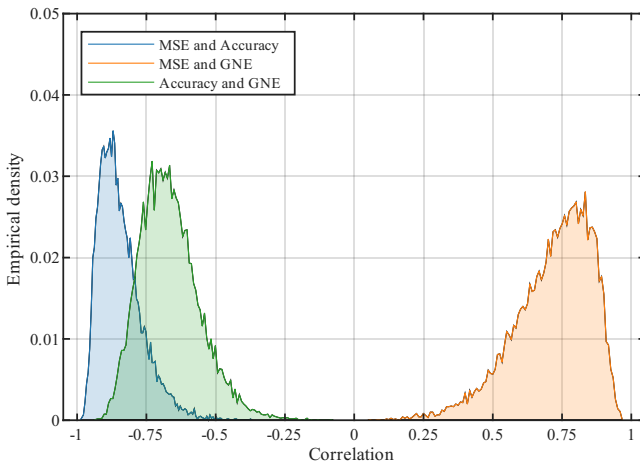


Figure 11. Correlation between accuracy, MSE, and GNE with rising training subset size over all $k = 12,000$ repetitions and using a smoothing order of $o = 5$.

ined here, the saturation of the selected marginal distribution thus provides an empirical indication of the sufficiency of the multivariate dataset, even though the criterion itself considers only one dimension. The implications of the univariate design are discussed in Section 4.5.3.

Sensitivity Analysis of the Smoothing Order

To quantify the influence of the smoothing order o , Evaluations II-1 and II-2 were repeated for $o \in \{1, 10\}$ with all other settings remaining unchanged. Table 2 lists the median accuracy at termination, the median deviation $\Delta \text{Acc}(p)$ from the maximum achievable accuracy, and the median value of $p_{i_{\text{sat}}}$ for each combination of o and t . The counterparts of Figures 9–11 for $o = 1$ and $o = 10$ are provided in Appendix D-1 and Appendix D-2 as Figures D1–D3 and Figures D4–D6, respectively.

As explained in the first part of Section 4.4, smoothing affects both the GNE values and the running maximum Max_i , and these effects act in opposite directions on the termination

point. In addition, the entropy curves in the chosen evaluation example are, on average, already fairly smooth without smoothing: Figure D7 in Appendix D-3 shows that the median GNE curve for $o = 1$ decreases largely monotonically. Another effect contributing to a small difference in the determination of $p_{i_{\text{sat}}}$ between smoothing with $o = 5$ and $o = 10$ is the rapid attainment of the termination threshold for small $p_{i_{\text{sat}}}$ values, which is particularly the case for larger t values. Due to the rapid termination, the different smoothing window lengths have little effect: since the truncated window in Eq. (17) contains only the existing past values, $o = 5$ and $o = 10$ yield identical smoothed entropy values up to $i = 5$.

For these reasons, it is expected that the choice of the smoothing order o during the evaluation will have little effect on the results. This is precisely what the analysis confirms. Between the median termination accuracies and the median $p_{i_{\text{sat}}}$ values in Table 2, only slight and non-monotonic differences across o can be observed. Comparing this with the changes that result from a decreasing stopping criterion t confirms the initial assumption that t has a significantly stronger influence on the results than o . Varying t changes the accuracy by 0.20 ($o = 1$), 0.23 ($o = 5$), and 0.22 ($o = 10$). Varying o changes it by at most 0.03. The accuracy percentiles in Figures 9, D1, and D4 also show only very slight changes between different o values compared to the changes between the t values. The same is true for the distributions of $p_{i_{\text{sat}}}$ shown in Figures D3, 10, and D6. The correlations shown in Figures 11, D2, and D5 also vary only slightly between the o values.

It can therefore be concluded that the findings in Section 4.4 are not sensitive to the specific choice of o , which makes it reasonable to limit the main analysis to $o = 5$.

4.5. Discussion

The results from both approaches demonstrate complementary strengths and limitations that are worth examining in conjunction. The following sections begin with a discussion of these results and subsequently delineate the present limitations of the approaches and the evaluations conducted.

Table 2. Sensitivity of the criterion to the smoothing order o . Median accuracy at termination, median deviation $\Delta\text{Acc}(p)$ from the maximum achievable accuracy, and median $p_{i_{\text{sat}}}$ for each combination of the smoothing order o and the stopping threshold t .

o	Stopping criterion t						
	0.5	0.4	0.3	0.2	0.1	0.05	0.01
Median accuracy at termination							
1	0.49	0.50	0.51	0.53	0.57	0.61	0.69
5	0.48	0.49	0.51	0.53	0.58	0.63	0.71
10	0.48	0.49	0.50	0.52	0.56	0.60	0.70
Median accuracy deviation $\Delta\text{Acc}(p)$							
1	0.29	0.28	0.27	0.26	0.22	0.18	0.09
5	0.30	0.30	0.28	0.26	0.21	0.16	0.08
10	0.30	0.30	0.28	0.27	0.22	0.18	0.07
Median $p_{i_{\text{sat}}}$							
1	0.04	0.04	0.04	0.05	0.06	0.08	0.16
5	0.03	0.03	0.04	0.04	0.06	0.09	0.20
10	0.03	0.03	0.04	0.04	0.06	0.07	0.18

4.5.1. Approach I: Extrapolation of Learning Curves

Approach I provides an estimate of the amount of training data required n_{req} for a given target accuracy V_{req} . This estimate depends on two factors: the choice of the fitting function and the available fitting set $\mathcal{P}_{\text{fit}}^{(p_{\text{fit}})}$. Regarding the choice of fitting function, the results of Evaluation I-1 show that no single function dominates across all settings. However, it is evident that when p_{fit} is small, logarithmic and power law functions provide the most accurate predictions for the subsequent course of the learning curve, as they can flexibly capture the steep initial rise. So, if only a small amount of data is available, it makes sense to use these functions to predict accuracy. For larger p_{fit} values, the predictions of the upper-bounded arctan and exponential functions improve steadily, eventually outperforming even the logarithmic and power law functions. Their saturation behavior matches the empirically observed flattening of the accuracy learning curves very well. The algebraic root function cannot simultaneously model both the steep initial rise and the saturation region of the learning curve and is therefore less suitable for modeling the learning curve.

The prediction of the required training data size to a requested target accuracy Acc_{req} , observed in Evaluation I-2, shows a systematic dependence on Acc_{req} . At accuracy levels of up to about 70 %, all functions, except logarithm and power law, yield median errors $\Delta p_{\text{Acc}_{\text{req}}}$ close to zero, indicating reliable estimates. At higher accuracy levels, most functions exhibit a positive median bias at 75 % and all functions at 80 %, meaning the fitted functions predict a smaller training subset size than is actually required, resulting in an overly optimistic estimate. This is a consequence of the significantly flattened

shape of the learning curve in the saturation region, resulting in small errors in the estimated asymptote leading to large errors in the predicted p_{req} . It should be noted here once again that, for the reasons described in *Evaluation I-2: Required Training Size* of Section 4.3, the number of learning curves available for analysis in Evaluation I-2 naturally varies—a fact that must be taken into account when interpreting the results.

These bias patterns, visible in Figure 8 and in Table B1 of Appendix B, support the recommendation of Mahmood, Lucas, Acuna, et al. (2022) to bound n_{req} by an ensemble rather than relying on a single function, since the logarithmic and power law functions consistently provide the most conservative estimate, whereas the medians of the remaining three functions lie close to zero for target accuracies up to 70 %, with both slight over- and underestimation of the required amount of data occurring. As all five functions underestimate the required amount of data at $\text{Acc}_{\text{req}} = 80$ %, however, such an ensemble should be complemented by a correction factor in the spirit of the factor τ proposed by the same authors.

4.5.2. Approach II: Analysis of the Entropy within Data

Approach II addresses a different but complementary question: not how much data are required to achieve a certain level of accuracy, but whether the currently available data are already sufficient to discontinue further data acquisition. Its main advantages are that it requires no model training and operates incrementally. Each new batch of acquired data can be evaluated immediately with respect to its information contribution. Therefore, Approach II can be applied online during data acquisition with negligible computational effort, enabling practitioners to make informed decisions about whether to continue or terminate data acquisition at any point.

The evaluations confirm that the GNE has a strong negative correlation with the classification accuracy and that the adjustable threshold value t allows controlling the achievable accuracy, making the GNE suitable as a stopping rule for data acquisition. If t is chosen to be sufficiently small, one can assume that when the GNE algorithm terminates, the most recently acquired data points have provided little new information. The evaluation conducted shows that the training subset available at termination provides a good approximation of the full training set. Conversely, this means that the accuracy achievable when training the ML model is close to the maximum accuracy attainable with the full available training set. In practice, one would assume that the training subset available at termination approximates the underlying operational population with sufficient accuracy, provided that the data acquisition covers the relevant operational state space (see Section 4.5.3).

4.5.3. Limitations

The two approaches examined have different limitations, discussed in the following. The main limitations of *Approach I: Extrapolation of Learning Curves* are as follows:

Computational cost: A practical limitation of Approach I is its computational cost. Generating the learning curves requires that the ML model be retrained from scratch repeatedly for each training subset size up to the available amount of data. While this is feasible for simple architectures and small datasets, the approach does not scale well to computationally intensive models or large datasets. In this study, a fixed MLP architecture whose hyperparameters were set based on previous work was used for evaluation. In practical applications, however, the situation is further complicated by the fact that such prior experience is usually not available. Hyperparameter optimization must therefore be performed for each training subset to determine the corresponding point of the learning curve, which means an even greater computational burden.

Required data: Since Approach I is based on training ML models, the data must meet all the requirements necessary for this training in order to be usable. For example, if the approach is based on supervised learning, as in this study, labeled data must be available. Only then can the learning curve be determined. For fault classification, the data must contain a sufficient number of samples from each fault class, as the underrepresentation of rare classes renders the corresponding learning curve points unreliable. To determine a meaningful learning curve, the dataset must be split into independent training and test sets for training and testing the ML models, which further reduces the amount of available training data.

The main limitations of *Approach II: Analysis of the Entropy within Data* are:

Missing accuracy: For Approach II only the GNE threshold t can be set and not the target accuracy V_{req} . Although the entropy saturation criterion indicates that the empirical distribution of the selected feature x_{sel} has stabilized, the level of accuracy achieved at this point depends on the dataset and cannot be determined a priori.

Calibration of the GNE threshold: The GNE threshold t is a user-defined hyperparameter that has a significant impact on the termination of the GNE algorithm. Setting t correctly requires expertise and empirical calibration using representative datasets. In practical applications, it remains to be seen whether a consistent, meaningful value for t can be determined across various use cases.

Calculation of entropy: The evaluation in this paper uses a single feature x_{sel} selected by a one-way ANOVA F -test to calculate the entropy used in Approach II. As justified in Section 3.3, this one-dimensional design avoids the estima-

tor bias that a joint histogram would incur at the subset sizes examined here. The cost of this one-dimensional approach is that the saturation of a single marginal distribution does not imply that dependencies among features are accounted for. As a result, the chosen procedure could terminate data collection prematurely. However, Evaluation II-2 indicates that this effect is not dominant here, since the accuracy with which the univariate GNE correlates is determined based on the complete feature vector. In any case, it should be noted that considering one-dimensional entropy is an evaluation design choice in this paper and not a limitation inherent in Approach II. Extending entropy analysis to multiple features—which are evaluated jointly or combined into a single stop criterion—could improve the robustness and generalizability of Approach II.

Beyond the one-way ANOVA, other selection criteria are conceivable, such as a redundancy-aware information-theoretic ranking based on joint mutual information. Whereas the ANOVA F -statistic scores each feature individually, such a ranking favors features that are complementary. Selecting a small subset in this way and combining the corresponding univariate GNE curves into a single stopping rule would capture more of the multivariate structure while retaining univariate entropy estimation. The ranking itself, however, requires estimating mutual information between discretized features, which reintroduces the small-sample effects discussed in Section 3.3 precisely during the early phase of data acquisition. The ANOVA F -statistic was therefore preferred here, as it requires no discretization. As an alternative to feature selection, a dimensionality reduction such as principal component analysis (PCA) could capture multivariate interactions. The projection is, however, unsupervised, and is itself estimated from the growing dataset, so that the GNE would conflate a change of the data distribution with a change of the projection. An empirical comparison of these variants remains subject to future work.

In addition to the limitations inherent in the specific approaches, further general limitations were imposed during the *evaluation conducted* in this paper:

Generalizability of the dataset: Both approaches were evaluated using a single dataset on rolling bearings with $M = 600$ samples and a fixed 10-dimensional feature space. Although this dataset provides a well-controlled basis for comparison, the generalizability of the results to other application domains, larger datasets, or higher-dimensional feature spaces cannot be assumed. Further empirical validation across multiple datasets and use cases is required before practical deployment. For this reason, the authors have made the dataset and the source code publicly available on GitHub⁴.

⁴<https://github.com/PHM-Hochschule-Esslingen/TrainingDataAmountEvaluation>

In-distribution scope of the evaluation: In accordance with the definition introduced in Section 1, the evaluation performed here quantifies in-distribution sufficiency: $\mathcal{D}_{\text{test}}$ is drawn randomly from $\mathcal{D}$ and therefore contains the same bearing types, rotational speeds, and load levels as the training data. Since the split is random at the level of individual recordings, the ten recordings for each of the 60 measurement configurations are distributed across both sets, so that recordings from the same bearing at the same operating point appear in both $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$. This is a deliberate design decision, since the object of study is the estimation methods themselves and not the diagnostic model. To evaluate them, a controlled ground truth is required, i.e., learning curves with a canonical shape and low variance, which are provided by the random splits across all $k = 12,000$ repetitions. A learning curve $V(n)$ is moreover well-defined only if increasing the training set changes the number of samples but not the distribution from which they are drawn. In the case of a grouped or leave-one-condition-out partition, each additional group would expand the covered part of the operational state space, so that the resulting curve would mix the effect of sample size with that of condition coverage and would no longer be a function of n alone. The evaluation of performance across disjoint sets of conditions is a separate research question addressed, e.g., in works on transfer learning (Braig & Zeiler, 2025). If the evaluation distribution were shifted, a discrepancy between the extrapolated and observed learning curves could no longer be attributed to the fitting function, as it would be confounded with the effect of the shift.

Nevertheless, there are two implications for the interpretation of the results. First, the in-distribution data requirement estimated using Approach I should be understood as a lower bound for the operational data requirement, since a shift in the distribution is likely to result in flatter learning curves that reach saturation later and at a lower asymptote. Second, while Approach II detects that the distribution of the data collected so far has stabilized, it does not take into account that a portion of the operating state space has never been sampled; its failure mode is therefore to terminate the process too early. We note that, while a random split does correspond to a class of deployment scenarios—namely, those in which operating conditions are not controlled but arise incidentally from the work cycles or environmental conditions within the range covered during acquisition—it does not represent deployment under conditions not included in that range. Since neither approach places any requirement on how the partition of $\mathcal{D}$ is formed—Approach II does not access $\mathcal{D}_{\text{test}}$ at all—an evaluation using a grouped or leave-one-condition-out partition is possible using the same procedure and will be the focus of future work.

Architecture dependence: The accuracy learning curves used as ground truth for both approaches in this paper are obtained with a fixed MLP architecture whose hyperparameters were

adopted from previous work. Since these learning curves serve as the reference against which the approaches are evaluated, the fixed architecture is an evaluation design choice rather than a property of the approaches themselves. However, it does limit the generality of the results reported here, as the shape of the learning curves is architecture-dependent. For example, for more complex models or architectures with different inductive biases, the five fitting functions examined may exhibit varying relative performance. The GNE criterion of Approach II is not affected in itself, as it requires no model training; the accuracy achieved at its stopping point, however, is architecture-dependent as well.

One-dimensional convolutional and recurrent architectures, which are also widely used for diagnosing and predicting the health of engineering systems, would typically be applied to the raw vibration signal (one second sampled at 15.625 kHz) rather than to the ten handcrafted features used here. Both the architecture and the input representation would therefore change, and the two effects cannot be separated a priori. Nevertheless, since such models have to learn from a far higher-dimensional input and since they possess substantially more free parameters, a later onset of saturation is to be expected. Consequently, the unbounded logarithmic function and the power law function, unbounded depending on its parameters, should retain their advantage up to considerably larger p_{fit} , whereas the upper-bounded arctan and exponential functions may saturate prematurely more frequently. For the algebraic root function, unboundedness would become less of a disadvantage, but its limited curvature flexibility—the inability to represent the steep rise and the subsequent flattening simultaneously—would remain decisive as soon as the observed range covers both regimes. In addition, greater run-to-run variance at small n and non-monotonic effects such as sample-wise double descent would result in noisier learning curve points and could violate the monotonicity assumption shared by all five fit functions.

5. CONCLUSION AND OUTLOOK

This paper examines two complementary approaches to determine whether a given dataset is sufficient for training data-driven fault diagnosis models. The evaluation was performed using a dataset for the classification of rolling bearing faults.

The evaluation shows that both approaches are applicable in practice and address different aspects of the same problem. *Approach I: Extrapolation of Learning Curves* enables an explicit estimation of the required amount of training data for a given target accuracy, with reliable results for target accuracies up to approximately 70 %, albeit at the cost of repeated model retraining. *Approach II: Analysis of the Entropy within Data* offers a simple, training-free stopping rule that can be applied online during data acquisition but does not allow for the direct specification of a target accuracy. Both

approaches complement each other: Approach II can monitor the progress of data acquisition in real time, while Approach I provides quantitative estimates of the remaining data requirement when a specific accuracy target must be achieved. Among the remaining key challenges are the sensitivity of the learning curve extrapolation to the choice of fitting function and fitting range, the difficulty of calibrating the GNE threshold to a desired level of accuracy, and the limited generalizability, as both approaches were evaluated using a single dataset.

This paper suggests several ideas for future research. First, both approaches were evaluated using a single dataset. To assess the generalizability of the results, validation using additional datasets from different application domains with varying sizes and dimensions of the feature space is required. Second, the predictive accuracy of Approach I could potentially be improved by imposing parameter constraints on the fitting functions, e.g., enforcing non-negative asymptotes or lower bounds on the saturation degree. By that, the occurrence of implausible extrapolations may be reduced. Third, the validity of the GNE criterion could be strengthened by extending the entropy analysis to multiple features that are evaluated jointly or combined into a unified stopping criterion, rather than relying on a single univariate feature. Finally, the evaluation presented here quantifies in-distribution sufficiency, as discussed in Section 4.5.3. Since both approaches are independent of how $\mathcal{D}_{\text{test}}$ is defined, a grouped or leave-one-condition-out evaluation is possible with the same procedure and would be a valuable addition to future work.

ACKNOWLEDGMENT

This work was supported in part by the Research and Development Projects at Universities of Applied Sciences (Innovative Projects) by the State of Baden-Württemberg, Germany; in part by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under grant no. 514247199; and in part by the German Federal Ministry of Education and Research through the Program “Professorales Personal an Fachhochschulen” (Professorial Staff at Universities of Applied Sciences) under Grant 03FHP115.

We would like to thank Moritz Fischer for his contribution to the preliminary work that formed the basis for the investigations described in this article.

The authors used ChatGPT and Claude during the preparation of this manuscript to assist with language editing and stylistic refinement. All assisted content was subsequently reviewed and revised by the authors, who take full responsibility for the final content of this publication.

REFERENCES

- Arias Chao, M., Kulkarni, C., Goebel, K., & Fink, O. (2022). Fusing physics-based and deep learning models for prognostics. *Reliability Engineering & System Safety*, 217, 107961. doi: 10.1016/j.res.2021.107961
- Bachoc, F., Gamboa, F., Halford, M., Loubes, J.-M., & Risser, L. (2023). Explaining machine learning models using entropic variable projection. *Information and Inference: A Journal of the IMA*, 12(3), 1686–1715. doi: 10.1093/imaia/iaad010
- Braig, M., & Zeiler, P. (2025). A study on using transfer learning to utilize information from similar systems for data-driven condition diagnosis and prognosis. *IEEE ACCESS*, 13, 98485–98503. doi: 10.1109/ACCESS.2025.3576435
- Chen, J., Zhou, D., Guo, Z., Lin, J., Lyu, C., & Lu, C. (2019). An active learning method based on uncertainty and complexity for gearbox fault diagnosis. *IEEE ACCESS*, 7, 9022–9031. doi: 10.1109/ACCESS.2019.2890979
- Crowther, P. S., & Cox, R. J. (2006). Accuracy of neural network classifiers as a property of the size of the data set. In B. Gabrys, R. J. Howlett, & L. C. Jain (Eds.), *Knowledge-based intelligent information and engineering systems* (pp. 1143–1149). Berlin, Heidelberg: Springer. doi: 10.1007/11893011_144
- Estepa, R., Diaz-Verdejo, J. E., Estepa, A., & Madinabeitia, G. (2020). How much training data is enough? a case study for http anomaly-based intrusion detection. *IEEE ACCESS*, 8, 44410–44425. doi: 10.1109/ACCESS.2020.2977591
- Figuerola, R. L., Zeng-Treitler, Q., Kandula, S., & Ngo, L. H. (2012). Predicting sample size required for classification performance. *BMC Medical Informatics and Decision Making*, 12(1), 8. doi: 10.1186/1472-6947-12-8
- Fink, O., Nejjar, I., Sharma, V., Faghih Niresi, K., Sun, H., Dong, H., ... Kesmen, Y. (2026). From physics to machine learning and back: Part ii - learning and observational bias in prognostics and health management (phm). *Reliability Engineering & System Safety*, 274, 112376. doi: 10.1016/j.res.2026.112376
- Guo, Y., Dai, J., & Zhang, J. (2025). Few-shot cross-domain fault diagnosis via adversarial meta-learning. *SCIENTIFIC REPORTS*, 15(1), 41876. doi: 10.1038/s41598-025-25854-z
- Gwon, Y., Hwang, S., Kim, H., Ok, J., & Kwak, S. (2025). Enhancing cost efficiency in active learning with candidate set query. arXiv. doi: 10.48550/arXiv.2502.06209
- Hagmeyer, S., Mauthe, F., & Zeiler, P. (2021). Creation of publicly available data sets for prognostics and diagnostics addressing data scenarios relevant to industrial applications. *International Journal of Prognostics and Health Management*, 12(2), 1–20. doi:

10.36001/ijphm.2021.v12i2.3087

- Hogg, R. V., & Ledolter, J. (1989). *Engineering statistics* (Internat. ed. ed.). New York: MacMillan.
- Jo, D. U., Yun, S., & Choi, J. Y. (2022). How much a model be trained by passive learning before active learning? *IEEE ACCESS*, 10, 34677–34689. doi: 10.1109/ACCESS.2022.3162253
- Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., & Nandi, A. K. (2020). Applications of machine learning to machine fault diagnosis: A review and roadmap. *Mechanical Systems and Signal Processing*, 138, 106587. doi: 10.1016/j.ymssp.2019.106587
- Li, C., Li, S., Feng, Y., Gryllias, K., Gu, F., & Pecht, M. (2024). Small data challenges for intelligent prognostics and health management: a review. *Artificial Intelligence Review*, 57(8). doi: 10.1007/s10462-024-10820-4
- Mahmood, R., Lucas, J., Acuna, D., Li, D., Pillion, J., Alvarez, J. M., ... Law, M. T. (2022). How much more data do i need? estimating requirements for downstream tasks. In *2022 IEEE/CVF conference on computer vision and pattern recognition* (pp. 275–284). Piscataway, NJ: IEEE. doi: 10.1109/CVPR52688.2022.00037
- Mahmood, R., Lucas, J., Alvarez, J. M., Fidler, S., & Law, M. T. (2022). Optimizing data collection for machine learning. In *Proceedings of the 36th international conference on neural information processing systems*. Red Hook, NY, USA: Curran Associates Inc.
- Paninski, L. (2003). Estimation of entropy and mutual information. *Neural Computation*, 15(6), 1191–1253. doi: 10.1162/089976603321780272
- Raj, K. K., Kumar, S., & Kumar, R. R. (2025). Systematic review of bearing component failure: Strategies for diagnosis and prognosis in rotating machinery. *Arabian Journal for Science and Engineering*, 50(8), 5353–5375. doi: 10.1007/s13369-024-09866-x
- Wang, B., Lei, Y., Li, N., & Li, N. (2020). A hybrid prognostics approach for estimating remaining useful life of rolling element bearings. *IEEE Transactions on Reliability*, 69(1), 401–412. doi: 10.1109/TR.2018.2882682
- Xie, Y., Ding, M., Tomizuka, M., & Zhan, W. (2023). Towards free data selection with general-purpose models. *Advances in Neural Information Processing Systems*, 36, 1309–1325.
- Zio, E. (2022). Prognostics and health management (phm): Where are we and where do we (need to) go in theory and practice. *Reliability Engineering & System Safety*, 218, 108119. doi: 10.1016/j.ress.2021.108119

BIOGRAPHIES

Moritz Neu received the B.Eng. degree in Mechatronics Engineering from the University of Applied Sciences Esslingen,

Germany, and is currently studying in the degree program Mechatronics / Systems Engineering (M.Eng.) at Aalen University of Applied Sciences and Esslingen University of Applied Sciences, Germany.

Marcel Braig received the B.Sc. degree in technology development from the University of Applied Sciences Ravensburg-Weingarten (RWU), Germany, and the joint M.Eng. degree in mechatronics/systems engineering from Aalen University of Applied Sciences and Esslingen University of Applied Sciences, Germany. He is currently pursuing the Ph.D. degree in cooperation with the University of Stuttgart. He is a Research Assistant with the Institute for Technical Reliability and Prognostics (IZP), Esslingen University of Applied Sciences. His research interests include diagnostics and prognostics, condition monitoring, data-driven methods, and machine learning, with a focus on application to similar systems.

Fabian Mauthe received the B.Sc. degree in Mechanical Engineering and Mechatronics and the M.Sc. degree in Mechanical Engineering from Furtwangen University of Applied Sciences, Germany. He is currently pursuing the Ph.D. degree in cooperation with the University of Stuttgart. He is also working as a Research Assistant with the Institute for Technical Reliability and Prognostics (IZP), Esslingen University of Applied Sciences. His research interests include diagnostics and prognostics, condition monitoring, data-driven methods, and hybrid methods.

Peter Zeiler received the Dipl.-Ing. and Dr.-Ing. degrees in mechanical engineering from the University of Stuttgart. He worked as a Team Leader of Research and Development at Robert Bosch GmbH. Thereafter, he became the Head of the Department of Reliability Engineering and the Department of Drive Technology, Institute of Machine Components, University of Stuttgart. In 2017, he joined the Esslingen University of Applied Sciences, Germany. Since 2021, he has been associated with the Faculty of Engineering Design, Production Engineering and Automotive Engineering, University of Stuttgart. He is currently a Professor with the Faculty of Mechanical and Systems Engineering, Esslingen University of Applied Sciences, where he is the director of the Institute for Technical Reliability and Prognostics (IZP). His research interests include prognostics and health management, methods for calculating reliability under operating conditions, modeling and simulation of reliability, and availability for complex systems. He is committed to Verein Deutscher Ingenieure (VDI) (English: Association of German Engineers). Then, he is the Chairperson of the Safety and Reliability Advisory Board, and the Technical Committees for Prognostics and Health Management and Monte Carlo Simulation. Furthermore, he is the Conference Chair of the German Conference Fachtagung Technische Zuverlässigkeit (TTZ) (English: Technical Reliability Conference).

APPENDIX A

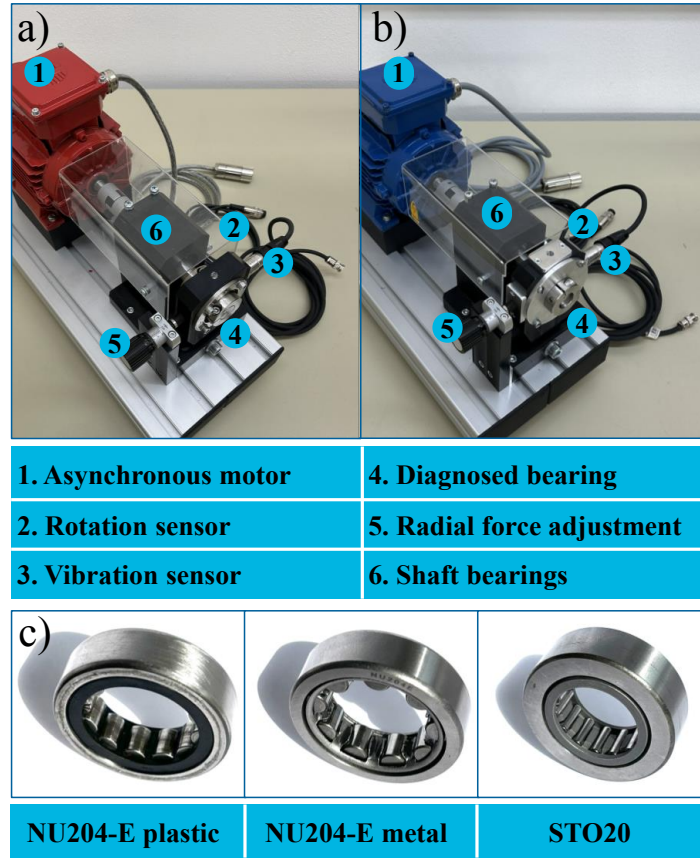


Figure A1. The setup used to generate the measurement data, including two rolling bearing test rigs (a) and (b), and different types of rolling bearings used (c). The shaft position and vibration acceleration signals are recorded, along with the measuring time. The speed is calculated from the position and time values.

Table A1. Configurations used to record the rolling bearing measurement dataset. For each combination of bearing types, faults, speeds, and loads, ten recordings over one second are logged.

Bearing type ⁽¹⁾	Fault ⁽²⁾	Speed / min ⁻¹ (3)	Load ⁽⁴⁾
NU204-E (a, plastic, 12, cylinder, 7.5, 20, 47)	OK / OR / IR / RE	500 / 1000	1 / 2
NU204-E (b, plastic, 12, cylinder, 7.5, 20, 47)	OK / OR / IR / RE	500 / 1000	1 / 2
NU204-E (b, metal, 9, cylinder, 7.5, 20, 47)	OK / IR / RE	500 / 1000	1 / 2
STO20 (b, metal, 14, needle, 3.5, 20, 47)	OK / OR / IR / RE	500 / 1000	1 / 2

⁽¹⁾(test rig in Fig. A1, cage material, number of rolling elements, rolling element type, diameter of rolling elements / mm, inner diameter / mm, outer diameter / mm); ⁽²⁾ OK=healthy, OR=outer race fault, IR=inner race fault, RE=rolling element fault; ⁽³⁾ Controlled set speed; ⁽⁴⁾ Specified in the virtual unit “load level”. The load is set using a micrometer screw and a rubber spring element.

APPENDIX B

Table B1. Evaluation values of $\Delta p_{\text{Acc}_{\text{req}}}$ and the number of evaluated learning curves for Evaluation I-2 of Approach I.

Acc _{req}	0.60			0.65			0.70			0.75			0.80		
	p _{fit}	0.05	0.10	0.15	0.05	0.10	0.15	0.05	0.10	0.15	0.05	0.10	0.15	0.05	0.10
Function	Median														
Algebraic root	0.00	-0.01	-0.02	0.00	0.00	0.00	0.02	0.01	0.02	0.08	0.06	0.06	0.28	0.26	0.24
Arctan	0.01	0.00	-0.01	0.01	0.01	0.00	0.04	0.02	0.01	0.11	0.08	0.05	0.33	0.29	0.23
Exponential	0.01	0.00	-0.01	0.02	0.01	0.00	0.04	0.02	0.01	0.12	0.08	0.05	0.34	0.28	0.23
Logarithmic	-0.03	-0.07	-0.10	-0.03	-0.05	-0.06	-0.03	-0.04	-0.04	0.01	-0.01	-0.01	0.17	0.12	0.11
Power law	-0.03	-0.06	-0.09	-0.03	-0.05	-0.06	-0.03	-0.04	-0.04	0.01	-0.01	0.00	0.17	0.13	0.11
Function	Width of the 5th–95th percentile interval														
Algebraic root	0.35	0.28	0.40	0.41	0.29	0.28	0.49	0.31	0.26	0.70	0.51	0.44	0.92	0.76	0.69
Arctan	0.27	0.25	0.37	0.29	0.22	0.36	0.35	0.28	0.42	0.55	0.46	0.59	0.79	0.74	0.81
Exponential	0.23	0.14	0.18	0.25	0.16	0.19	0.32	0.21	0.22	0.53	0.40	0.40	0.77	0.72	0.69
Logarithmic	0.56	0.58	0.49	0.62	0.54	0.51	0.69	0.58	0.52	0.87	0.74	0.63	1.06	0.98	0.87
Power law	0.55	0.57	0.49	0.63	0.53	0.51	0.69	0.58	0.52	0.88	0.74	0.63	1.06	0.98	0.87
Function	5th percentile														
Algebraic root	-0.30	-0.24	-0.34	-0.33	-0.23	-0.21	-0.34	-0.19	-0.14	-0.32	-0.16	-0.09	-0.17	-0.06	-0.02
Arctan	-0.21	-0.20	-0.30	-0.20	-0.15	-0.29	-0.18	-0.14	-0.29	-0.13	-0.09	-0.24	-0.01	-0.01	-0.12
Exponential	-0.17	-0.09	-0.12	-0.16	-0.09	-0.12	-0.15	-0.08	-0.11	-0.10	-0.04	-0.07	0.01	0.02	-0.01
Logarithmic	-0.52	-0.55	-0.45	-0.56	-0.50	-0.47	-0.57	-0.51	-0.44	-0.56	-0.48	-0.38	-0.40	-0.37	-0.30
Power law	-0.51	-0.54	-0.45	-0.56	-0.49	-0.46	-0.57	-0.50	-0.44	-0.56	-0.48	-0.37	-0.39	-0.37	-0.30
Function	25th percentile														
Algebraic root	-0.05	-0.06	-0.06	-0.05	-0.05	-0.05	-0.04	-0.03	-0.02	0.00	0.00	0.01	0.11	0.11	0.10
Arctan	-0.02	-0.04	-0.08	-0.02	-0.02	-0.05	0.00	-0.01	-0.03	0.05	0.02	0.00	0.18	0.15	0.08
Exponential	-0.02	-0.02	-0.04	-0.01	-0.02	-0.03	0.01	0.00	-0.02	0.05	0.02	0.01	0.19	0.15	0.09
Logarithmic	-0.12	-0.17	-0.19	-0.14	-0.16	-0.18	-0.16	-0.15	-0.13	-0.15	-0.13	-0.09	-0.02	-0.04	-0.02
Power law	-0.12	-0.16	-0.19	-0.14	-0.16	-0.17	-0.16	-0.15	-0.12	-0.15	-0.13	-0.09	-0.01	-0.04	-0.02
Function	75th percentile														
Algebraic root	0.02	0.01	0.01	0.03	0.02	0.03	0.06	0.05	0.05	0.17	0.14	0.14	0.48	0.44	0.42
Arctan	0.02	0.02	0.02	0.04	0.03	0.03	0.08	0.06	0.04	0.20	0.16	0.13	0.53	0.47	0.42
Exponential	0.02	0.01	0.01	0.04	0.03	0.02	0.09	0.05	0.04	0.21	0.15	0.12	0.53	0.47	0.41
Logarithmic	0.01	-0.01	-0.02	0.01	0.00	-0.01	0.03	0.01	0.01	0.11	0.07	0.06	0.38	0.31	0.29
Power law	0.01	-0.01	-0.02	0.01	0.00	0.00	0.03	0.01	0.01	0.12	0.07	0.07	0.38	0.31	0.30
Function	95th percentile														
Algebraic root	0.05	0.04	0.07	0.08	0.06	0.07	0.15	0.12	0.12	0.38	0.35	0.35	0.75	0.70	0.67
Arctan	0.06	0.05	0.08	0.09	0.07	0.07	0.17	0.14	0.13	0.42	0.37	0.35	0.78	0.73	0.69
Exponential	0.06	0.05	0.06	0.09	0.07	0.07	0.17	0.13	0.11	0.43	0.36	0.33	0.78	0.74	0.68
Logarithmic	0.04	0.03	0.04	0.06	0.04	0.04	0.12	0.07	0.08	0.31	0.26	0.25	0.66	0.61	0.57
Power law	0.04	0.03	0.04	0.07	0.04	0.05	0.12	0.08	0.08	0.32	0.26	0.26	0.67	0.61	0.57
Function	Number of evaluated learning curves														
Algebraic root	7815	1602	186	9672	4636	1047	10218	8882	4449	10095	11190	9510	8178	9471	9513
Arctan	6677	1188	110	8057	3384	657	8384	6477	2844	8234	7888	5903	6714	6506	5431
Exponential	4076	618	47	5022	1848	320	5256	3613	1482	5153	4367	3197	4166	3417	2817
Logarithmic	6536	1413	161	7785	4130	965	7725	7907	4149	7069	9524	8820	5134	7468	8466
Power law	6513	1382	159	7770	4059	947	7718	7780	4053	7073	9403	8633	5152	7395	8309

APPENDIX C

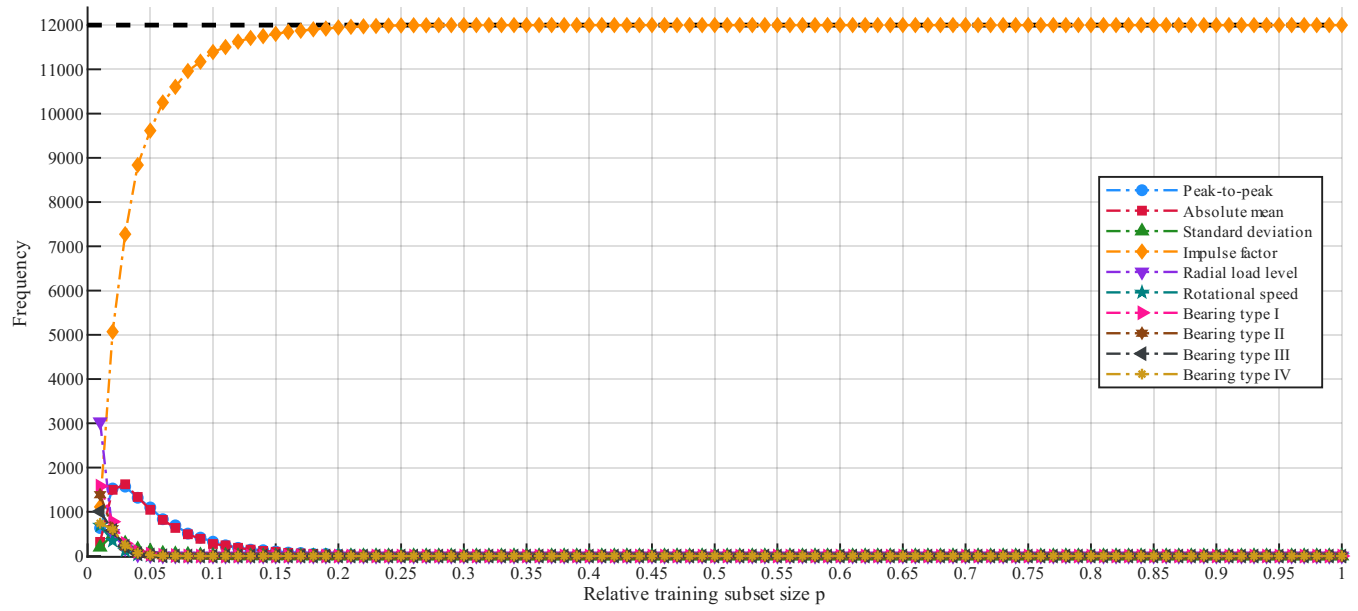


Figure C1. Frequency with which each of the ten features is selected as the most discriminative feature by the one-way ANOVA F -test, evaluated separately for every relative training subset size p and over all $k = 12,000$ repetitions.

APPENDIX D-1

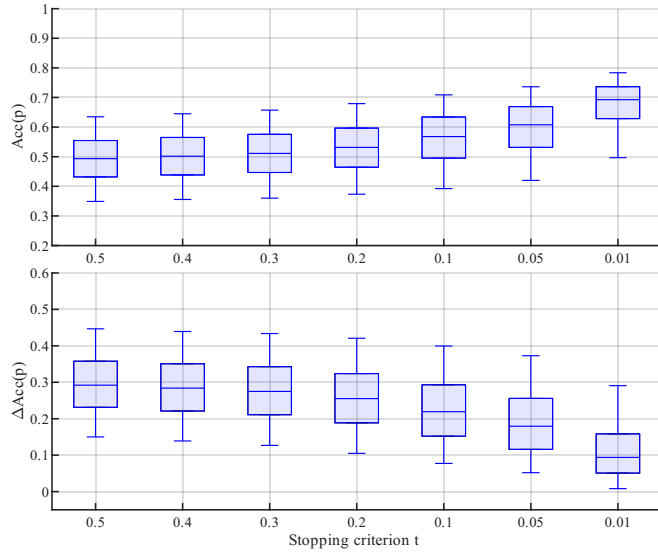
Smoothing order $o = 1$ 

Figure D1. The learning curve accuracies $Acc(p)$ at the $p_{i_{sat}}$ determined by the GNE using the threshold value t and for a smoothing order of $o = 1$ over all $k = 12,000$ repetitions, as well as the corresponding accuracy deviation $\Delta Acc(p)$ to the maximum accuracy achievable with the full training set. The 5th, 25th, 50th, 75th, and 95th percentiles are shown.

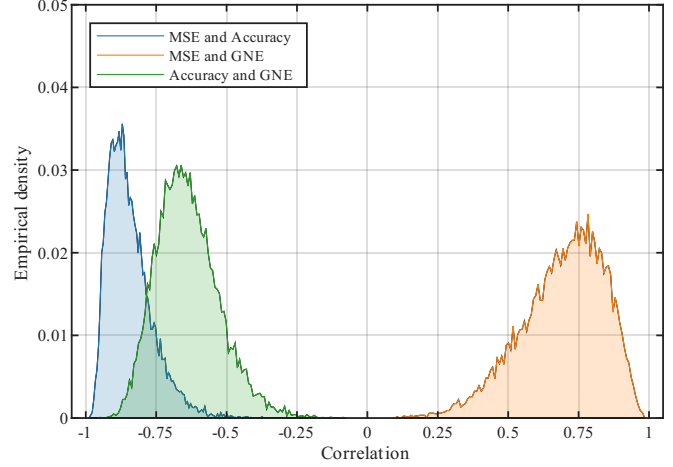


Figure D2. Correlation between accuracy, MSE, and GNE with rising training subset size over all $k = 12,000$ repetitions and using a smoothing order of $o = 1$.

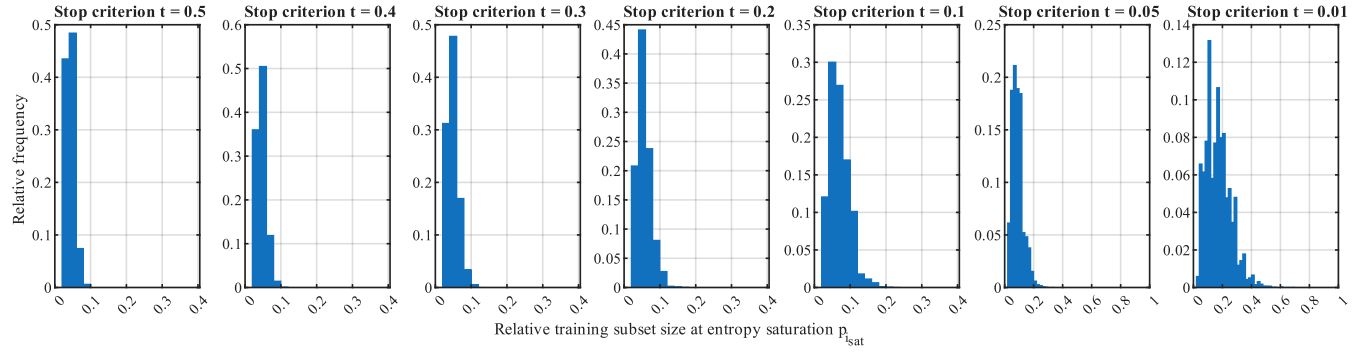


Figure D3. Distribution of $p_{i_{sat}}$ across all $k = 12,000$ repetitions for different threshold values t and for a smoothing order of $o = 1$.

APPENDIX D-2

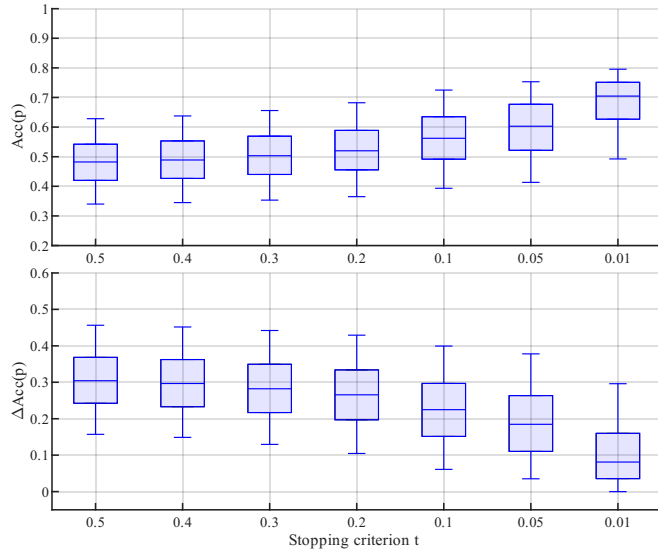
Smoothing order $o = 10$ 

Figure D4. The learning curve accuracies $Acc(p)$ at the $p_{i_{sat}}$ determined by the GNE using the threshold value t and for a smoothing order of $o = 10$ over all $k = 12,000$ repetitions, as well as the corresponding accuracy deviation $\Delta Acc(p)$ to the maximum accuracy achievable with the full training set. The 5th, 25th, 50th, 75th, and 95th percentiles are shown.

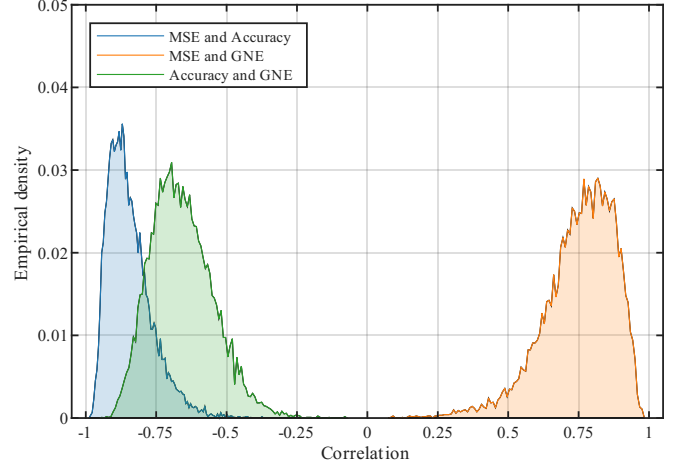


Figure D5. Correlation between accuracy, MSE, and GNE with rising training subset size over all $k = 12,000$ repetitions and using a smoothing order of $o = 10$.

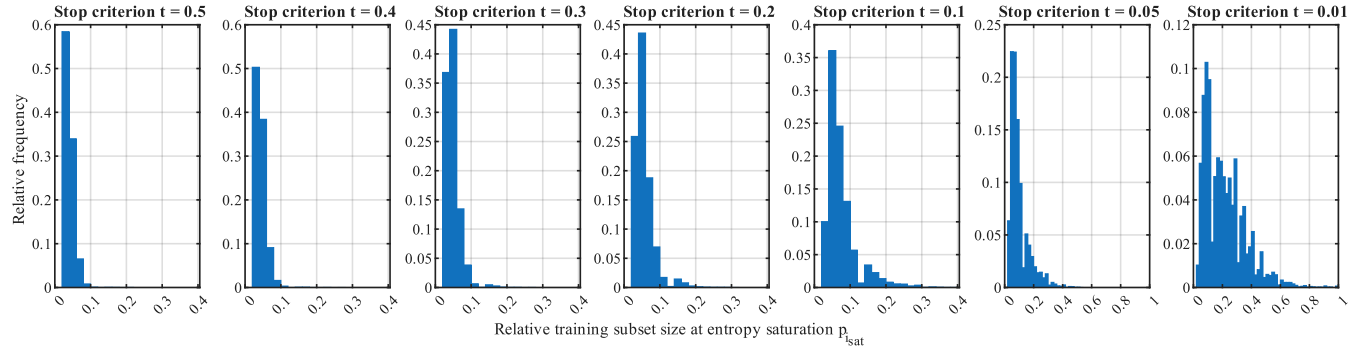


Figure D6. Distribution of $p_{i_{sat}}$ across all $k = 12,000$ repetitions for different threshold values t and for a smoothing order of $o = 10$.

APPENDIX D-3

GNE without entropy smoothing

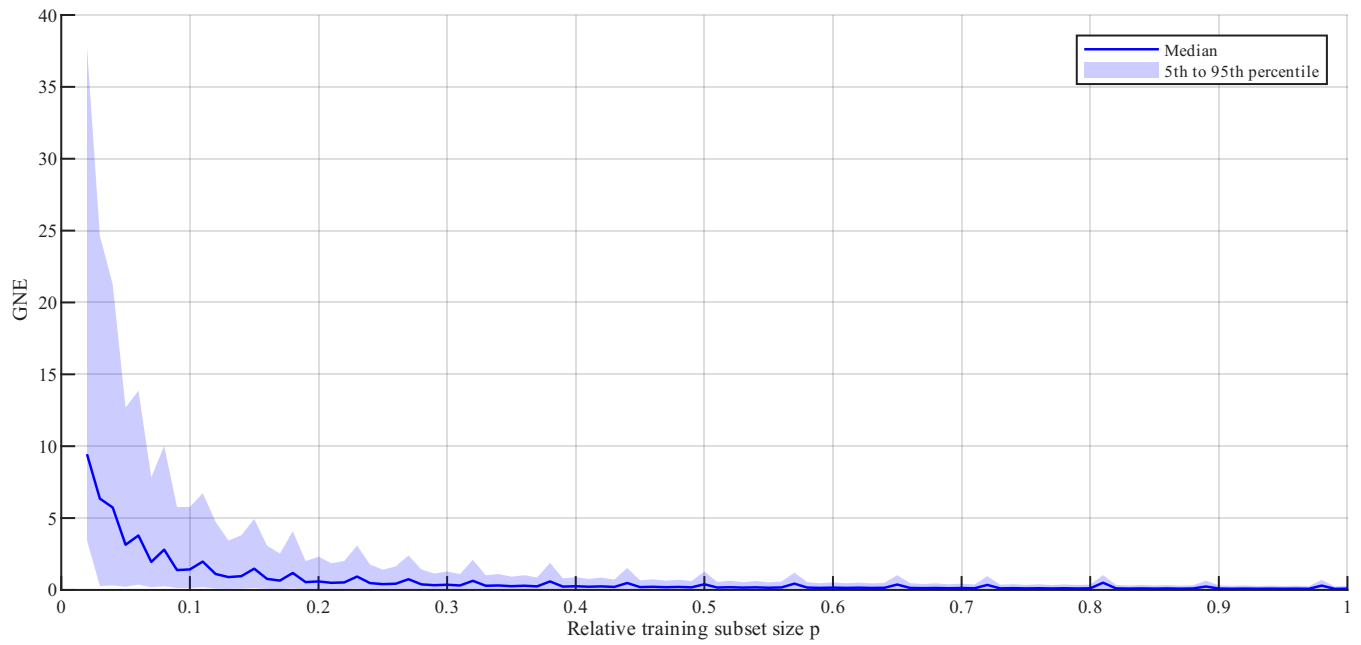


Figure D7. The median as well as the 5th to 95th percentile interval of the resulting GNE curves over all $k = 12,000$ repetitions, determined using the entropy curves without smoothing, i.e., $\sigma = 1$.