**ORIGINAL ARTICLE**

# Application of machine learning for fleet-based condition monitoring of ball screw drives in machine tools

Berend Denkena[1] · Marc-André Dittrich[1] · Hendrik Noske[1] · Dirk Lange[2] · Carolin Benjamins[3] · Marius Lindauer[3]

**Abstract**

Ball screws are frequently used as drive elements in the feed axes of machine tools. The failure of ball screw drives is associated with high downtimes and costs for manufacturing companies, which harm competitiveness. Data-based monitoring approaches derive the ball screw condition based on sensor data in cases where no knowledge is available to derive a physical model-based approach. An essential criterion for selecting the condition assessment method is the availability of fault data. In the literature, fault patterns are often artificially created in an experimental test bench scenario. This paper presents ball screw drive monitoring approaches for machine tool fleets based on machine learning. First, the potentials of automated machine learning for supervised anomaly detection are investigated. It is shown that the AutoML tool Auto-Sklearn achieves a higher monitoring quality compared to literature approaches. However, fault data are often not available. Therefore, unified outlier scores are applied in a semi-supervised anomaly detection mode. The unified outlier score approach outperforms threshold-based approaches commonly used in industry. The considered data set originates from a machine tool fleet used in series production in the automotive industry collected over 8 months. Within the observation period, multiple ball screw failures are observed so that sensor data about the transient phases between normal and fault conditions is available.

**Keywords** Condition monitoring · Machine learning · Ball screw · Failure

## 1 Introduction

### 1.1 Need for condition monitoring of ball screw drives in machine tools

Machine tool feed drives are used for high-precision positioning of the milling tool and workpiece. Ball screw drives are suitable for this task due to their high-efficiency level [1, 2]. Ball screws also exhibit low heating and length variation and high positioning accuracy [3]. Ball screws also have a low failure frequency. However, in case of failure, high downtime follows, reducing machine tools' technical availability. A total of 38% of the downtimes of feed axes are caused by ball screws and feed axes, accounting for nearly 40% of the leading causes of machine tool failure [3]. A ball screw drive consists of multiple components, including a raceway, ball screw, screw nut, drive motor, support bearings, and the table. The ball screw is subjected to preloading to increase rigidity [4]. Various types of ball screw damage exist. In the case of sudden early damage, running instability occurs due to damage sustained by the deflection elements resulting in defects of balls and the raceways. Gradual late damage occurs in ball screws used for longer than the intended operating time. In this case, pitting is created in the raceway and ball surfaces, leading to running irregularities. Another type of damage is the insidious loss of preload. Over time, the ball diameter decreases, reducing the preload and, thus, the stiffness properties of the drive. The stiffness variations increase the chatter tendency of the axis, and thereby surface tolerances of workpieces can no longer be maintained [5]. Additionally, ball screws exhibit higher wear than linear drives due to their higher friction component [2]. If the wear exceeds 80%, the ball screw is irreparable and must be replaced. If a ball screw is repaired in time, 30–50% of the replacement costs can be saved

✉ Hendrik Noske
    noske@ifw.uni-hannover.de

[1]  Institute of Production Engineering and Machine Tools, An der Universität 2, 30823 Garbsen, Germany

[2]  Marposs Monitoring Solutions GmbH, Buchenring 40, 21272 Egestorf, Germany

[3]  Institute of Artificial Intelligence, Appelstraße 9a, 30167 Hannover, Germany

[6]. Due to the diversity of wear and fluctuating operating parameters (temperature, load, lubrication, etc.), predicting the operating time of ball screws is difficult [2].

Condition monitoring is used to reduce downtimes and high replacement costs of machine components and thus increase the availability of machine tools [7]. In addition, condition monitoring can assist in optimizing maintenance activities [4]. Condition monitoring approaches are divided into model-based and data-based approaches. Model-based approaches include physical models, and classical AI approaches like expert systems. Physical models comprise approaches based on parameter estimates, which use estimation methods and differential equations to determine the model parameters. Data-based approaches learn the system behavior automatically based on past data. This group includes machine learning methods such as artificial neural networks used as classifiers. In addition, machine learning methods are used to output an outlier score if fault data is unavailable (semi-supervised anomaly detection) [8]. In contrast to threshold-based approaches (also called limit-value based) which allow fault detection, machine learning methods can be used for fault diagnosis. This requires that information about different types of faults is available [9].

## 1.2 Our contribution

This work presents a ball screw drive monitoring approach for machine tool fleets based on machine learning. An industrial data set of a machine tool fleet (monitoring data of 13 five-axis machine tools MAG SPECHT 600 collected over 8 months) used in series production in the automotive industry is considered. Within the monitoring period under consideration, the ball screw drives of the Z-axis are replaced on 4 machines. The distinctive feature of the data set is that information about the transition between normal and faulty conditions is apparent in three ball screw drives. In the literature, anomalies are often artificially generated in an experimental test bench scenario. There is usually no data available that (a) describes the entire life cycle of the ball screws in industrial practice and (b) describes the transition phase between normal and faulty conditions. These approaches also neglect the fact that the normal state of the machines changes over time. For this reason, an in-depth analysis of the monitoring signals in the normal and faulty condition of ball screws of 13 five-axis machine tools MAG SPECHT 600 is performed.

In the past, many researchers used machine learning classifiers for condition monitoring of ball screw drives [10–16]. This approach can be followed when fault data is available (supervised anomaly detection). These studies arbitrarily select the methods at the respective stages

of data and feature preprocessing, dimensionality reduction, and classification. Often, it is not shown to what extent the model hyperparameters, e.g., how to configure the method, are optimized. In this context, automated machine learning (AutoML) offers the possibility to systematically support the practical user in selecting methods at the respective stages [17]. In addition, past studies have shown that AutoML tools like Auto-Sklearn achieve better classification results on average through ensemble building and meta-learning [18]. However, the potential for performance improvements of Auto-ML tools for ball screw condition monitoring has not been investigated to date. In this paper, a methodology for supervised anomaly detection using Auto-Sklearn is developed for ball screw drive monitoring in machine tool fleets. The proposed method is able to detect fault states of ball screw drives, and because of the generality of AutoML, it is not restricted to the machine types monitored in this paper.

Supervised anomaly detection methods are only applicable when sufficient fault data is available. For this reason, a semi-supervised anomaly detection approach is applied and evaluated. A so-called baseline model is created based on data describing the normal state of ball screws. The baseline model produces a unified outlier score to perform condition assessment. The monitoring quality of the unified outlier score approach outperforms threshold-based approaches commonly used in industry.

The paper is organized as follows: Chapter 2 presents the related work in machine learning based ball screw drive condition monitoring. The data set is described in Chapter 3. In Chapter 4, the monitoring methodologies are introduced. The results of the experimental study are presented in Chapter 5.

## 2 Related work on monitoring approaches of ball screw drives based on machine learning

Usually, machine axes are evaluated via a test cycle executed intermittently during the manufacturing process. To ensure robust monitoring, the influence of any sources of interference must be avoided. One source of interference is the manufacturing process. During the process, process parameters and the workpiece mass change within metal-cutting manufacturing processes. Consequently, the monitoring signals change regardless of the ball screw drive condition. For this reason, the monitoring signals are recorded during the process-free time in a predefined test cycle [2].

Anomalies are often artificially generated in recent studies to evaluate monitoring approaches. Jin et al. and Denkena et al. use different ball sizes to simulate the preload loss [10, 11]. Emilia et al. induce defects on the

running surface of the ball screw with the laser powder cladding method [12]. Feng and Pan use a double-nut system to vary the preload [13]. Benker et al. use two ball screws with different levels of preload [14]. Balaban et al. block the return channel with a detached piece of insulation. Additionally, the backlash is simulated using undersized balls and spalling defects on the ball screw are generated using electro-discharge machining [15]. Li et al. use different wear levels of ball screws acquired from an industrial partner [16]. An overview of the faults considered as well as the internal and external sensors used, is given by Butler et al. [4].

To detect anomalies, a distinction is made between two different procedures in condition monitoring: In the context of semi-supervised anomaly detection, it is assumed that only data describing the normal state is available [19]. For example, control charts based on $T^2$ and $Q$-statistics, as well as the Mahalanobis-distance, have already been used for ball screw monitoring [20, 21]. In contrast, supervised anomaly detection uses fault classes in conjunction with a classifier that distinguishes between normal and fault states [19]. Table 1 gives an overview of supervised anomaly detection approaches of ball screw drives. Jin et al. apply various methods such as Gaussian Mixture Models, Self-Organizing-Maps, and the Mahalanobis-distance in a supervised mode for ball screw monitoring based on vibration and temperature data. The presented methods output a health index based on extracted features to evaluate the machine component's health. The authors show that the health indices correlate with the anomalies such as lack of lubrication and preload loss. Suitable features for classification are identified using the Fisher-score [10]. Benker et al. use Gaussian Process Classification to classify different preload levels [11, 14]. Li et al. employ a support vector machine to classify the condition of ball screws. Sensor data from the machine control, such as torque, and data from three accelerometers are given. Relevant features are preselected in the first step using the Fisher-score. Furthermore, only a small subset of the preselected features is used for classification by sequential forward selection. The authors show that torque is more suitable for classifying the ball screw condition than vibration signals [16]. Feng and Pan develop a low-cost sensor system to collect temperature and vibration data for ball screw monitoring. Support Vector Machines are applied to classify different preload levels [13]. Emilia et al. present an approach for ball screw monitoring based on vibration and acoustic emission data. A Naive-Bayes classifier and a K-Nearest Neighbor classifier are employed to classify different states. The authors obtained improved results using vibration data compared to acoustic emission data [12]. Denkena et al. use the F-score and the principal component analysis (PCA) for feature selection and feature extraction. It is shown that the position error is more suitable for the classification of different preload levels than the acceleration signal data [11]. Schmidt et al. performed condition monitoring using a so-called ball-bar measurement. This method is used to determine the positioning accuracy of the machine tool. In total, data from 32 ball screws, including 145 measurements, are used. A K-Nearest Neighbor model is applied for classification. However, the data set is not described in detail [22]. Other authors use deep learning methods for ball screw monitoring, such as convolutional neural networks [23–26]. In the literature, there is often no comparison with "simpler" classifiers when deep learning methods are applied.

As described earlier, there is no systemic nature to the previously described work on supervised anomaly detection concerning method selection. It is rarely described why a specific method is selected for data and feature preprocessing and classification. Therefore, using an AutoML tool to create the model pipeline to predict ball screw conditions is a systematic and replicable approach. AutoML tools are increasingly being applied in the manufacturing context. For example, ML-Plan-RUL, presented by Tornede et al., allows for predicting machines'

**Table 1** Overview of machine learning approaches for supervised anomaly detection of ball screw drives

| Ref | Data splitting mode | Hyperparameter optimization | Data preprocessing / Dim. reduction | Classifier |
|-----|---------------------|-----------------------------|-------------------------------------|-----------|
| [10] | Hold out (train, test) | None | Feature selection (Fisher-Score) | Self-organizing map, Gaussian mixture models, Mahalanobis distance |
| [11] | Hold out (train, test) | None | Normalization | Decision Tree |
| [12] | Hold out (train, test) | None | None | Naïve Bayes, K-Nearest-Neighbors |
| [13] | Hold out (train, test) | 2 Kernels of SVC | None | Support Vector Machine |
| [14] | None | Maximum likelihood | None | Gaussian Process Classification |
| [15] | Hold out (train, test) | None | Feature scaling (Standardization) | Feed Forward Neural Network |
| [16] | Cross validation | None | Feature scaling (Standardization), feature selection (Fisher-Score, Sequential Forward Selection) | Support Vector Machine |
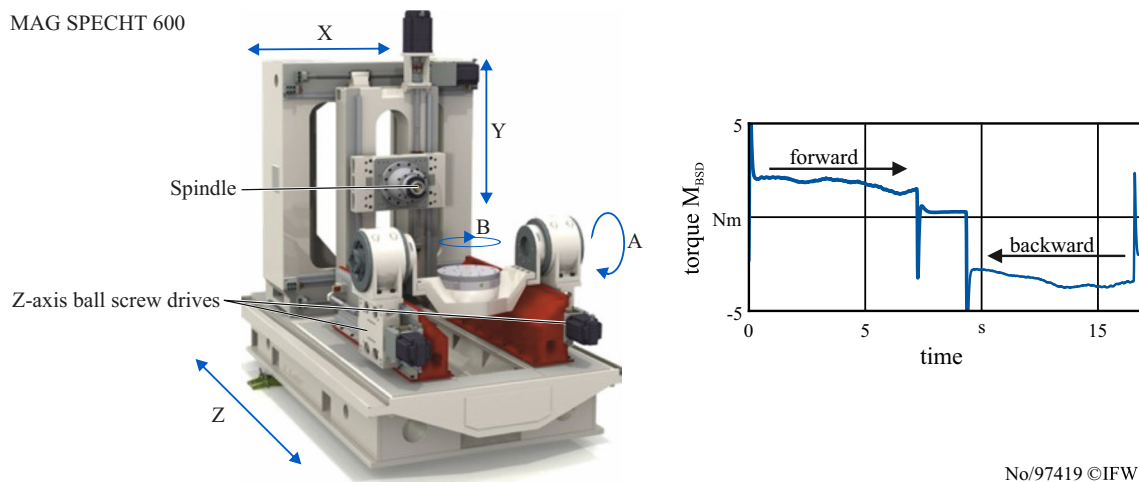
**Fig. 1** Axis kinematics and torque signal of the test cycle (Z-axis)

remaining useful life (RUL) for predictive maintenance [27]. For predicting the shape error for pocket milling operations in process planning, Denkena et al. use Auto-Sklearn [28]. Auto-Sklearn is also used by Kißkalt et al. to predict tool wear during lot milling [29].

In contrast to literature approaches, data from a machine tool fleet are available in this work. Fleet-based condition monitoring assumes data from several identical machines or machine components are available. This increases the probability that failures of machine components occur in an observation period and thus that fault data are available. In addition, the question arises as to whether monitoring can be improved using data from other machines. Fleet-based monitoring approaches can be found in the literature focusing on specific machine components. For instance, Hendrickx et al. [30] develop a clustering-based condition monitoring approach for electrical drivetrain fleets. However, literature

on ball screw drive monitoring in machine tool fleets that include long-term datasets is missing.

## 3 Data set description and analysis

The data set is collected from 13 five-axis machine tools of the type MAG Specht 600, recorded over more than 8 months. These machines are used in the automotive industry, where the Z-axis is heavily stressed. After the production of a lot, an identical test cycle of the Z-axis is performed. The machine's axis kinematics and the Z-axis's torque in normal condition from a test cycle are shown in Fig. 1. For each machine, the Z-axis torque $M_{BSD}$ is recorded at a sampling frequency of 100Hz via the machine control. In addition, the data from a 3-axis acceleration sensor $Acc_{1-3}$ from Marposs Monitoring Solution GmbH (Artis) is recorded,
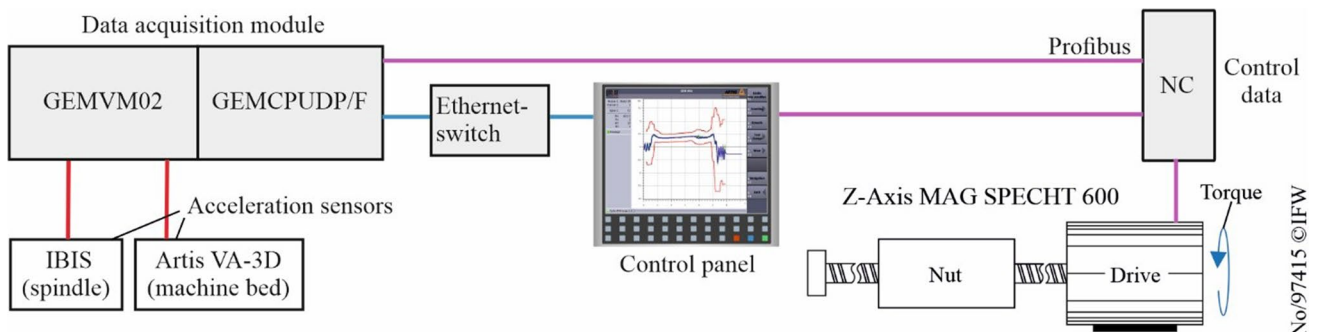


**Fig. 2** Measuring setup for ball screw drive monitoring

which is attached to the machine bed. Another acceleration sensor *Spi* is mounted on the tool spindle. The acceleration sensor *Spi* is originally installed for spindle monitoring. The acceleration sensors are connected to an industrial PC which stores the signal data for each test cycle. The industrial PC accesses the machine control data via Profibus. In addition, the test cycle data can be visualized via a control panel at the machine. The measuring setup is depicted in Fig. 2.

The sensor data is available as discrete time series $x_{c,b,m}(t)\epsilon R$ for $c\epsilon\{1,\dots,n_b\}$ test cycles, $t\epsilon\{1,\dots,T\}$ time steps, and $m\epsilon\{1,\dots,M\}$ machine tools. The parameter $n_b$ represents the number of test cycles for $b\epsilon\{1,\dots,B\}$ ball screw drives. It should be noted that in the fault cases, two ball screw drives are assigned to one machine. Table 2 shows the number of test cycles with and without anomalies of the respective ball screws during the observation period. The numbering of the ball screws corresponds to the respective machine tool in which the ball screw is installed. A total of 1540 test cycles are performed for 13 identical machines. For a total of 4 machines, a ball screw drive is replaced during the observation period. The ball screw drives are replaced due to tolerance deviations concerning the manufactured products. The ball screws used before disassembly are marked "pre" in the table.

The first step involves analyzing the fault patterns of the monitoring signals in the time and frequency domains that occur before ball screw disassembly. In the case of three ball screws, test cycles are available that describe the transition between normal and faulty conditions (Bs7-pre, Bs11-pre, Bs13-pre). In the case of ball screw

Bs12-pre, it is noted that an advanced state of degradation is already present at the beginning of data acquisition. For ball screws Bs-11-pre and Bs-12-pre, damage to the raceways is detected after disassembly. In contrast, worn-out balls have been the root cause of failure in the case of ball screw Bs13-pre. No condition changes are detected for the newly replaced ball screws (Bs7-post, Bs11-post, Bs12-post, Bs13-post).

Figure 3 illustrates the segmented torque of the Z-axis $M_{BSD}$ and the accelerometer signals ($Acc_{1-3}$, $Spi$) for different degradation levels. The time series are segmented in such a way that only the segments in the forward direction with constant feed are considered. These fixed segments are selected based on expert knowledge. In the case of ball screw Bs7-pre, no significant changes in the torque signal $M_{BSD}$ are observed after the anomaly starts. In contrast to the observations of Lia et al. [16], this means that the internal control sensor signals are not sufficient for robust monitoring of ball screw drives in machine tools. In the case of ball screws Bs11-pre, Bs12-pre, and Bs13-pre, higher frequencies occur in thee torque signal $M_{BSD}$ at the start of the anomaly. For each faulty ball screw, changes in the accelerometer signals are visible when the abnormality occurs. In the case of ball screw Bs14-pre, signal $Acc_2$ is shown since no significant changes are visible in signal $Acc_1$. Therefore, it is concluded that the acceleration signals of the triaxial accelerometer should be evaluated in each direction. In the case of ball screw Bs11-pre, more significant peaks initially appear in the acceleration signal $Acc_1$ at the

**Table 2** Overview of the number of test cycles performed during the observation period

| Ball screw name | Number of recorded test cycles | Number of test cycles without anomalies | Number of test cycles containing anomalies |
|---|---|---|---|
| Bs1 | 123 | 123 | 0 |
| Bs2 | 127 | 127 | 0 |
| Bs3 | 129 | 129 | 0 |
| Bs4 | 126 | 126 | 0 |
| Bs5 | 124 | 124 | 0 |
| Bs6 | 125 | 125 | 0 |
| Bs7-pre | 109 | 102 | 7 |
| Bs7-post | 30 | 30 | 0 |
| Bs8 | 116 | 116 | 0 |
| Bs9 | 100 | 100 | 0 |
| Bs10 | 142 | 142 | 0 |
| Bs11-pre | 95 | 42 | 53 |
| Bs11-post | 28 | 28 | 0 |
| Bs12-pre | 74 | 0 | 74 |
| BS12-post | 24 | 24 | 0 |
| Bs13-pre | 52 | 18 | 34 |
| Bs13-post | 16 | 16 | 0 |
| Total | 1.540 | 1.372 | 168 |
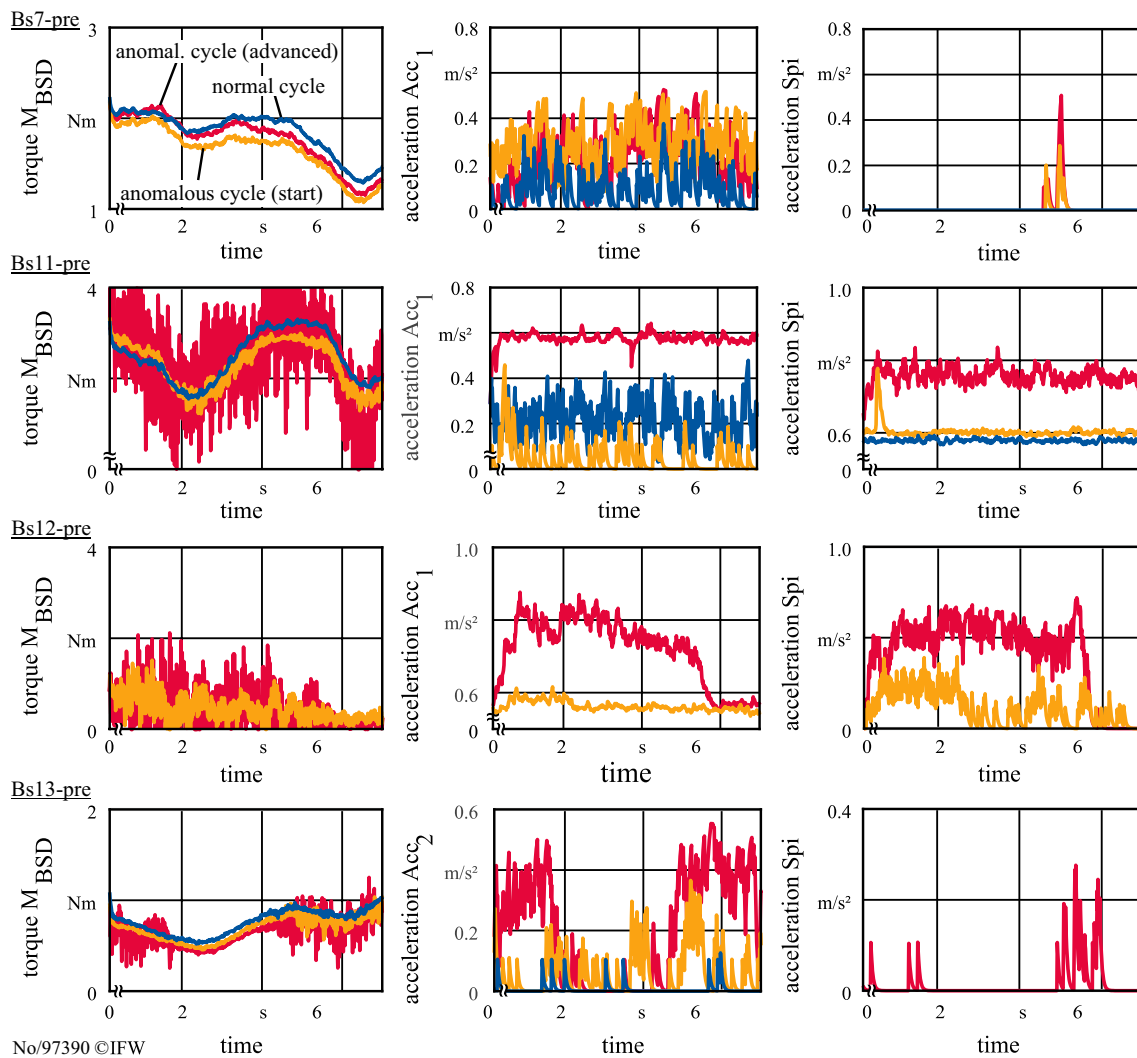
**Fig. 3** Visualization of different segmented signal trajectories in normal and abnormal condition

beginning of the abnormality. This is also observed in the signal of the acceleration sensor *Spi* of the spindle. As wear progresses, new signal plateaus are formed in all cases after several test cycles. These signal plateaus initially form for specific value ranges and increase in size over time.

Figure 4 depicts the frequency spectra of different ball screw conditions of the torque signal. For this purpose, the signals are transformed using a fast Fourier transform (FFT). In the case of the ball screws Bs11-pre and Bs13-pre, it can be seen that peaks occur in similar areas at the beginning of the anomaly. It is noted that in addition to the amplitude, the signal's frequency also changes as wear progresses. The frequency changes may be due to the fact that the damage to the ball raceways gets wider and thus the excitations change. Changes in the frequency range of the accelerometer is only observed in the case of the ball screw Bs11-pre.

However, the monitoring signals vary in the normal state. Recent studies have shown that monitoring signals change due to factors such as temperature, axis position, and ball screw exchanges regardless of the ball screw conditions [31]. Other reasons could be different lubrication and preload states. In addition, a slight tilting of the machine axes and adapted controller settings could also cause different signal trajectories. Figure 5 illustrates the distributions of the segmented time series of the torque as well as the acceleration sensors in the normal state. The acceleration sensor $Acc_{1-3}$ takes the value 0 for some test cycles in the case of ball screw Bs2, which indicates incorrect data acquisition. It is observed that the value range of the acceleration signal *Spi* is significantly larger than the signals of the triaxial acceleration sensor $Acc_{1-3}$. For those machines without a ball screw disassembly, the sensor values' ranges are very similar. However, the distributions of the torque take different shapes in distribution. It is observed that the value range of newly assembled ball screws,
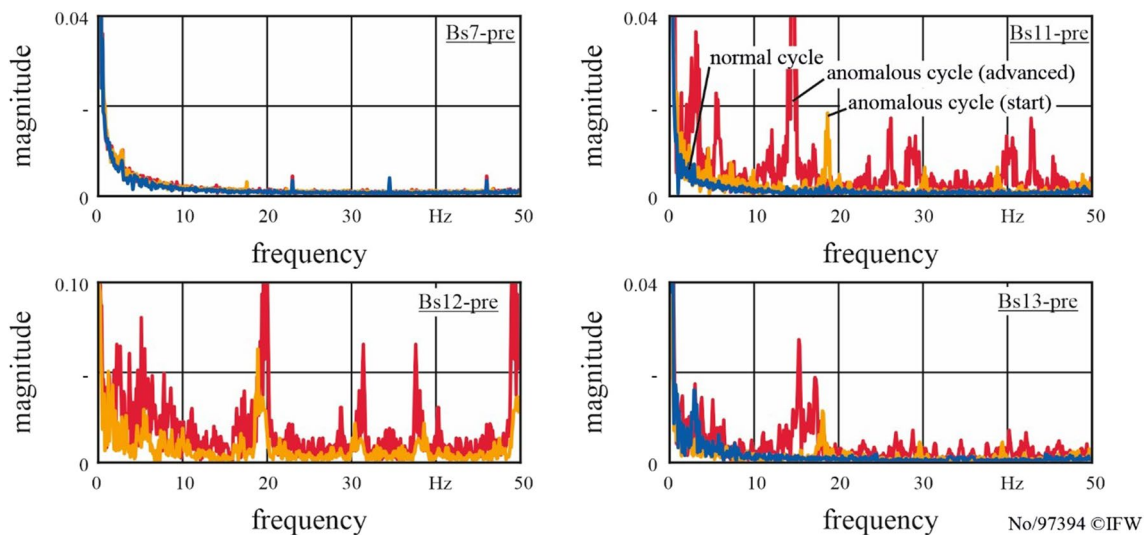
**Fig. 4** Visualization of the frequency spectrum of segmented torque signals for different ball screw conditions (anomalous cycles often show more and higher frequencies in the spectrum)

like torque $M_{BSD}$ and the acceleration signals ($Acc_{1-3}$, $Spi$) is significantly larger. This is due to the running-in processes of newly installed ball screws. Figure 6 presents the trajectories of the first 5 segmented torque signals after assembly. In the case of ball screw Bs7-post, Bs12-post, and Bs13-post, there are apparent differences in signal level and progression.

In addition to the signal changes in the running-in process, other signal patterns occur independently of condition changes of the ball screws. In the case of ball screw Bs6, higher frequencies occur in the torque of the forward motion without any replacement being documented. For ball screws Bs5, Bs9, and Bs10, higher frequencies are visible in the torque in the backward movement of the test cycle. In the case of torque, random level changes occur between test cycles in the normal condition. In addition, a gradual level shift is visible for the entire observation period for the torque and acceleration signals. In the case of acceleration signals, random peaks occur at irregular intervals in the normal state. As a result, robust monitoring strategies are needed to prevent false alarms.

## 4 Ball screw drive monitoring approaches for machine tool fleets using machine learning

### 4.1 Supervised anomaly detection approach using automated machine learning

In the first step, machine learning is used for supervised anomaly detection of ball screw drives assuming that fault data is available. AutoML methods are used for decision support for model selection. In short, AutoML refers to methods for the optimization, automation, and analysis of design decisions regarding the complete machine learning (ML) pipeline to obtain a model with peak performance. The ML pipeline comprises data preprocessing, feature selection, model selection, and the optimization of their hyperparameters, as well as post-processing of the results. The challenge involves determining a suitable solution within a computational budget in this large search space. Numerous approaches have been developed in the past to solve this problem [32–41]. These approaches allow domain experts without ML expertise to easily use ML methods in practice [18, 32]. Thornton et al. introduce Auto-WEKA to select models and optimize their hyperparameters for classification problems simultaneously. They treat the choice of the model as another hyperparameter and use sequential model-based algorithm configuration (SMAC) [42, 43] as their solver. SMAC is an iterative, global optimizer based on Bayesian optimization. In Bayesian Optimization, the true objective function which should be optimized is approximated by a surrogate model. This makes it very sample-efficient and requires only few function interactions which is especially useful if the function evaluation is costly or time-consuming [44]. Extensions of Auto-WEKA allow the selection of a model and its hyperparameters for regression and clustering tasks. The developed approach also enables the evaluation of features using filtering methods. The authors show that Auto-Weka can achieve better results than grid search or random search for model and hyperparameter selection [32].

A more recent approach inspired by Auto-WEKA is Auto-Sklearn, which can be used for regression and classification problems. Auto-Sklearn also uses SMAC as the optimizer. It further allows data preprocessing, e.g., the imputation of incomplete data, feature scaling, and
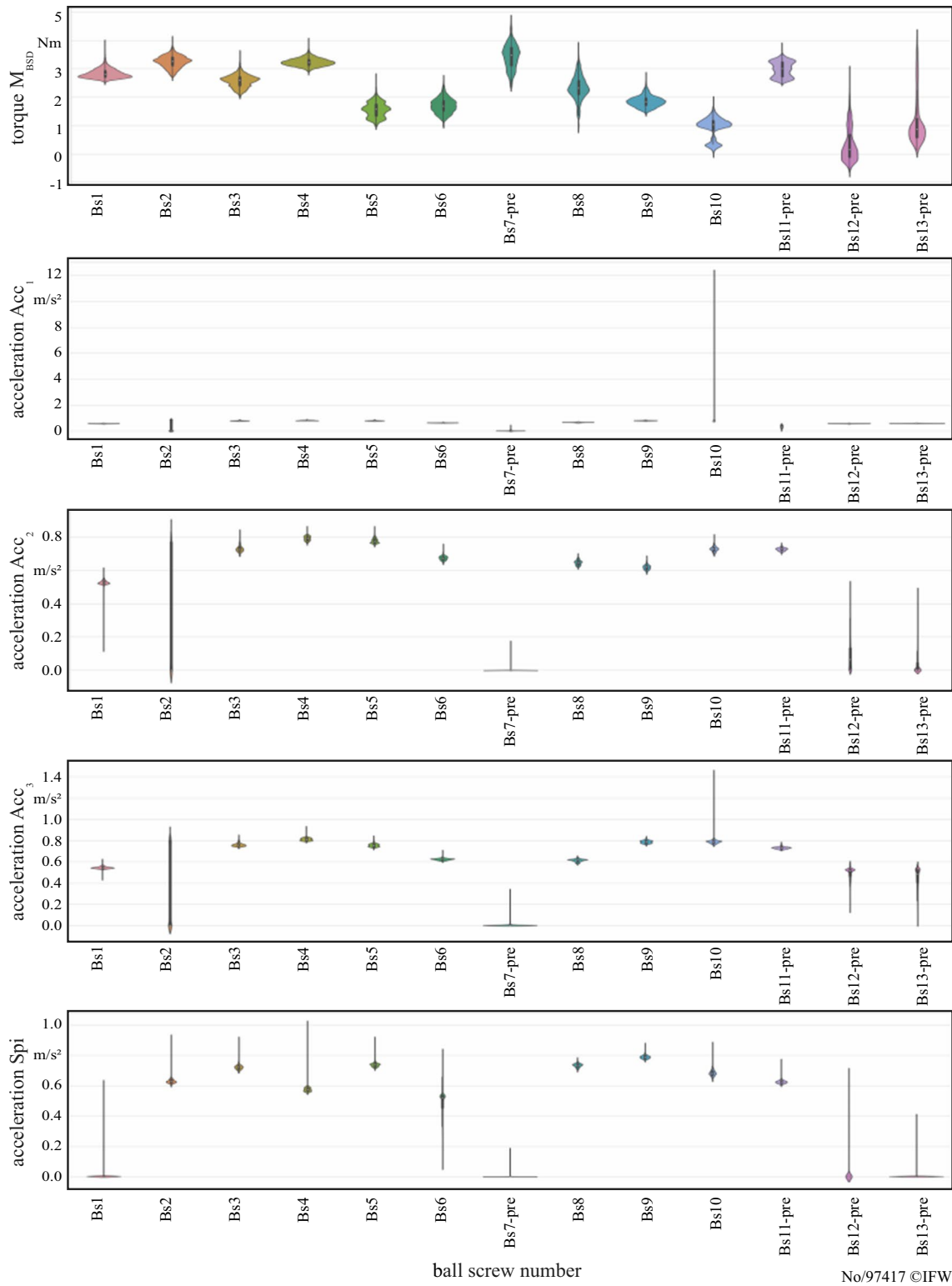
**Fig. 5** Distribution of segmented sensor values of ball screws in normal condition (signals from the same sensors for normal test cycles are highly distributed across machines, emphasizing the need for robust monitoring strategies)
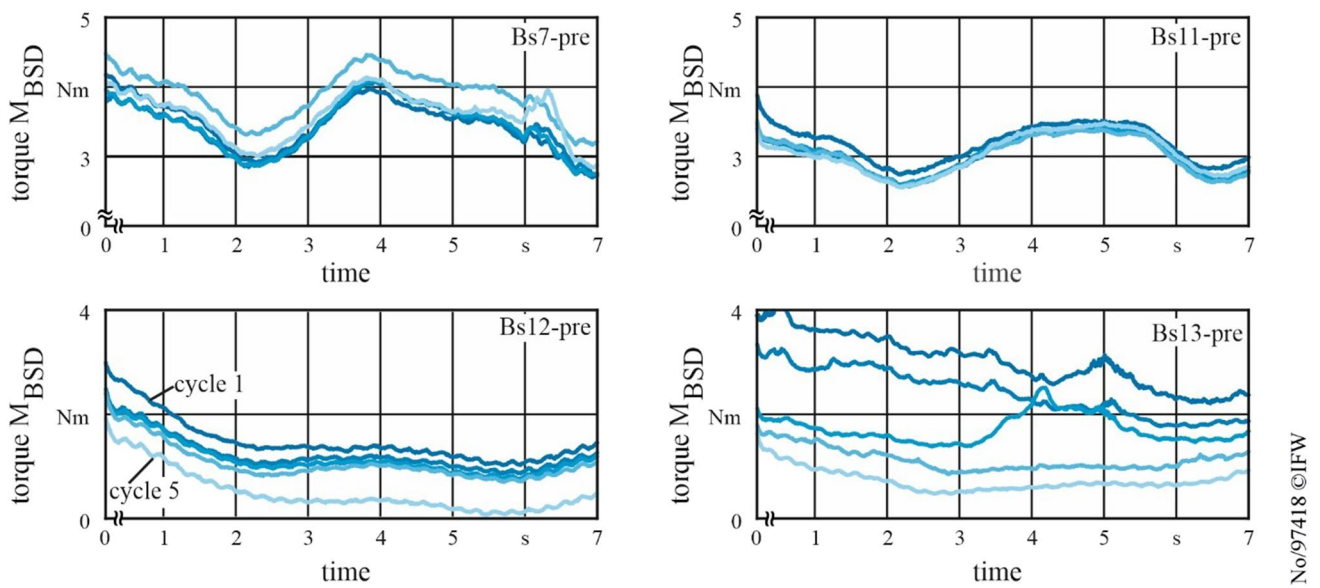
**Fig. 6** Segmented torque signals of the first five test cycles after assembly of a new ball screw (running-in behavior)

dimension reduction (e.g., PCA). In contrast to Auto-WEKA, Auto-Sklearn has two additional components. Meta-learning is utilized for finding good instantiations of Auto-Sklearn based on already-seen data sets. For this purpose, in an offline phase, data sets of the OpenML [45] database are described using meta-features. In the next step, optimal configurations for these data sets are determined by SMAC. A new data set is assigned to a group of similar data sets in the OpenML database using the meta-features. This enables quick access to precomputed optimal configurations stored in the database saving computational costs on the user's side. The second innovation enables the construction of ensembles with good prediction quality, allowing for more robust predictions. The authors showed that the prediction quality of Auto-Sklearn can outperform the results of other Auto-ML approaches for several data sets of the OpenML repository [18]. The recently released version Auto-Sklearn 2.0 provides a new meta-learning technique for improved handling of iterative algorithms.

Besides Auto-Sklearn and Auto-WEKA, other AutoML approaches such as hyperopt-sklearn, TPOT, TuPAQ, ATM, Automatic Frankensteining, ML-Plan, Auto-stacker, AlphaD3M, Collaborative Filtering, and Auto-Keras have also been published [17, 46]. An overview of different AutoML approaches and their features is given in Waring et al. [46]. Besides approaches from academia, there are numerous commercial approaches to AutoML, such as Rapidminer, Microsoft Azure Machine Learning, Google's Prediction API, Amazon Machine Learning, etc. [35]. In this study, Auto-Sklearn is used for supervised anomaly detection of ball screw drives in machine tool fleets.

The overall workflow with Auto-Sklearn is depicted in Fig. 7. Segments of the time series are usually selected to increase the monitoring quality. In addition to the time series of the test cycles, the labels for each time series are also available (see Table 2). In this work, a distinction is made between normal and faulty conditions (fault detection). For condition monitoring after data acquisition
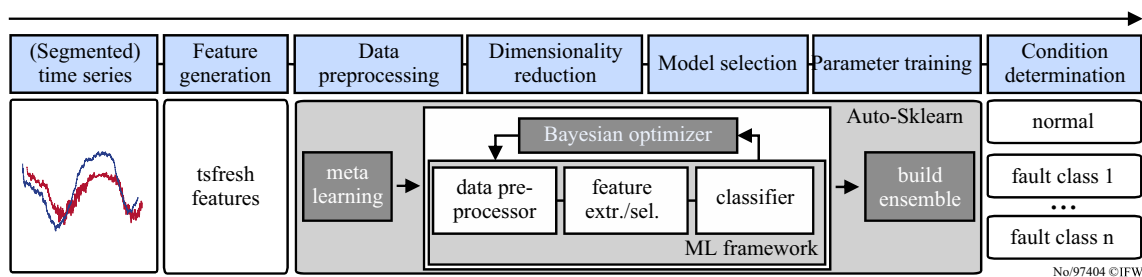


**Fig. 7** Supervised anomaly detection using automated machine learning

is extracting features from the time series because the supervised learning methods in Auto-Sklearn require a fixed length input. However, it is not possible to determine in advance which signal features are best suited for the respective monitoring application. For this reason, a high quantity of features needs to be generated from the data to obtain a few useful features [47]. This highlights the need for an automatic selection of the data and feature processing. More than 700 time series features are generated for each sensor using the tsfresh [48] python library to determine the condition of the ball screw drive. The default hyperparameters of the signal feature generation methods contained in tsfresh are applied. However, tsfresh does vary the hyperparameters of the feature methods. For this purpose, a set of different discrete values within a defined range of values is defined for the hyperparameters of the feature methods. Using the library tsfresh, features from the time domain (mean, standard deviation, kurtosis, skewness, etc.) and features from the frequency domain are generated. Thus, various statistical quantities of the absolute Fourier transform spectrum are considered features. In addition, the Fourier coefficients of the one-dimensional discrete Fourier transform are applied as features.

It should be noted that feature engineering and selection is essential for the monitoring quality. The generated features serve as input for Auto-Sklearn to determine the condition of the ball screws. Each pipeline constructed by Auto-Sklearn consists of up to three data preprocessors, one feature preprocessor and one classifier plus their respective hyperparameters. The search space for the ML pipeline is hierarchically organized as a tree and contains continuous, categorical and conditional hyperparameters. Auto-Sklearn can select from 16 classifiers, 19 feature preprocessing methods, and numerous data preprocessing methods for the classification task. In total, there are more than 150 hyperparameters [17]. The data preprocessing can include feature scaling, imputation of missing values, one-hot encoding, and/or balancing of target classes. Examples of feature preprocessing are PCA and ICA. Available classifiers are Adaboost, Naive Bayes, Decision Tree, Extra Trees, Gaussian Naive Bayes, Gradient Boosting, K-Nearest Neighbor, Linear Discriminant Analysis, Linear Support Vector Machine (SVM), Non-Linear SVM, Multi-layer Perceptron, Multinomial Naive Bayes, Passive Aggressive, Quadratic Discriminant Analysis, Random Forest, and Stochastic Gradient Descent. In addition, Auto-Sklearn builds ensembles for robust predictions. The idea behind ensemble building is based on the fact that classifiers have different advantages and disadvantages on different data sets that complement each other.

In contrast to many literature approaches, data from several machine tools are available in this work. Figure 5 illustrates that the data distribution in the normal state of ball screws differs from machine to machine. In addition, signal characteristics change over time without any defect of the ball screws being present. This raises the question of the generalizability or applicability of the ML-pipeline to new ball screws and the robustness against false alarms. For this reason, Chapter 5 evaluates different strategies for applying the presented approach to new and unseen data.

## 4.2 Semi-supervised anomaly detection

### 4.2.1 Computation of unified outlier scores using machine learning

In supervised anomaly detection, a labeled data set containing fault data is assumed to be available. If only insufficient fault data is available to train a classifier, semi-supervised anomaly detection approaches can be considered. A so-called baseline model is trained based on data describing the normal state. An outlier score is produced which varies in case of condition changes. In this work, the approach of Denkena et al. [49] is used and adapted for anomaly detection of ball screw drives in machine tool fleets. Thereby, methods for unsupervised anomaly detection are used for semi-supervised anomaly detection. According to Kriegel et al. [50], the approach for calculating uniform outlier scores is employed. Using the uniform outlier scores, the scores of several outlier score methods can be combined into an ensemble. Moreover, scores from multiple sensors can be aggregated for robust monitoring. In contrast to the work of Denkena et al. [49], data from multiple machine tools are considered. In addition, different scaling strategies are applied.

In the first step, feature groups are extracted based on the segmented signals. In contrast to the supervised approach, only simple signal features are considered. This is due to the fact that no fault data is available for model training. Table 3 provides an overview of the feature groups used.

The first group consists of the general-purpose features in the time domain, which are adopted from the study of Denkena et al. [49]. Another feature group uses information on the sample autocovariance. The autocovariance indicates how similar a time series $x_{i-l}$ shifted by $l$ discrete time steps is to the original time series $x_i$. According to Eq. (1), the sample autocovariance is calculated as follows [38, 51]:

$$\widehat{\gamma}(l) = \frac{1}{I-1} \sum_{i=1}^{I-1} x_i x_{i-l}. \tag{1}$$

**Table 3** Generated signal features for condition monitoring

| | |
|---|---|
| Time domain | General-purpose features (mean, standard deviation, root mean square, kurtosis, skewness, signal-to-noise-ratio, peak2peak, shape factor, crest factor, interquartile range, integral) |
| | Autocovariance (First ten lag coefficients) |
| Frequency domain | Amplitude and frequency of the five most dominant peaks between 10 and 50 Hz |

The sample autocovariance is calculated for $l \in \{0, \ldots, 9\}$. Features are also extracted from the frequency domain by transforming the raw data of all signals using an FFT. The amplitude and frequency of the five most dominant peaks between 10 and 50 Hz are used as another set of features. The *sciPy* library is used to calculate the features from the time domain [52]. Additionally, the *statsmodels* library is applied to compute the sample autocovariance [51].

In the next step, an outlier score calculation method is selected. Various methods exist for unsupervised anomaly detection that makes different assumptions about the data and the occurrence of anomalies. In this work, the K-Nearest Neighbor (KNN) method is used to evaluate the ball screw condition based on the extracted features of the test cycles. This method is characterized by a small number of hyperparameters and makes no assumptions about the data distribution or signal features. Using the KNN method, an anomaly score $S(o)$ is calculated for a new observation $o \in O$. Thereby, according to Eq. (2), the distance of a new observation $o \in O$ to its nearest neighbor $i \in N_k(o)$ is used as an anomaly score [53]:

$$S(o) = d(o,i), \forall i \in N_k(o). \tag{2}$$

For this purpose, a distance metric $d$ needs to be selected. An observation $o$ is the standardized feature vector extracted from the time series of test cycles $c \in \{1, \ldots, n_b\}$. Additionally, the outlier score is scaled using the approach of Kriegel et al. [50]. The scaling of the outlier score allows the calculation of decision boundaries and the construction of robust ensembles. According to Kriegel et al. [50], the outlier score is scaled to be *regular* and *normal*. An anomaly score $S$ is *regular* if $S(o) \gg 0$ is true for a new observation $o$ when anomalies occur. Thus, if a new observation $o$ does not represent an anomaly, $S(o) \approx 0$ holds. Equation (3) is used to regularize the KNN-score:

$$Reg_S^{base_S}(o) = max\{0, S(o) - base_S\}. \tag{3}$$

The minimum distance of an observation $o$ of the training dataset $O^{train}$ to its $N_k$ nearest neighbors is used as the basis:

$$base_{KNN} = min(dist_{o,i}), \forall o \in O^{train}, \forall i \in N_{k,o}. \tag{4}$$

An outlier score is *normal* if it assumes the value range [0,1]. There are different options for normalization, for example, linear scaling, Gaussian scaling, or gamma scaling. For a new observation $o \in O^{test}$, linear scaling is used to calculate the normalized anomaly score according to Eq. (5):

$$Norm_S^{linear}(o) = min\left\{ \frac{Reg_S(o) - Reg_{S,min}^{train}}{Reg_{S,max}^{train} - Reg_{S,min}^{train}}, 1 \right\}, \forall o \in O^{test}. \tag{5}$$

The linear scaling assumes an equal distribution of the regularized outlier scores. It should be noted that the optimal choice of the correct distribution depends, for example, on the method chosen to calculate the outlier scores. In this work, the Gaussian scaling as well as the Gamma scaling are applied. The Gaussian scaling contains only two adjustable parameters (mean and standard deviation). According to Eq. (6), the Gaussian scaling is calculated:

$$Norm_S^{gauss}(o) = max\left\{ 0, erf\left( \frac{Reg_S(o) - \mu_S^{Reg^{train}}}{\sigma_S^{Reg^{train}} \bullet \sqrt{2}} \right) \right\}, \forall o \in O^{test}. \tag{6}$$

Before normalization, the mean $\mu_S^{Reg^{train}}$ and standard deviation $\sigma_S^{Reg^{train}}$ of the regularized outlier scores of the training set are determined. The Gaussian error function ($erf$) is also employed. Kriegel et al. [50] note that low-dimensional KNN-scores are more likely to reflect a gamma distribution. To perform gamma scaling, the cumulative density function is calculated according to Eq. (7):

$$cdf_S^{gamma}(o) = P\left( \hat{k}, \frac{Reg_S(o)}{\hat{\theta}} \right). \tag{7}$$

Thereby, $P$ represents the regularized gamma function. The parameters of the gamma function (shape $k$ and mean $\theta$) are replaced by the estimators $\hat{k} = \frac{\mu_S^{Reg^{train}}}{\sigma_S^{Reg^{train}}}$ and $\hat{\theta} = \frac{\sigma_S^{Reg^{train}}}{\mu_S^{Reg^{train}}}$. Equation (8) is applied for normalization using gamma scaling:

$$Norm_S^{gamma}(o) = max\left\{0, \frac{cdf_S^{gamma}(o) - \mu_{cdf}}{1 - \mu_{cdf}}\right\}, \forall o \in O^{test}, \tag{8}$$

where $\mu_{cdf} = cdf_S^{gamma}\left(\mu_S^{Reg^{train}}\right)$. After calculating the normalized anomaly scores, an aggregated score considering $OD_j \in OD$ outlier scores of the ensemble is calculated according to Eq. (9):

$$P(o) = \frac{1}{|OD|} \sum_{OD_j \in OD} Norm_j(o), \forall o \in O^{test}. \tag{9}$$

In this work, the scores of the accelerometer signals $Acc_{1-3}$ are aggregated into an ensemble to minimize the number of false alarms. To decide whether a new observation $o \in O^{test}$ represents an anomaly, Eq. (10) is evaluated:

$$S^{final}(o) = \begin{cases} 1, & if\ P(o) > (1 - \beta), \forall o \in O^{test} \\ 0, & otherwise \end{cases}. \tag{10}$$

An alarm is issued in case of $S^{final}(o) = 1$. The risk factor $\beta$ allows us to adjust the sensitivity of the monitoring system.

### 4.2.2 Signal threshold-based approaches

In addition to machine learning, signal threshold-based approaches have been used for monitoring in the literature [9]. For example, fixed limits and tolerance bands are designed for process monitoring in machining to detect various anomalies such as collisions, overload situations of jammed tools, or tool breakage [54]. Two signal threshold-based approaches for semi-supervised anomaly detection are evaluated in this work. The first approach proceeds in such a way that certain signal features $sf_c$ fixed limits are calculated based on safety factor $\alpha$:

$$GP\_up = Max\left(sf_c\right) \bullet \alpha, \forall c \in C^{train}. \tag{11}$$

The safety factor $\alpha$ typically takes values of 1.1 or 1.2. An alarm is triggered if a signal feature $sf_c$ for $c \in C^{test}$ is greater than the limit value $GP\_up$.

In another approach, tolerance bands, according to Brinkhaus [54], are used for monitoring. In the first step, upper and lower envelopes $[h\_up_c(i), h\_lo_c(i)]$ around $x_c(i)$ are formed according to Eqs. (12) and (13):

$$h\_up_c(i) = Max\left[x_c(i - \theta), \dots, x_c(i + \theta)\right], \tag{12}$$

$$h\_lo_c(i) = Min\left[x_c(i - \theta), \dots, x_c(i + \theta)\right]. \tag{13}$$

It is assumed that the upper and lower envelopes follow a normal distribution. The parameter $\theta$ represents the shift factor of the time series. Based on the determined envelopes,

an upper and a lower limit value are determined according to the Eqs. (14) and (15):

$$GP\_up(i) = \overline{h\_up_c}\ (i) + \alpha \bullet s[h\_up(i)_c]\ , \forall c \in C^{train}, \tag{14}$$

$$GP\_lo(i) = \overline{h\_lo_c}\ (i) - \alpha \bullet s[h\_lo(i)_c], \forall c \in C^{train}. \tag{15}$$

The mean values $\overline{h\_up}\ (i)$ and $\overline{h\_lo}\ (i)$ and the standard deviations $s\left[h\_up(i)\right]$ and $s[h\_lo(i)]$ of the envelopes are used to calculate the tolerance bands. The safety factor $\alpha$ is adjusted to set the distance between the decision boundaries and the mean values of the envelopes. In the work of Brinkhaus [54], time series are weighted differently depending on their occurrence. Thus, the mean values and standard deviations of the envelopes are calculated based on the Eqs. (16) and (17) as a function of the memory parameter $a$:

$$\overline{h}_{c+1}(i) = (1 - a) \bullet \overline{h}_c(i) + a \bullet h_{c+1}(i), \tag{16}$$

$$s_{c+1}(i) = \sqrt{(1 - a)s_c(i)^2 + a[h_{c+1}(i) - \overline{h}_{c+1}(i)]^2}. \tag{17}$$

For larger values for the memory parameter $a$, the weight of past time series for calculating the mean and standard deviation of the envelopes is reduced and vice versa.

## 5 Results

### 5.1 Supervised anomaly detection

In the first step, the supervised anomaly detection approach presented in Chapter 4.1 is applied for fleet-based condition monitoring of ball screw drives in machine tools. In an experimental study, the prediction quality of different machine learning methods used in the literature (see Table 1) is compared to Auto-Sklearn. Auto-Sklearn 2.0 [17] (version 0.12.6) is used in the experiments. The data from all machines are combined into one set, and the time series are randomly shuffled. After feature generation using tsfresh, Auto-Sklearn is applied to perform fault detection. During optimization, fivefold cross-validation is performed in the inner training loop. Auto-Sklearn is compared to baseline methods used in literature with default hyperparameters. All baseline methods use the standard scaler as feature preprocessing (removing the mean and scaling to unit variance). Baseline methods are SVM, Decision Tree (DT), Gaussian Process Classifier (GP), K-Nearest Neighbor (KNN), Multilayer Perceptron (MLP), and Gaussian Naïve Bayes (GNB). In addition, methods for dimension reduction (feature extraction and feature selection) of the literature approaches are adopted. In this setting, Auto-Sklearn selects one single classifier for predictions. All experiments are performed on Intel

Core i9-9900KF CPUs with 3.6 GHz and 32 GB RAM. A time budget of 1500 s is defined for Auto-Sklearn (fivefold inner cross-validation).

The predictions of binary classifiers can be evaluated using various metrics. Table 4 shows a confusion matrix for predictions of binary classifiers. The so-called *false positives* represent the number of false alarms issued by the monitoring system. The *false negatives* represent the number of anomalies not detected by the monitoring system. Combined with the number of test cycles correctly detected as anomalies, the values for *Precision* and *Recall* are calculated according to Eqs. (18) and (19). Based on these values, the f1-metric is calculated according to Eq. (20). The proposed monitoring approach and the baselines are evaluated on an outer fivefold cross-validation for 5 different random seeds using f1-metric. The f1-metric is applied because the data set is unbalanced by the fewer number of faulty test cycles.

$$Precision = \frac{tp}{tp + fp}, \tag{18}$$

$$Recall = \frac{tp}{tp + fn}, \tag{19}$$

$$f1score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}. \tag{20}$$

**Table 4** Confusion matrix and the relation between f1-score, precision, and recall

| | | True condition | | |
|---|---|---|---|---|
| | | Faulty | Healthy | |
| Predicted condition | **Faulty** | True positive (tp) | False positive (fp) | **Precision** |
| | **Healthy** | False negative (fn) | True negative (tn) | |
| | | **Recall** | | **f1-score** |

Thereby, a perfect classifier achieves an f1-score of 1. It should be noted that this evaluation procedure is used for model comparison. In practice, Auto-Sklearn only needs to be run once using inner cross-validation on the whole data set. Table 5 shows the results for the case of non-segmented time series. The highest classification accuracy of the baseline approaches is achieved by the MLP classifier (f1-score of 0.9059). For GP, the f1-score with baseline settings is 0.0000. This is due to the fact that GP finds no true positives. Auto-Sklearn achieves the highest f1-score of 0.9509. A further step involves segmenting the time series. Thereby, only the segment of the time series in which the ball screw moves in the forward direction is considered, i.e., $t\epsilon[SB, SE]$. Thereby, *SB* and *SE* represent the start and the end of the segmentation window, respectively. It is observed that across all baselines, the classification accuracy is lower compared to the non-segmented case. The best baseline approach MLP realizes an f1-score of 0.8924. Auto-Sklearn achieves the best result (f1-score of 0.9576). Overall, the standard deviations are lower compared to the non-segmented case. In summary, Auto-Sklearn performs well in a short amount of time whereas the baselines from the literature provide poor results. Auto-Sklearn also achieves robust monitoring results in both the segmented and non-segmented case.

Furthermore, it is evaluated how often a certain classifier and feature preprocessing method is considered by Auto-Sklearn. Figure 8 illustrates that RF is most commonly selected by Auto-Sklearn in case of non-segmented test cycles. It is noticeable that no preprocessing is applied most frequently.

The final f1-score depends significantly on the preset time budget of Auto-Sklearn. Figure 9 illustrates the incumbent changes of Auto-Sklearn and the best baseline approach over time. Thereby, incumbent denotes the currently best hyperparameter configuration. Auto-Sklearn outperforms the best baseline approach after a few seconds.

Furthermore, the evaluation mode is adapted in a further step. In the previous evaluation mode, the time series of all machine tools are combined into one data set and randomly shuffled. As shown in Chapter 3, the distributions and value

**Table 5** Performance by method (shuffled mode)

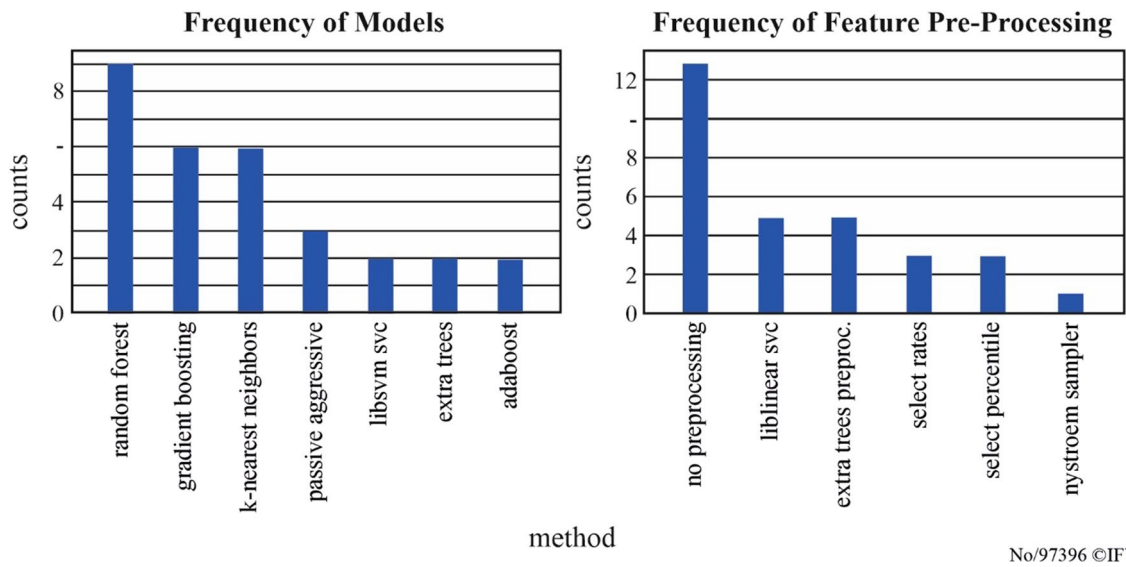| Method | Segmented signals | | Non-segmented signals | |
|---|---|---|---|---|
| | f1 mean | f1 std | f1 mean | f1 std |
| Decision Tree (baseline [11]) | 0.8214 | 0.0323 | 0.8881 | 0.0512 |
| K-Nearest Neighbor (baseline [12]) | 0.8042 | 0.0518 | 0.8921 | 0.0604 |
| Gaussian Naive Bayes (baseline [12]) | 0.7475 | 0.0182 | 0.7893 | 0.0295 |
| Support Vector Machine (baseline [13]) | 0.8821 | 0.0216 | 0.8946 | 0.0241 |
| Gaussian Process Classifier (baseline [14]) | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| Multi-layer Perceptron (baseline [15]) | 0.8924 | 0.0349 | 0.9059 | 0.0464 |
| Support Vector Machine (baseline [16]) | 0.8755 | 0.0176 | 0.9004 | 0.0455 |
| **Auto-Sklearn (ensemble size 1)** | **0.9576** | **0.0158** | **0.9509** | **0.0291** |

**Fig. 8** Frequency of chosen models and chosen feature pre-processing methods by Auto-Sklearn (ensemble size 1) in case of non-segmented test cycles

ranges of the sensor data, especially the torque, vary between the respective machine tools. Therefore, the question arises how robust the monitoring system is for new and unseen ball screws. In the adapted evaluation mode, the data is iteratively partitioned so that Auto-Sklearn is applied to the ball screws of each machine tool separately (outer ball screw cross-validation mode). In each iteration, data from one ball screw is included in the test set and data from the remaining ball screws are included in the training set. To optimize Auto-Sklearn, a fivefold inner cross-validation is performed using training data. The ensemble size is set to 1 for Auto-Sklearn. The results for an ensemble size of 10 is shown in the appendix. For the ball screws that contain anomalies (Bs7-pre, Bs11-pre, Bs12-pre, Bs13-pre), the f1-metric is used to evaluate the monitoring quality. For the remaining

ball screws that do not contain faulty time series, the false alarm rate *FAR* according to Eq. (21) is used for evaluation:

$$FAR = \frac{misclassified\ normal\ cycles}{number\ of\ normal\ cycles}. \tag{21}$$

The false alarm rate *FAR* is calculated by dividing the normal condition time series that are falsely declared as faulty time series by all normal condition time series to be tested. The results of the evaluation are shown in Table 6.

The evaluation is performed considering segmented and non-segmented time series and different sensor groups. It should be noted that ball screw Bs12-pre is in a faulty state when the data acquisition started. It is observed that the number of detected faulty time series is significantly lower compared to the previous evaluation mode. This is due to

**Fig. 9** Incumbent performance over time (median) in case of non-segmented test cycles
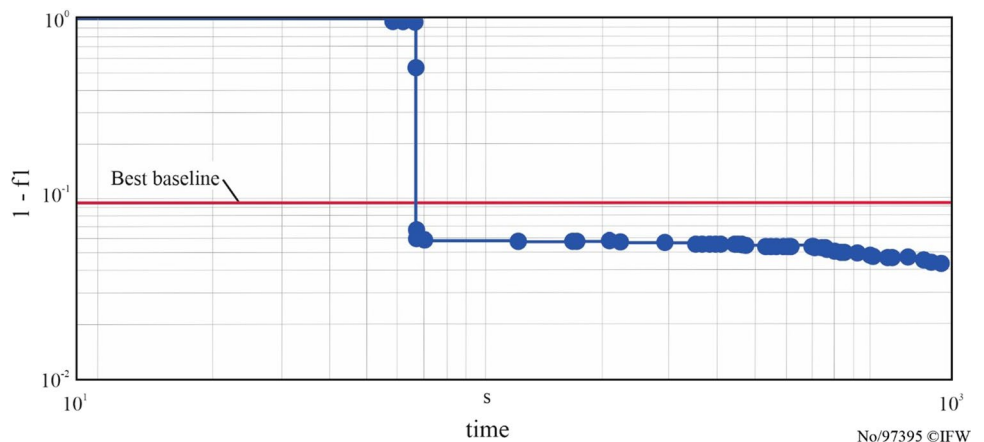
**Table 6** Performance indicators overview applying Auto-Sklearn (outer ball screw cross-validation mode)

| Ball screw (test) | Segmented | | | | | | Non-segmented | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | All sensors $M_{BSD}$, $Acc_{1-3}$, $Spi$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | All sensors $M_{BSD}$, $Acc_{1-3}$, $Spi$ | |
| | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR |
| Bs1 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs2 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs3 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs4 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs5 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs6 | - | 13.87 | - | 0.01 | - | 9.49 | - | 0.03 | - | 0.00 | - | 0.00 |
| Bs7-pre | 0.00 | - | 37.50 | - | 0.00 | - | 0.00 | - | 25.00 | - | 0.00 | - |
| Bs7-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs8 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs9 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs10 | - | 2.65 | - | 0.00 | - | 0.00 | - | 21.12 | - | 0.00 | - | 0.70 |
| Bs11-pre | 0.00 | - | 65.82 | - | 49.32 | - | 65.82 | - | 58.67 | - | 46.38 | - |
| Bs11-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs12-pre* | 22.22 | - | 91.18 | - | 90.41 | - | 100.0 | - | 99.32 | - | 100.00 | - |
| BS12-post | - | 0.00 | - | 37.5 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs13-pre | 24.39 | - | 38.10 | - | 20.00 | - | 5.71 | - | 34.15 | - | 25.64 | - |
| Bs13-post | - | 0.00 | - | 12.50 | - | 0.00 | - | 0.00 | - | 75.00 | - | 0.00 |
| Mean | | 1.27 | | 3.85 | | 0.73 | | 1.63 | | 5.77 | | 0.05 |

*At the start of the data acquisition, ball screw drive bs12-pre is already in fault condition

the fact that the sensor value trajectories and distributions differ for each ball screw. In addition, the adaptive evaluation mode provides significantly fewer fault data to learn anomaly patterns in cases where faulty ball screw drives are tested. Condition changes are detected only in advanced faulty states in the case of ball screws bs7-pre, bs11-pre, and bs13-pre. As a result, the number of available fault data is not sufficient to detect incipient anomalies in the transition phase. Condition changes of ball screw bs7-pre are only detected using the acceleration sensors $Acc_{1-3}$. Considering the acceleration signals $Acc_{1-3}$, a larger number of faulty test cycles are detected compared to using the torque signal $M_{BSD}$. Therefore, it is concluded that the torque signal is not sufficient for robust detection of faulty conditions. When utilizing all available sensor signals ($M_{BSD}$, $Acc_{1-3}$, $Spi$), condition changes of ball screws bs11-pre, bs12-pre, and bs13-pre are detected in the segmented and non-segmented case. Due to the lower amount of detected fault cycles, the acceleration and torque signals should be evaluated separately. However, the false alarm rate is the lowest across all ball screws considering all available sensors.

## 5.2 Semi-supervised anomaly detection

The first step evaluates the suitability of signal threshold-based approaches for semi-supervised anomaly detection of ball screw drives in machine tool fleets. These approaches are applied when limited or no information about faults is available. The results for the segmented sensor signals are presented since the monitoring quality is superior compared to the non-segmented case. The signal threshold-based approaches are applied first. According to Eq. (11), fixed limits are determined for various signal features based on the test cycles that describe the normal condition. However, a variety of challenges exist in the application of fixed limits. This approach is suitable for simple anomalies where complicated interactions between signal features do not need to be evaluated. Figure 10 illustrates the fixed limits ($\beta = 1.1, 1.2$) for the *peak-to-peak* value of the segmented torque signal $M_{BSD}$ computed based on the first ten normal running test cycles.

Condition changes are reliably detected in the case of ball screw bs11-pre. It is observed that in the case of ball screws bs7-pre, bs12-pre, and bs13-pre, the feature changes with the replacement of the ball screw rather than with the occurrence of the anomaly. Similarly, in case ball screw bs13-pre, the feature changes at the beginning of data recording, so anomalies are not detected. In addition, false alarms are issued for the *peak-to-peak* feature in case of ball screws bs3, bs8, and bs9 without any anomalies occurring. In summary, fault patterns vary, and thus, the
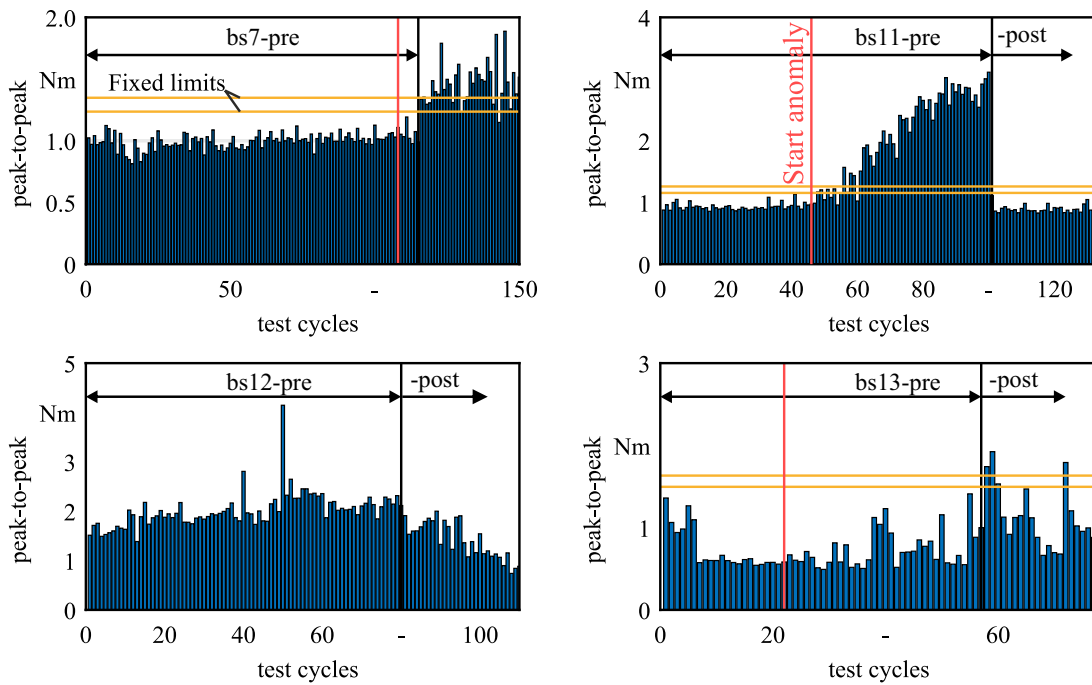
**Fig. 10** Visualization of fixed limits for ball screw drive monitoring; (red) start of degradation and (black) replacement of a ball screw (Robust monitoring results cannot be achieved by fixed limits over individual features)

present monitoring problem cannot be solved considering single features without evaluating interactions of features. Some sensor features vary independently of the ball screw condition which increases the risk of false alarms. This is also true for the triaxial accelerometer signals $Acc_{1-3}$ and the spindle accelerometer $Spi$.

In addition, the monitoring quality of the tolerance bands presented in Chapter 4.2.2 is evaluated. In Fig. 11, tolerance bands ($\alpha = 6$, $\theta = 0.4$) using the segmented torque $M_{BSD}$ and the acceleration signal $Acc_1$ of ball screw bs11-pre are shown. Signal changes in the case of torque and acceleration signals are not detected as the anomalies occur. The number of false alarms increases significantly when the safety factor $\alpha$ is reduced. It should be noted that tolerance bands only

evaluate the time domain of the signals. In summary, it can be stated that the presented threshold-based approaches are not suitable for robust ball screw monitoring in machine tools.

The next step uses uniform outlier scores for ball screw drive monitoring. For this purpose, a so-called baseline model is trained based on data describing the normal condition of ball screws. The outlier score is used as a health indicator to evaluate the ball screw condition. Since no fault data is available, the outlier score is calculated based on certain feature groups described in Chapter 4.2. In addition, the evaluation of the monitoring quality for the torque $M_{BSD}$ and acceleration signals $Acc_{1-3}$ is performed separately. This is due to the fact that condition changes are not always visible
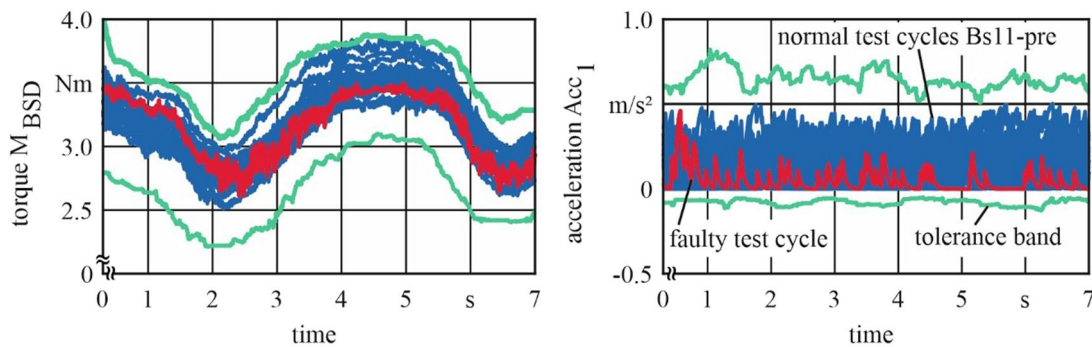


**Fig. 11** Visualization of tolerance bands for ball screw drive monitoring (condition changes of ball screw drives are often not detected)

in the torque signals (e.g., for ball screw Bs7-pre). Consequently, the number of detected anomalies is reduced by combining the outlier scores of the torque and the acceleration signals.

The KNN-score is utilized to produce outlier scores. The number of $k$-nearest neighbors ($k = 5$) and the distance metric (Minkowski metric) are chosen. The risk factor $\beta$ is set to $10^{-5}$. The PyOD python library [55] is applied to calculate the raw values of the outlier scores. Gaussian scaling is first implemented to normalize the outlier scores. The outlier scores of the triaxial accelerometers $Acc_{1-3}$ are aggregated into an ensemble using Eq. (9). This step is necessary because these signals vary significantly compared to the torque signal in the normal state, increasing the risk of false alarms.

Overall, two approaches are evaluated to split the dataset and apply the baseline model. The first approach performs a ball screw cross-validation. In each iteration, one ball screw is used as the test data set. The remaining ball screws without anomalies represent the training data set. Table 7 depicts the results of the evaluation. It is observed that in the case of torque $M_{BSD}$, faulty states are detected for ball screw Bs11-pre by using all feature groups. However, the number of detected faulty test cycles depends on the feature group used. The highest f1-score of 85.42 is obtained using the peaks of the frequency spectrum. No faulty test cycles are detected

for the Bs7-pre and Bs13-pre ball screws. This is due to the fact that no changes in the torque signal occur in the case of the Bs7-pre ball screw. In the case of the $Acc_{1-3}$ accelerometers, no faulty test cycles are detected overall. The result indicates that this application method is unsuitable for robust monitoring regarding the low number of detected anomalies.

The evaluation mode is changed in the second step. In each iteration, only the data of the particular ball screw to be tested is considered (separate training mode). The initial training database represents the first 10 test cycles of the tested ball screw. For all remaining test cycles without anomalies of the same ball screw, it is iteratively checked whether false alarms are issued. After each iteration, the tested test cycle is added to the training database. For those ball screws without anomalies, the false alarm rate $FAR$ is calculated. For all other ball screws containing faulty test cycles, the f1-score is applied to determine the monitoring quality.

The evaluation results are presented in Table 8. For the torque $M_{BSD}$ signals, an f1-score of 98.18 is obtained using the peaks of the frequency spectrum for ball screw Bs11-pre. In addition, faulty test cycles are also detected for ball screw Bs13-pre (f1-score: 72.41). It is recognized that the number of false alarms increased significantly compared to the first evaluation mode. This is caused by a lower number of training samples. False alarms are generated in the case of 5 ball

**Table 7** Performance indicators overview applying unified outlier scores (ball screw cross-validation, gaussian scaling)

| Ball screw (test) | General purpose features | | | | Autocov. features | | | | Peaks frequency spectrum | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | |
| | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR |
| Bs1 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs2 | - | 0.00 | - | 62.59 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs3 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs4 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs5 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs6 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs7-pre | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - |
| Bs7-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs8 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs9 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs10 | - | 1.32 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs11-pre | 35.82 | - | 0.00 | - | 73.56 | - | 0.00 | - | 85.42 | - | 0.00 | - |
| Bs11-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs12-pre* | - | - | - | - | - | - | - | - | - | - | - | - |
| Bs12-post | - | 68.97 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs13-pre | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - |
| Bs13-post | - | 5.00 | - | 0.00 | - | 10.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| **Mean** | | **5.79** | | **4.81** | | **0.77** | | **0.00** | | **0.00** | | **0.00** |

*At the start of the data acquisition, ball screw drive bs12-pre is already in a faulty condition

**Table 8** Performance indicators overview applying unified outlier scores (separate training, gaussian scaling)

| Ball screw (test) | General purpose features | | | | Autocov. features | | | | Peaks frequency spectrum | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | |
| | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR |
| Bs1 | - | 0.00 | - | 0.00 | - | 0.79 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs2 | - | 0.78 | - | 3.10 | - | 6.20 | - | 0.78 | - | 0.00 | - | 0.00 |
| Bs3 | - | 2.33 | - | 0.78 | - | 1.55 | - | 1.55 | - | 0.78 | - | 0.00 |
| Bs4 | - | 0.78 | - | 1.55 | - | 2.33 | - | 0.78 | - | 0.00 | - | 0.00 |
| Bs5 | - | 1.59 | - | 3.17 | - | 3.97 | - | 3.17 | - | 0.79 | - | 0.00 |
| Bs6 | - | 2.36 | - | 0.00 | - | 3.15 | - | 0.00 | - | 1.57 | - | 0.00 |
| Bs7-pre | 20.00 | - | 60.00 | - | 18.18 | - | 0.00 | - | 0.00 | - | 0.00 | - |
| Bs7-post | - | 4.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 4.00 | - | 0.00 |
| Bs8 | - | 6.14 | - | 1.75 | - | 3.51 | - | 1.75 | - | 1.75 | - | 0.00 |
| Bs9 | - | 2.83 | - | 0.00 | - | 2.83 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs10 | - | 3.55 | - | 0.00 | - | 2.13 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs11-pre | 90.00 | - | 82.98 | - | 86.87 | - | 65.85 | - | 98.18 | - | 0.00 | - |
| Bs11-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs12-pre* | - | - | - | - | - | - | - | - | - | - | - | - |
| Bs12-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs13-pre | 0.00 | - | 40.00 | - | 0.00 | - | 43.48 | - | 72.41 | - | 0.00 | - |
| Bs13-post | - | 10.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| **Mean** | | **2.64** | | **0.80** | | **2.04** | | **0.62** | | **0.68** | | **0.00** |

*At the start of the data acquisition, ball screw drive bs12-pre is already in a faulty condition

screws (Bs3, Bs5, Bs6, Bs7-post, Bs8). Condition changes are detected for the acceleration signals $Acc_{1-3}$ considering the general purpose features for the ball screws BS7-pre, Bs11-pre, and Bs13-pre. However, false alarms are issued for ball screws Bs2, Bs3, Bs4, Bs5, and Bs8.

Subsequently, gamma scaling is applied to normalize the regularized outlier scores. The corresponding results are illustrated in Table 9. In comparison with Gaussian scaling, the number of false alarms is reduced. Robust monitoring results are obtained for the torque signals $M_{BSD}$ considering the peaks of the frequency spectrum. Condition changes are detected for ball screws Bs11-pre and Bs13-pre. At the same time, no false alarms are produced. No false alarms are generated in the case of ball screw bs6 despite signal changes in the frequency spectrum of the torque signal. This is due to the fact that these signal changes occurred in the first test cycles which are part of the initial training database. For the acceleration signals $Acc_{1-3}$, the features of the auto-covariance are suitable for monitoring. However, the number of detected faulty test cycles is lower than the torque for the Bs11-pre (f1-score: 58.97) and Bs13-pre

(f1-score: 36.36) ball screws. Comparing the results between Table 7 and Table 9, it is evident that the monitoring quality is significantly increased by the separated training of the baseline model. Apart from ball screw bs11-pre, condition changes of ball screw bs13-pre are also detected in the separate training mode.

In summary, the separate training of the baseline model is necessary because the distribution of the sensor data for each machine tool shows significant differences. In addition to Gaussian and gamma scaling, linear scaling is also applied to normalize the outlier scores. However, the number of false alarms generated is significantly higher than Gaussian and gamma scaling.

## 6 Conclusion

This paper presents machine learning approaches for ball screw drive monitoring in machine tool fleets. The data set originates from test cycles of thirteen identical 5-axis machine tools used in series production. The results are as follows:

**Table 9** Performance indicators overview applying unified outlier scores (separate training, gamma scaling)

| Ball screw (test) | General purpose features | | | | Autocov. features | | | | Peaks frequency spectrum | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | |
| | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR |
| Bs1 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs2 | - | 0.00 | - | 1.55 | - | 0.00 | - | 0.78 | - | 0.00 | - | 0.00 |
| Bs3 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs4 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs5 | - | 0.00 | - | 0.79 | - | 0.00 | - | 0.79 | - | 0.00 | - | 0.00 |
| Bs6 | - | 0.79 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs7-pre | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - |
| Bs7-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs8 | - | 0.88 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs9 | - | 0.00 | - | 0.00 | - | 0.94 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs10 | - | 2.13 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs11-pre | 87.76 | - | 0.00 | - | 73.56 | - | 58.97 | - | 98.15 | - | 0.00 | - |
| Bs11-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs12-pre* | - | - | - | - | - | - | - | - | - | - | - | - |
| Bs12-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs13-pre | 0.00 | - | 24.39 | - | 0.00 | - | 36.36 | - | 72.41 | - | 0.00 | - |
| Bs13-post | - | 10.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| **Mean** | | **1.06** | | **0.18** | | **0.07** | | **0.12** | | **0.00** | | **0.00** |

*At the start of the data acquisition, ball screw drive bs12-pre is already in a faulty condition

1. Challenges in ball screw drive monitoring consist of the limited amount of fault data and changes in the monitoring signals in the normal state.

2. The data analysis reveals that the internal control data (torque) evaluation is insufficient for detecting condition changes in all ball screw drives.

3. Supervised machine learning methods are suitable for data-based ball screw anomaly detection in case the condition labels are given. In this context, a monitoring approach based on automated machine learning is developed to detect condition changes. Several strategies are examined to split the data to achieve the highest possible generalizability and robustness. The proposed approach achieved better classification results compared to literature approaches. Taking into account external sensors (acceleration data), condition changes are correctly detected for all ball screw drives. However, the available data are not sufficient to learn the transition phase between normal and faulty states.

4. In addition, a semi-supervised anomaly detection approach based on uniform outlier scores is applied. A baseline model is used to learn the normal state of the ball screw drives. Condition changes are detected using an outlier score of the baseline model. By using unified outlier scores it is possible to build robust ensembles of acceleration signals to prevent false alarms. Robust results are obtained applying the k-nearest neighbor outlier score and gamma scaling. It is found that a baseline model should be trained specifically for each ball screw separately. In addition, the sensor signals should be evaluated separately in the semi-supervised anomaly detection mode. The presented approach achieves a better monitoring quality than signal threshold-based approaches such as tolerance bands and fixed limits.

# Appendix

**Table 10** Performance indicators overview applying Auto-Sklearn (outer ball screw cross-validation, ens. size = 10)

| Ball screw (test) | Segmented | | | | | | Non-segmented | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | All sensors $M_{BSD}$, $Acc_{1-3}$,$Spi$ | | Torque $M_{BSD}$ | | Acceleration $Acc_{1-3}$ | | All sensors $M_{BSD}$, $Acc_{1-3}$ ,$Spi$ | |
| | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR | f1 | FAR |
| Bs1 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs2 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs3 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs4 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs5 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs6 | - | 12.41 | - | 0.00 | - | 9.49 | - | 1.60 | - | 0.00 | - | 0.00 |
| Bs7-pre | 0.00 | - | 72.72 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - |
| Bs7-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs8 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs9 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs10 | - | 0.00 | - | 1.99 | - | 0.00 | - | 9,86 | - | 0.00 | - | 0.70 |
| Bs11-pre | 0.00 | - | 0.00 | - | 3.57 | - | 0.00 | - | 23.33 | - | 0.00 | - |
| Bs11-post | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs12-pre* | 31.58 | - | 96.77 | - | 100.0 | - | 100.0 | - | 95.04 | - | 100.00 | - |
| BS12-post | - | 0.00 | - | 55.17 | - | 0.00 | - | 0.00 | - | 0.00 | - | 0.00 |
| Bs13-pre | 15.38 | - | 20.00 | - | 15.38 | - | 11.11 | - | 25.64 | - | 16.22 | - |
| Bs13-post | - | 0.00 | - | 10.00 | - | 0.00 | - | 0.00 | - | 25.00 | - | 6.25 |

*At the start of the data acquisition, ball screw drive bs12-pre is already in a faulty condition

**Co-funded by the European Union**

# Declarations

**Ethics approval** Not applicable.

**Consent to participate** Given by all involved.

**Consent for publication** Given by all involved.

**Competing interests** The authors declare no competing interests.

# References

1. Altintas Y, Verl A, Brecher C et al (2011) Machine tool feed drives. CIRP Ann 60:779–796. https://doi.org/10.1016/j.cirp.2011.05.010

2. Imiela J (2006) Verfügbarkeitssicherung von Werkzeug-maschinenachsen mit Kugelgewindetrieb durch modellbasierte Verschleißüberwachung. Zugl.: Hannover, Univ., Diss., 2005. Berichte aus dem IFW / Institut für Fertigungstechnik und Werkzeug-maschinen, 01/2006. PZH Produktionstechn, Zentrum, Garbsen

3. Schopp M (2009) Sensorbasierte Zustandsdiagnose und -prognose von Kugelgewindetrieben. Zugl.: Karlsruhe, Univ., Diss., 2009. Forschungsberichte aus dem wbk, Institut für Produktionstech-nik, Karlsruher Institut für Technologie (KIT), vol 152. Shaker, Aachen

4. Butler Q, Ziada Y, Stephenson D et al. (2022) Condition monitor-ing of machine tool feed drives: a review. Journal of Manufacturing Science and Engineering 144. https://doi.org/10.1115/1.4054516

5. Haberkern A (1998) Leistungsfähigere Kugelgewindetriebe durch Beschichtung. Dissertation, Universität Karlsruhe

6. Machelski EJ (1996) Get more life out of that ball screw. Power Transmission Design 38

7. Brecher C (2021) Werkzeugmaschinen Fertigungssysteme 3: Mechatronische Systeme, Steuerungstechnik und Automati-sierung, 9th edn. Springer, Berlin / Heidelberg, Berlin, Heidelberg

8. Schwabacher M, Goebel K (2007) A survey of artificial intelli-gence for prognostics. AAAI Fall Symposium - Technical Report

9. Isermann R (2005) Model-based fault-detection and diagnosis – status and applications. Annu Rev Control 29:71–85. https://doi.org/10.1016/j.arcontrol.2004.12.002

10. Jin W, Chen Y, Lee J (2013) Methodology for ball screw com-ponent health assessment and failure analysis. In: Proceedings of the ASME 2013 International Manufacturing Science and Engineering Conference. ASME, Madison, Wisconsin, USA. https://doi.org/10.1115/MSEC2013-1252

11. Denkena B, Bergmann B, Schmidt A (2021) Preload monitoring of single nut ball screws based on sensor fusion. CIRP J Manuf Sci Technol 33:63–70. https://doi.org/10.1016/j.cirpj.2021.02.006

12. DEmilia G, Gaspari A, Hohwieler E, et al (2018) Improvement of defect detectability in machine tools using sensor-based con-dition monitoring applications. Procedia CIRP 67:325–331. https://doi.org/10.1016/j.procir.2017.12.221

13. Feng G-H, Pan Y-L (2012) Establishing a cost-effective sensing system and signal processing method to diagnose preload levels of ball screws. Mech Syst Signal Process 28:78–88. https://doi.org/10.1016/j.ymssp.2011.10.004

14. Benker M, Kleinwort R, Zah MF (2019) Estimating remaining useful life of machine tool ball screws via probabilistic classifi-cation. In: 2019 IEEE International Conference on Prognostics and Health Management (ICPHM). IEEE, San Francisco, CA, pp 1–7. https://ieeexplore.ieee.org/document/8819445

15. Balaban E, Bansal P, Stoelting P et al (2009) A diagnostic approach for electro-mechanical actuators in aerospace systems. In: 2009 IEEE Aerospace conference. IEEE, Big Sky, MT, pp 1–13. https://ieeexplore.ieee.org/document/4839661

16. Li P, Jia X, Feng J et al (2018) Prognosability study of ball screw degradation using systematic methodology. Mech Syst Signal Process 109:45–57. https://doi.org/10.1016/j.ymssp.2018.02.046

17. Feurer M, Eggensperger K, Falkner S et al (2022) Auto-Sklearn 2.0: hands-free AutoML via meta-learning. J Mach Learn Res 23(261). https://doi.org/10.48550/arXiv.2007.04074

18. Feurer M, Klein A, Eggensperger K, Springenberg J, Blum M, Hutter F (eds) (2015) Efficient and robust automated machine learning. In: Proceedings of the 28th International Conference on Neural Information Processing Systems, vol 2. MIT Press, Cambridge, MA, USA

19. Chandola V, Banerjee A, Kumar V (2009) Anomaly detection. ACM Comput Surv 41:1–58. https://doi.org/10.1145/1541880.1541882

20. Ruiz-Carcel C, Starr A (2018) Data-based detection and diagnosis of faults in linear actuators. IEEE Trans Instrum Meas 67:2035–2047. https://doi.org/10.1109/TIM.2018.2814067

21. Zhao S, Huang Y, Wang H et al. (2016) A modified Mahalano-bis-Taguchi system analysis for monitoring of ball screw health assessment. In: 2016 IEEE International Conference on Prognos-tics and Health Management (ICPHM). IEEE, pp 1–7

22. Schmidt B, Gandhi K, Wang L (2018) Diagnosis of machine tools: assessment based on double ball-bar measurements from a popu-lation of similar machines. Procedia CIRP 72:1327–1332. https://doi.org/10.1016/j.procir.2018.03.208

23. Azamfar M, Li X, Lee J (2020) Intelligent ball screw fault diagno-sis using a deep domain adaptation methodology. Mechanism and Machine Theory 151:103932. https://doi.org/10.1016/j.mechmachtheory.2020.103932

24. Pandhare V, Li X, Miller M et al (2021) Intelligent diagnostics for ball screw fault through indirect sensing using deep domain adaptation. IEEE Trans Instrum Meas 70:1–11. https://doi.org/10.1109/TIM.2020.3043512

25. Shih W-C, Furqanuddin F, Lee P-L et al (2021) Monitoring of preload variation of linear guide positioning stage using artificial neural net-work. Appl Sci 11:7902. https://doi.org/10.3390/app11177902

26. Zhang L, Gao H, Wen J et al (2017) A deep learning-based rec-ognition method for degradation monitoring of ball screw with multi-sensor data fusion. Microelectron Reliab 75:215–222. https://doi.org/10.1016/j.microrel.2017.03.038

27. Tornede T, Tornede A, Wever M et al (2020) AutoML for predic-tive maintenance: one tool to RUL them all. In: Gama J, Pashami S, Bifet A et al (eds) IoT Streams for Data-Driven Predictive Maintenance and IoT, Edge, and Mobile for Embedded Machine Learning, vol 1325. Springer International Publishing, Cham, pp 106–118

28. Denkena B, Dittrich M-A, Lindauer M et al (2020) Using AutoML to optimize shape error prediction in milling processes. SSRN Journal. https://doi.org/10.2139/ssrn.3724234

29. Kißkalt D, Mayr A, Lutz B et al (2020) Streamlining the develop-ment of data-driven industrial applications by automated machine learning. Procedia CIRP 93:401–406. https://doi.org/10.1016/j.procir.2020.04.009

30. Hendrickx K, Meert W, Mollet Y et al (2020) A general anom-aly detection framework for fleet-based condition monitor-ing of machines. Mechanical Systems and Signal Processing 139:106585. https://doi.org/10.1016/j.ymssp.2019.106585

31. Benker M, Junker S, Ellinger J et al (2022) Experimental deriva-tion of a condition monitoring test cycle for machine tool feed drives. Prod Eng Res Devel 16:55–64. https://doi.org/10.1007/s11740-021-01085-9

32. Thornton C, Hutter F, Hoos HH et al (2013) Auto-WEKA. In: Grossman RL, Uthurusamy R, Dhillon I et al (eds) Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '13. ACM Press, New York, NY, USA, p 847

33. Komer B, Bergstra J, Eliasmith C (2014) Hyperopt-Sklearn: auto-matic hyperparameter configuration for Scikit-Learn. In: Proceed-ings of the 13th Python in Science Conference. SciPy, pp 32–37

34. Olson RS, Moore JH (2019) TPOT: A tree-based pipeline optimi-zation tool for automating machine learning. In: Hutter F, Kotthoff L, Vanschoren J (eds) Automated Machine Learning. Springer International Publishing, Cham, pp 151–160

35. Sparks ER, Talwalkar A, Franklin MJ et al (2015) TuPAQ: an efficient planner for large-scale predictive analytic queries. arXiv. https://doi.org/10.48550/arXiv.1502.00068

36. Swearingen T, Drevo W, Cyphers B et al (2017 - 2017) ATM: A distributed, collaborative, scalable system for automated machine learning. In: 2017 IEEE International Conference on Big Data (Big Data). IEEE, pp 151–162

37. Wistuba M, Schilling N, Schmidt-Thieme L (2017) Automatic Frankensteining: creating complex ensembles autonomously. In: Chawla N, Wang W (eds) Proceedings of the 2017 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, Philadelphia, PA, pp 741–749. https://doi.org/10.1137/1.9781611974973.83

38. Mohr F, Wever M, Hüllermeier E (2018) ML-Plan: Automated machine learning via hierarchical planning. Mach Learn 107:1495–1515. https://doi.org/10.1007/s10994-018-5735-z

39. Chen B, Wu H, Mo W et al (2018) Autostacker. In: Aguirre H, Takadama K (eds) Proceedings of the genetic and evolutionary computation conference. ACM, New York, NY, USA, pp 402–409. https://doi.org/10.1145/3205455.3205586

40. Drori I, Krishnamurthy Y, Rampin R et al (2018) AlphaD3M: machine learning pipeline synthesis. Mach Learn. https://doi.org/10.48550/arXiv.2111.02508

41. Fusi N, Sheth R, Elibol HM (2017) Probabilistic Matrix Factorization for Automated Machine Learning

42. Hutter F, Hoos HH, Leyton-Brown K (2011) Sequential model-based optimization for general algorithm configuration. In: Coello CAC (ed) Learning and Intelligent Optimization, vol 6683. Springer, Berlin, Heidelberg, pp 507–523

43. Lindauer M, Eggensperger K, Feurer M et al (2022) SMAC3: a versatile Bayesian optimization package for hyperparameter optimization. J Mach Learn Res 23:1–9. https://doi.org/10.48550/arXiv.2109.09831

44. Brochu E, Cora VM, Freitas Nd (2010) A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. https://doi.org/10.48550/arXiv.1012.2599

45. Vanschoren J, van Rijn JN, Bischl B et al (2014) OpenML SIGKDD Explor Newsl 15:49–60. https://doi.org/10.1145/2641190.2641198

46. Waring J, Lindvall C, Umeton R (2020) Automated machine learning: review of the state-of-the-art and opportunities for healthcare. Artif Intell Med 104:101822. https://doi.org/10.1016/j.artmed.2020.101822

47. Jemielniak K (2019) CONTEMPORARY CHALLENGES IN TOOL CONDITION MONITORING. Journal of Machine Engineering 19:48–61. https://doi.org/10.5604/01.3001.0013.0448

48. Christ M, Braun N, Neuffer J et al (2018) Time Series FeatuRe Extraction on basis of Scalable Hypothesis tests (tsfresh – A Python package). Neurocomputing 307:72–77. https://doi.org/10.1016/j.neucom.2018.03.067

49. Denkena B, Dittrich M-A, Noske H et al (2021) Data-based ensemble approach for semi-supervised anomaly detection in machine tool condition monitoring. CIRP J Manuf Sci Technol 35:795–802. https://doi.org/10.1016/j.cirpj.2021.09.003

50. Kriegel H-P, Kroger P, Schubert E et al (2011) Interpreting and unifying outlier scores. In: Liu B, Liu H, Clifton C et al (eds) Proceedings of the 2011 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, Philadelphia, PA, pp 13–24. https://doi.org/10.1137/1.9781611972818.2

51. Seabold S, Perktold J Statsmodels: Econometric and Statistical Modeling with Python:92–96. https://doi.org/10.25080/Majora-92bf1922-011

52. Virtanen P, Gommers R, Oliphant TE et al (2020) SciPy 1.0–Fundamental Algorithms for Scientific Computing in Python. Nat Methods 17:261–272. https://doi.org/10.1038/s41592-019-0686-2

53. Ramaswamy S, Rastogi R, Shim K (2000) Efficient algorithms for mining outliers from large data sets. In: Dunham M, Naughton JF, Chen W et al (eds) Proceedings of the 2000 ACM SIGMOD international conference on Management of data - SIGMOD '00. ACM Press, New York, NY, USA, pp 427–438. https://doi.org/10.1145/342009.335437

54. Brinkhaus J-W (2009) Statistische Verfahren zur selbstlernenden Überwachung spanender Bearbeitungen in Werkzeugmaschinen. Zugl.: Hannover, Univ., Diss., 2008. Berichte aus dem IFW, vol 2009,5. PZH Produktionstechn, Zentrum, Garbsen

55. Zhao Y, Nasrullah Z, Li Z (2019) PyOD: a python toolbox for scalable outlier detection. J Mach Learn Res (JMLR). https://doi.org/10.48550/arXiv.1901.01588