A fault detection strategy for an ePump during EOL tests based on a knowledge-based vibroacoustic tool and supervised machine learning classifiers

Pasquale Borriello · **Fabrizio Tessicini** · **Giuseppe Ricucci** · **Emma Frosina** · **Adolfo Senatore**

Abstract This paper presents a methodology for identifying faulty components in an electric pump during the end-of-line test based on accelerations and pressure pulsation data used to train an ensemble learning algorithm based on supervised machine learning classifiers. Despite various quality control measures in pump manufacturing, some out-of-tolerance components can pass through and end up on the assembly line, potentially leading to premature failure or abnormal noise during real-field operation. Because of the high impact, it is very important to put in place actions to mitigate the risk of delivering non-conform units, even if properly working in terms of pressure-flow rate performances. In this paper, an innovative knowledge-based vibroacoustic tool together with a machine learning built-in Python® library have been used to post-process acceleration and pressure pulsations data to generate features, which are then used to train, and test several supervised machine learning algorithms. The ensemble learning algorithm combines the best classifiers to identify healthy electric pump units with high accuracy, achieving above 95% accuracy in an experimental test campaign carried out on eighty electric pumps. Results are compared using principal component analysis for dimensionality reduction, and a sensor sensitivity study is conducted.

List of symbols

Δp	Delta pressure
σ	Standard deviation
γ	Regularization parameter
AI	Artificial intelligence
AMR	Amplitude of modulation at rotation frequency
AOR	Amplitude of orders of rotation
$ApEn$	Approximate entropy
$BLDC$	Brushless direct current
C	Penalty parameter
cm	Confusion matrix
DOE	Design of experiments
DL	Deep learning
DM	Dummy model
DT	Decision tree
DUT	Device under test
DWT	Discrete wavelet transform

P. Borriello (✉) · A. Senatore
Department of Industrial Engineering, University of Naples Federico II, Via Claudio, 21, 80125 Naples, Italy
e-mail: pasquale.borriello@unina.it

F. Tessicini · G. Ricucci
Fluid-O-Tech S.R.L., Via Leonardo da Vinci, 40, 20094 Corsico, Milan, Italy

E. Frosina
Department of Engineering, University of Sannio, Piazza Guerrazzi, 21, 82100 Benevento, Italy

EGM	External gear machine
EOL	End of line test
ePump	Electric pump
FFT	Fast Fourier transformation
HP	High pressure
KNN	K-nearest neighbors
LLE	Locally linear embedding
LP	Low pressure
ML	Machine learning
MLP	Multilayer perceptrons
MODWPT	Maximal overlap discrete wavelet packet transform
Mtry	Number of the random input variables
NB	Naïve Bayes
NI DAQ	National instruments data acquisition
Ntree	Number of trees
NVH	Noise, vibration, and harshness
PCA	Principal component analysis
PE	Permutation entropy
PACF	Partial autocorrelation function
PHM	Prognosis and health management
Q	Volumetric flow rate
rbf	Radial basis function
RF	Random forest
R&D	Research and development
RMS	Rooth mean square
RUL	Remaining useful life
SDG	Stochastic gradient descent
SVM	Support vector machine
t-SNE	T-distributed stochastic neighbor embedding
TDV	Tooth defect vibration
X	Input variable
$\overline{X}$	Time series average of input variable

1 Introduction

Faulty components within a pump increase the risk of premature failure or unsatisfactory performance. In the context of an electrically driven external gear machine, defects related to gears, bearings, sealings, and components associated with the electric motor are recognized as having a significant impact on performance. A critical aspect is identifying any potential issues from the very beginning of operation. This paper presents a novel methodology designed to identify faulty components in an ePump during end-of-line testing, harnessing acceleration and pressure pulsation data and employing an ensemble learning algorithm. By focusing on the early identification of critical components, this research aims to enhance the reliability and operational effectiveness of electric pumps.

Electric pumps (ePumps) are an important class of low-cost pumps widely used in many applications [1], which spread has increasingly been pushed because of the electrification global trend [2]. Due to challenges in cost and unavoidable randomness in high-volume manufacturing, faulty component detection has been a challenging problem for the industry for a long time [3]. As stated by [4], even if quality control actions are in place, there is always a residual risk to have some non-conform components on the assembly line. Flawed components inside a unit can eventually accelerate the degradation process and premature failures may occur [5]. Besides that, faulty components lead to abnormal pressure pulsation [6], vibration [7, 8], and noise emission [9], which can ultimately alter the normal vibro-acoustic behavior of the machine and be a cause of high annoyance and perceived low quality for the final user.

It is essential for manufacturers then, to identify any anomalies within a pump unit from the very beginning of the operation, and end-of-line (EOL) test is an opportunity to achieve this target. Pump manufacturers generally carry out a functional test of the pump before it goes to the market to confirm one or more aspects of the design or performance. The most common type of EOL test is the pump performance test where the pressure-flow rate is measured, as well as the power necessary for operating the pump to match the requirements [10]. Since steady-state values are of interest, measurements of the flowrate, outlet pressure and absorbed current are normally averaged over the test time, and instantaneous values are not considered. Another critical limitation to acknowledge pertains to the time constraints imposed on these tests. In practice, to avoid impeding the production line, EOL tests are typically allocated a timeframe of less than 10 s. Since not every component has the same influence on the steady-state performances of a hydraulic unit, at the EOL test a pump is qualified as compliant if it can deliver a certain hydraulic performance in the accepted power range, without taking into account its vibroacoustic behavior or considering the possible presence of flawed components inside the unit. For this reason, developing

a procedure that can identify such faulty status without affecting EOL test time and employing common and not invasive sensors [11], such as accelerometers and high frequency pressure sensors, would greatly impact modern manufacturers, improving product quality without impacting costs.

Fault detections, together with Prognosis and Health Management (PHM), are the two fundamental techniques studied and recognized concerning predictive maintenance [12]. Fault diagnosis concerns detecting faults emerging in an equipment component; fault prognosis concerns determining the Remaining Useful Life (RUL). Since this study aims to detect faulty status during the ePump EOL test, more info concerning fault diagnosis techniques will follow. Based on [12], three main categories of fault diagnosis can be found: machine learning (ML), statistical and models based. Model-based techniques rely on explicit mathematical models of the monitored apparatus [13–15]. However, for complex equipment, an exact mathematical model can frequently be impractical. Common techniques for fault diagnosis that make use of condition monitoring data include statistical approaches [16]. Basically, employing different statistical parameters to measure the deviation between test data and reference data [12], the diagnosis can be treated as a null hypothesis test problem [9]. Instead, ML methodologies are based on pattern recognition [17]. Pattern recognition has traditionally been carried out manually using auxiliary graphical tools (such as power spectrum graphs, phase spectrum graphs, cestrum graphs, autoregressive spectrums, spectrograms, wavelet phase graphs, etc.) that, of course, rely on knowledge in the particular area of diagnosis application. IoT sensors have increased the amount of data that is available, and ML techniques having the capacity for handling high dimensional and multivariate data even in complicated apparatus and dynamic situations [18] [19], have nowadays the highest potential. The main drawback, as in general for data-driven approaches, is linked to explanation and reliability issues because of the lack of causality analysis [20].

ML approaches normally involve several steps, concerning selecting historical data, pre-processing, choosing a model, training and validating the model, and maintaining the model [21]. Data pre-processing aims to extract meaningful signal information and reduce data dimensionality.

Time–frequency analysis techniques are widely employed to discern local and transient components within vibration signals [22], demonstrating efficacy in identifying defective components, such as broken impellers, clogged elements, or faulty bearings, with an accuracy exceeding 96% [22]. In the domain of gears [23] and centrifugal pumps [24, 25], diagnostic features, including mean, kurtosis, skewness, variance, and fifth-to-ninth central moment, have been utilized to detect defects in teeth, seals, bearings, and impellers, achieving an accuracy surpassing 90%. Power spectral density and spectral kurtosis represent advanced signal processing methods, proving effective in fault detection. Spectral kurtosis, for instance, exhibits early detection capabilities for gear faults [26], even in noisy backgrounds [27], while power spectral density demonstrates efficacy in hydraulic pump applications [28], accurately identifying worn gears and journal bearings with a precision exceeding 95%. However, spectral-based features encounter limitations, particularly in terms of resolution at higher frequencies. Wavelet transformation addresses this challenge by providing simultaneous frequency and time information, demonstrating success in detecting multifault conditions in centrifugal pumps with high accuracy [29, 30]. Additionally, the Discrete Wavelet Transform (DWT) has been identified as a more accurate method for gear failure detection [31], albeit requiring a sample size in powers of two. The Maximal Overlap Discrete Wavelet Packet Transform (MODWPT) has been introduced and effectively applied to gear [32] and bearing diagnostics. This method eliminates down-sampling steps, offering greater frequency resolution. Across various cases and multiple operating conditions, MODWPT maintains an accuracy exceeding 98% [33]. The right ML algorithm must be chosen as part of the model selection process [11]. In the literature, a variety of ML strategies are discussed, but no one method or algorithm is established for every scenario [34]. Starting from the most ease-of-use and interpretable algorithms such as Naive Bayes (NB) [35, 36] and Bayesian network [37] to the most popular k nearest neighbors' algorithm (KNN) [38], Support Vector Machine (SVM) [39–41] and random forest (RF) [42], they all are used to address fault detection of rotary machines with good accuracy. Multilayer Perceptrons (MLP), a type of feedforward neural network with backpropagation, have emerged as the

most widely used neural network model for classification and pattern recognition in recent times. MLPs are extensively applied in fault detection scenarios [43]. Other neural network architectures, including recurrent neural networks [44, 45], various deep learning models [46], and convolutional neural networks [47], have demonstrated successful results across industrial sectors, particularly in detecting bearing defects— a prevalent failure mode in gearboxes and hydraulic machines. Quite significantly, the employment of Deep Learning (DL) across many disciplines thanks to its superior learning abilities may be the future trajectory for fault diagnosis research. On the flip side, both ML and DL methodologies suffer from limitations by their reliance on big data, the requirement for powerful computing, high dependence on excessive parameters, and interpretability issues [48, 49]. For these reasons, to improve diagnostic effectiveness, future diagnostic systems should closely incorporate not only data-driven artificial intelligence (AI) technologies but also the analysis of failure processes based prior knowledge of the system under study.

To overcome these limitations, a new methodology is proposed in this work: a classic data-driven approach is used together with an innovative vibroacoustic tool with prior knowledge for feature extraction. Data are used to independently train several supervised ML algorithms, and the most promising ones are combined to build an ensemble learning algorithm, which has been proven to be more robust and powerful than individual classifiers [50]. The employment of such a knowledge-based tool for feature extraction and an explicit classifier such as RF gives the user easy access to results, increasing their interpretability and phenomena understanding. Also, in contrast to limitations observed in prior studies, the proposed methodology overcomes the common constraint of examining isolated single defects. Within this work, the methodology identifies four distinct faulty scenarios, extending the analysis to encompass the simultaneous presence of two faults. This expanded approach involves the systematic design of prototypes and subsequent experimentation based on established design of experiments methodologies. This approach enhances robustness and provides a more realistic exploration of complex real-world scenarios.

The rest of the paper is organized as follows: under Sect. 2, the theoretical background on external gear machines, data pre-processing, and supervised machine learning is provided. Section 3 presents the experimental study, including the reference machine, choice of factors, design of experiments (DOE), and experimental test setup. The proposed fault detection methodology is discussed in Sect. 4, which includes the feature generation methodologies and various machine learning methods employed. The key findings are presented in Sect. 5, and lastly, in Sect. 6, conclusions are drawn based on the findings.

2 Theoretical background

In this section an introduction relative to the external gear machines, data preprocessing, and the ML classifiers is given to provide the reader of basic concepts related to the discussed topics.

2.1 External gear machines

In a variety of applications, including those in the fluid handling, aerospace, automotive, construction, and agricultural industries, external gear machines (EGMs) are critically important. These units are used widely because of their small sizes, strong reliability, high efficiency, low price, and ease of production. The designs of EGMs that are most frequently seen on the market today are usually distinguished by having a minimal number of internal components to properly fulfill their role as positive displacement machines. The depiction in Fig. 1a can be used to better understand how a typical non pressure compensated EGM unit pump works.

Low-pressure (LP) fluid at the unit's suction side is displaced as high-pressure (HP) fluid at the delivery side as a result of the driver and driven gears meshing. Thus, at the outlet, the external mechanical energy supplied by an external source, such as an electric motor, to the driving gear shaft is transformed into high-pressure fluid energy. To prevent wear and heat buildup, the fluid should ideally pump while also acting as a lubricant for the gears and major parts. The two primary lubricant interfaces are depicted in Fig. 1b. These interfaces are an important design element of an EGM that must take into account the complex physical phenomena involving structural, thermal, hydrostatic, and hydrodynamic impacts that

affect the dependability, operational efficiency, and life expectancy [51].

2.2 Data preprocessing

When input numerical properties have widely dissimilar scales, ML algorithms typically perform poorly [52] and a data preprocessing step has a high impact on the final model performances [52]. In general, normalization and standardization are two methods used to ensure that all properties have the same scale. Each attribute's values are shifted and rescaled during normalization such that they ultimately range from zero to one, as shown in Eq. 1:

$$X_{normal} = (X - X_{min})/(X_{max} - X_{min}) \tag{1}$$

Standardization instead, first subtracts the mean value ($\overline{X}$), then divides the result by the standard deviation (σ), as in Eq. 2:

$$X_{stand} = (X - \overline{X})/\sigma \tag{2}$$

Values are not constrained to a certain range by standardization and is much less affected by outliers. However, when a feature's distribution has a heavy tail, both min–max scaling and standardization will restrict most values into a small range resulting in a loss of information, so other transformations are required to make the distribution roughly symmetrical. For example, a common way to do this for positive features with a heavy tail is to replace the feature with its square root raise the feature to power between zero and one, or even replace the feature with its logarithm may help.

2.3 Most popular supervised ML classifiers

ML classifiers have become a fundamental tool in solving classification problems across many fields. These classifiers aim to get acquainted with the input–output mapping of features and labels through various training algorithms, enabling them to classify new data points accurately. A more comprehensive theoretical foundation for the most widely used ML algorithms for diagnostics and condition monitoring is provided in Appendix A. In the end, ensemble learning, a widely recognized technique, is employed to combine multiple models, thereby enhancing predictive performance. In Sect. 5.2, a Voting Classifier ensemble method is

utilized. This method amalgamates the predictions of several distinct classifiers. The ensemble learning process involves the independent training of each classifier on the dataset. During prediction, the input data is simultaneously presented to all three classifiers, producing individual predictions. These predictions are subsequently consolidated via a majority vote mechanism, yielding the final prediction. It is important to build the ensemble model by combining different type of classifiers to efficacy enhance accuracy, reduce overfitting, and increase model robustness.

3 Experimental study

This section provides information about the experimental investigation, including specifications of the reference machine, the choice of factors, the DOE plan, and the experimental test setup.

3.1 Reference machine

As previously mentioned, an external gear unit driven by a brushless DC (BLDC) electric motor with magnetic coupling served as the study's reference machine. The technical specifications of the machines used in this study can be found in Table 1. Figure 2 illustrates the three primary components of the pump head: the front body, cavity plate, and rear body. Both gear shafts are supported by several bushings that are installed in the front and rear pump bodies to ensure smooth operation.

Overall, the reference machine is a reliable and efficient ePump commonly used in various applications [53]. By employing this machine in this study, it is possible to obtain accurate and consistent experimental data to analyze the performance and characteristics of the pump.

3.2 Choice of factors

As emphasized in the introduction, the proposed methodology is designed to identify faulty components in ePumps that may escape detection through traditional EOL tests, which typically focus on steady-state performances. The selection of factors considered in this study was a meticulous process guided by the pump manufacturer, taking into account quality controls and residual risks associated with machining and assembly procedures. The chosen factors, explicitly described below, stem

from real-world scenarios and have been selected because manufacturers have demonstrated that detecting them during EOL tests would mitigate the risk, especially in high-volume manufacturing. This mitigation is crucial to avoid delivering machines to the market that might experience premature failure and/or unsatisfactory noise, vibration, and harshness (NVH) properties, potentially leading to customer claims.The identified faults, crucial to this investigation, encompass four distinct scenarios:

A. Missing one bushing in the front body: the absence of a bushing in the front body, as commonly encountered during the assembly process, reflects a real-world risk. This fault aligns with previous studies emphasizing the significance of defects in bushings or bearings as primary causes of pump failure and degradation [54].

B. Missing one bushing in the rear body: similarly, the absence of a bushing in the rear body replicates assembly errors, contributing to manufacturer's understanding of potential failure modes during the assembly process.

C. Drive or driven gear with a tooth defect: the geometry of the gear teeth has a direct impact on pump pressure pulsation [55] and accelerations [56]. To simulate this fault, a gear profile with a shape deviating from the nominal one was chosen, reflecting real-world scenarios involving shrinkage effects during injection molding or accidental faults during logistics operations [57].

D. Driving magnet not correctly magnetized: in the context of an external gear pump driven by a BLDC motor with a magnetic coupling, the magnetic characteristics of the driving magnet are pivotal for ensuring stable behavior and preventing issues related to NVH phenomena. Deviations from nominal values due to manufacturing errors can lead to non-stable behavior

and potential component failure [58]. Consequently, driving magnets with magnetic properties outside the nominal range were selected."

3.3 Design of experiments

The experimental test campaign was designed using the DOE methodology to minimize the number of experiments and identify interactions among factors. A fractional Factorial Design with replication and randomization was chosen to examine the interactions between the four chosen factors and their effects. Specifically, a half fractional factorial design with Resolution IV was used to assess the effect of main factors and 2-main factor interaction while neglecting the effect of 3-factor interaction considered as less likely to be influent [59]. The presence or absence of flaws in each pump was recorded in Table 2, with a total of eighty samples tested across sixteen physical ePumps, each tested five times and treated as a distinct unit. In situations where there is limited data available, it is not uncommon to treat each test on the same experimental unit as a separate sample to enlarge the sample collected and increase the model's precision, even considering the randomness and noise introduced with experiments [60, 61].

Randomizing test serves to introduce additional variability and this randomness reflects the natural variation introduced by different operators during the testing process. Operators inherently contribute to variability in how they handle and place devices on the test rig. Therefore, treating these repeated tests as separate samples not only captures the inherent noise in the experimental data but also simulates the practical scenario where different operators may conduct tests. To further validate this approach, an ablation study has been conducted excluding from the training

Fig. 1 **a** The operating mode of an EGP; **b** radial and axial lubricant interfaces of an EXM
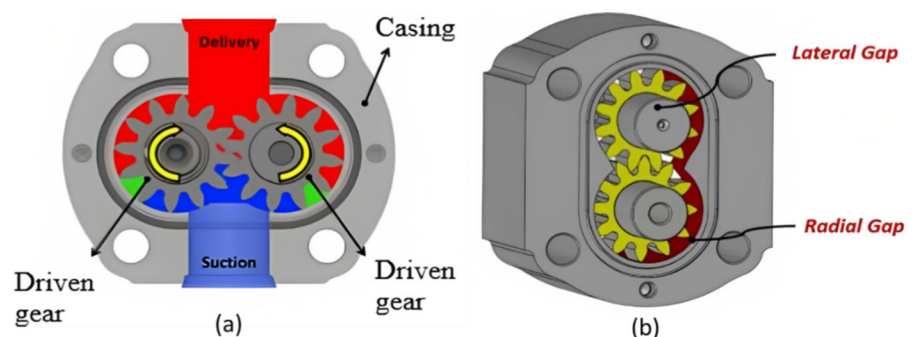
set specific defects one-at-time and results are presented in Sect. 4.2.

In order to conduct the experiments, each machine was meticulously assembled using components that were entirely measured, ensuring precise control over the factors under investigation. This also highlights the needed effort to produce prototypes and support the choice of using augmentation techniques using the intrinsic noise when performing real experiments. The experimental setup included the control of defectivity in three critical components: bushings, gears, and magnets. The control of bushings defectivity was executed in a binary manner. To replicate gear defects, the process initiated with gear components falling within specified tolerance limits. Subsequently, a controlled defect was introduced by manually reducing the size of a single gear tooth. For the magnet component, defectivity was implemented by selecting a magnetization level that deviated from the accepted tolerance range. These methods were employed to ensure a rigorous and controlled implementation of factors and defects within the experimental pump units. This meticulous approach enhances the reliability and reproducibility of the study's results."

3.4 Experimental test rig

A specific experimental test campaign was carried out at the laboratory of the company Fluid-o-Tech, the pump's manufacturer, using the test rig shown in Fig. 3. Each ePump unit was equipped with two accelerometers (axial and radial), a high-frequency dynamic pressure sensor, and a static pressure sensor. Details about each sensor are provided in Table 3. The experimental setup used a high-stiffness constant internal diameter duct at the outlet side, and a calibrated orifice was put in place to replicate the load [62]. EPump units were tested on a specific hydraulic operating condition, which represented the project requirement. Each unit was run to reach a specific pressure upstream of the calibrated orifice and was tested at the same hydraulic point (Q, Δp), resulting

in a slightly different motor speed and power absorption to consider for manufacturing tolerances. In other words, the acceptance criteria corresponded to a certain motor speed and power demand range for that specific hydraulic working condition. The proposed scenario mimics the real-word scenario considering the hardware and methodology normally used during EOL test. As said, randomizing test also capture the scenario where different operators may conduct test apart from considering the natural noise from experiments.

4 Proposed fault detection model

Figure 4 uses a flowchart to demonstrate the proposed fault detection procedure. Data are acquired using a National Instruments Data Acquisition (NI DAQ) system, and time series data of pressure and acceleration signals are processed through PeiVMGears® and tsfresh, which are further described in Sects. 4.1 and 4.2 to generate features. The problem of excessive computational time is addressed by feature extraction, which lower the dimensionality of the problem. The "feature engineering" step involves data transformation and feature reduction, the former of which was introduced in Sect. 2, and the latter is discussed in the next section. The entire data set is then compiled. The training set is then created using 70% of the samples, while the test set is made using the remaining 30% of the samples, which will be used to validate the proposed model. As described in Sect. 4.3, several ML algorithms are trained using the training set, and the most promising algorithms are identified using cross-validation. The hyper-parameters of the selected algorithms are optimized and used to construct the ensemble learning algorithm, which is the final model. Finally, the test set data, which represents unseen data for the model, is used for validation and results visualization.

Table 1 Technical specifications relative to experimental units			
Pump housing material	AISI 316L	Motor type	24V BLDC
Gears material	PEEK	Speed Range	500:5000 rpm
Max static Pressure	20 bar	Max Output Power	60 W
Displacement	0.6 cc/rev	Unit Weight	675 g

Fig. 2 Reference motor
pump unit
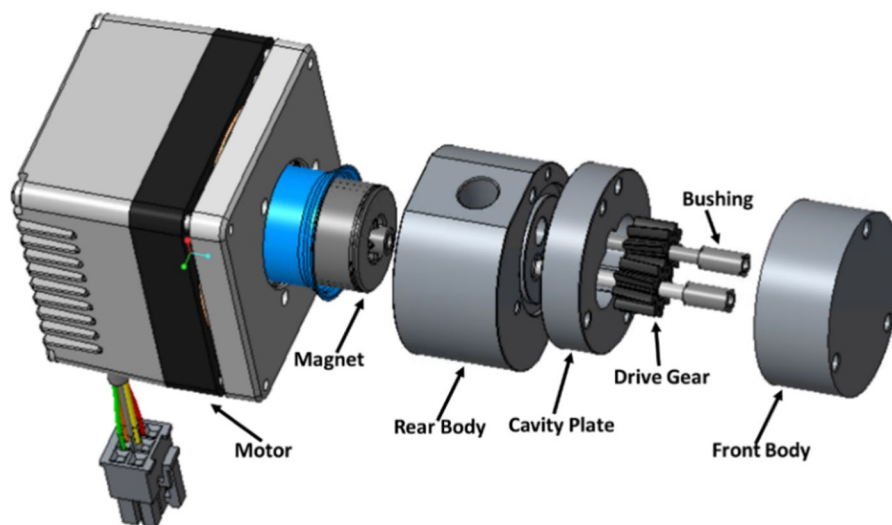
4.1 Feature extraction using tsfresh

The ML library tsfresh, which is quick and standardized for automatic time series feature extraction and selection [63], was used for feature extraction. In order to promptly extract and investigate different time series attributes, determine their statistical significance, and foresee the target, time series feature extraction is crucial in the early stages of data science projects. The Python package tsfresh (Time Series FeatuRe Extraction on the basis of Scalable Hypothesis tests) automates the process by combining feature selection based on automatically configured hypothesis tests with 63 time series characterization methods, which by default computes a total of 794 time series features. The library has been used for industrial big data applications [64] and several other cases involving pattern recognition and ML methodologies [65–67].

Originally, the data recorded from the dynamic pressure sensor, radial and axial accelerometers generated a total of 2,190 features. These features encompass various statistical measures, including maximum, minimum, root mean square, mean, kurtosis, absolute energy, standard deviation, FFT coefficients, continuous Wavelet transformation coefficients and more. A detailed mathematical description of each feature can be found in reference [63]. For brevity, in Sect. 5.1, a detailed description of the three most significant features for the machine learning model has been provided.

To reduce the dimensionality of the problem, a cross-correlation analysis of the generated features with respect to the labeling has been performed based on the Spearman method [68]. The number of features was in this way reduced to 191 by selecting the ones with a Spearman coefficient higher than 0.3. The cross-correlation matrix of the remaining metrics using the Pearson method is shown in Fig. 5, which demonstrates a relatively important number of highly correlated features. This can be explained due to the fact that the same analytical procedure has been done for each of the three time series of data coming from the pressure and accelerometer signals and so their values are not independent from each other.

4.2 Feature extraction using PeiVMGears®

PeiVMGears® is a software tool for sound and vibration analysis that enables advanced diagnostic operations in R&D, quality controls, maintenance, and remote controlling fields [69]. It can identify abnormal behaviors and faults related to rotating components, such as gearboxes, gears, bushings, bearings, and eccentricity defects. PeiVMGears® works as a DAQ system and manages the acquisition of data from different sensors. It provides both raw data and post-processed data specific to the model under study. Figure 6a shows the data process workflow, and Fig. 6b shows an example of the model set-up for a two coaxial stages gearbox. Basically, it is crucial to carefully set up the kinematic chain of interest to

Table 2 Detail of defects inside experimental units

# Pumps	A	B	C	D
25	0	0	0	0
5	1	0	0	0
5	0	1	0	0
5	0	0	1	0
5	0	0	0	1
5	0	1	0	1
5	1	1	0	0
5	1	1	1	1
5	0	1	1	0
5	0	0	1	1
5	1	0	0	1
5	1	0	1	0

(A) one bushing missing on the front body; (B) one bushing missing on the rear body; (C) tooth defect; (D) magnet defect

provide the right inputs for identifying the frequencies of interest for the specific problem. In this study, the pump head has been considered as a single-stage transmission with two equal gears to capture the mechanical behavior of each device under test. With this set-up, 33 features were generated. These features are calculated by PeiVMGears based on the kinematic chain, and all feature definitions can be found in the user manual [69]. Some of these features include RMS values for acceleration and pressure data, RMS ISO according to ISO010816, and specific ones developed by software's provider. These specific features relate to damage in rolling elements, cumulative gear vibration, relative eccentricity of the gears, relative teeth damage, maximum tooth damage vibration, and tooth quality vibration. For the sake of brevity, in the manuscript's Sect. 5.1, the mathematical descriptions of the three most important features relevant to the machine learning model is presented.

To reduce the dimensionality of the problem, a cross-correlation analysis of the generated features with respect to the labeling has been performed based on the Spearman method, as mentioned in a. Only the features whose correlation number is above a threshold of 0.3 have been considered. The histogram relative to each of the 13 remaining features is reported in Fig. 7. As visible, none of the features have the potential to clearly distinguish between the two classes (healthy and faulty pumps) recalling the use of accurate ML classifiers

to help finding hidden patterns [17]. Moreover, similar trends among different features are visible, and some feature preprocessing is needed for those features that present heavy tails (skewed), as explained in Sect. 2.2.

4.3 Shortlist of several ML models

Once the data set is pre-processed, it is customary to explore a variety of ML models using standard parameters and compare their performances, in accordance with established practices [52]. Scikit-learn, a free Python software package for machine learning that supports both supervised and unsupervised learning, has been used for the creation of ML algorithms [70]. Figure 8 displays the confusion matrix (cm) for each of the models taken into consideration.

Results have been obtained on training data using threefold cross-validation, and data transformation is applied to SVM, Stochastic Gradient Descent (SDG), and KNN classifiers, but not to Dummy Model (DM), Decision Tree (DT), and RF classifiers since they are not sensible to input numerical variables ranging on different scales. Test data are not utilized at this stage to explore the capabilities of each model and avoid biasing the models on test data. It can be observed how each model behaves differently on the training data. For instance, the Dummy Classifier (a) predicts only unhealthy pumps, which are the most common instances. At this stage, it is important to select the most promising models to ensure that the ensemble learning approach applied later can average out different types of errors and biases related to each of the chosen models. RF, SVM, and KNN show better accuracy (percentage of correct classifications that the model achieves) and are selected for the next stage. As described by the flow chart (Fig. 4), the next step involves hyperparameter optimization for the selected ML models (SVM, KNN, and RF).

4.4 Hyperparameter optimization

In the context of ML, hyper-parameters refer to parameters that are not learned directly by the estimator but are critical for achieving high accuracy in the models. This section presents and analyzes the hyperparameters utilized to optimize each model's performance.

Fig. 3 Test rig at Fluid-o-Tech laboratory: **1** tank, **2** filter, **3** static pressure sensor, **4** National Instruments Data Acquisition, **5** dynamic pressure sensor, **6** device under test, **7** axial accelerometer (the radial is not visible)
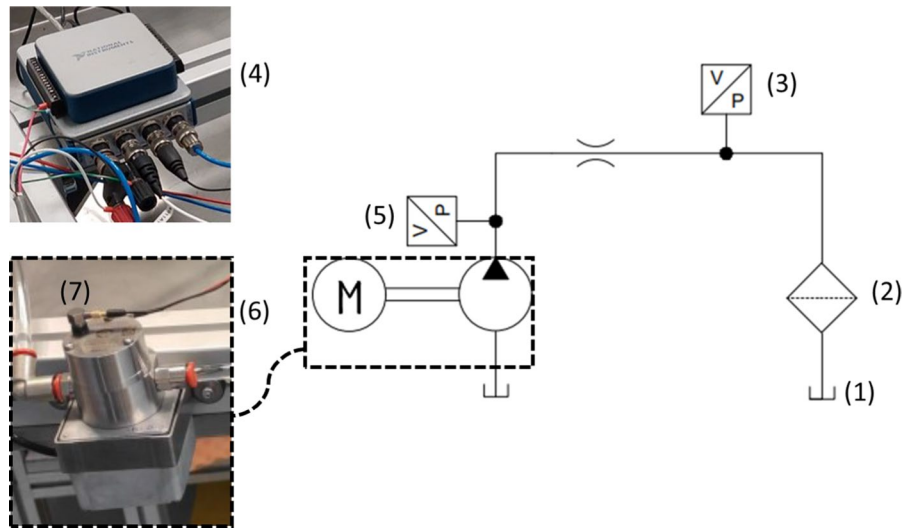
Table 3 Detail of the sensors used in the test setup

Sensor	Model	Specifications
Axial accelerometer	DYTRAN3225M24T	F.S.: 50 g, Sens.: 104.38 mV/g, Lin.: 2% F.S
Radial accelerometer	PCB M352C65	F.S.: 50 g, Sens.: 101.1 mV/g, Lin.: 2% F.S
Dynamic pressure	PCB WS113B28	F.S.: 3.5 bar, Sens.: 13.98 mV/kPa, Lin.: 0.5%F.S
Static pressure	UNIK 5000	F.S.: 25 bar, Sens.: 0.4 V/bar, Lin.: 0.1% F.S
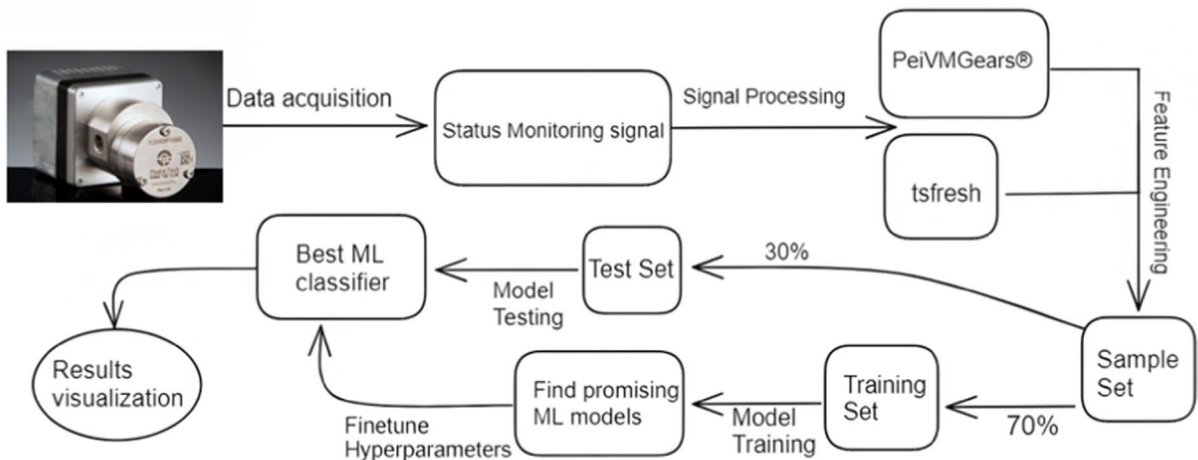
Fig. 4 Flow chart of the proposed fault detection strategy

4.4.1 Support vector machine: SVM

Scikit-learn offers a number of SVM classes with several kernels for the SVM model, including linear, polynomial, and Gaussian. To optimize the parameters, the hyper-parameter space was explored to achieve the best cross-validation score. Two methods were used in sklearn to pursue hyper-parameter optimization. While RandomizedSearchCV picks a predetermined number of candidates from a parameter space

with a defined distribution, GridSearchCV [71] thoroughly considers all parameter combinations for given values. GridSearchCV was utilized in this study, and a range of hyperparameters was explored using cross-correlation with three folders on the training set. Table 4 reports the set of hyperparameters considered, and their optimization results are presented. The misclassification or error term is represented by the penalty parameter C, which represents the SVM optimization's tolerance for deviation. When C is elevated, every single data point is accurately identified, although overfitting is a possibility. The Gamma parameter determines the range of influence of each instance with respect to the decision boundary. Thus, it acts as a regularization hyperparameter, controlling the model's level of overfitting or underfitting.

4.4.2 K-nearest neighbor: KNN

The K value's ideal selection for the KNN model heavily depends on the input information. Generally, larger K values suppress the effects of noise but result in less distinct classification boundaries. Figure 9 displays the accuracy in predicting the training data for several K values.

The highest accuracy at low K suggests a pretty clear distinction between the two classes of faulty and healthy units. The model's hyperparameters were optimized using GridSearchCV, and the results are shown in Table 4. The basic nearest neighbors classification uses uniform weights, and the "metric" concerns the methodology used for distance computation.

4.4.3 Random forest: RF

The most important hyperparameters for the RF model are thought to be the number of trees (Ntree) and the number of randomly chosen variables from the input features (Mtry) [72, 73]. Guan et al. [74] stated that as many Ntree values are possible, although Belgiu and Drăgut [73] and Gislason et al. [75] suggested 500 as a default number, relying on the RF classifier's robustness and resistance to overfitting. The square root of the number of input variables or log base two are frequent values for Mtry [75, 80]. However, establishing Mtry to the quantity of input variables [37] can impact the algorithm's speed,

as all data must be evaluated to split the nodes [76]. GridSearchCV was employed to optimize the hyperparameters, and the results are presented in Table 4. The "max depth" limits the leaf of each tree, and "Max_leaf_nodes" restricts how many leaf nodes a tree can have in total, and they both were limited to 9 in this study to avoid overfitting. As a default, "Gini" was used as a criterion to perform split operations and "Better" as a splitter strategy of how to split a node [71].

5 Results

According to the methodology shown in Fig. 4, firstly the most promising classifier were selected and their hyperparameter optimized using training data. In this paragraph, the results using test data are discussed focusing on each individual models, followed by an analysis of the ensemble learning algorithm performance.

5.1 Individual models

The confusion matrix for the test data regarding the three most promising models identified in sect. 4.3 is displayed in Fig. 10.

Results have been presented for three different scenarios, namely original data (I), data transformed (II), and data transformed with optimized hyperparameters (III), as discussed in Sects. 4.1, 4.2, and 4.3. Details concerning each model are reported in Table 5, where results are divided into training and test data. Notably, results concerning training data were obtained using threefold cross-validation and averaging the results. It is worth noting that SVC and KNN, as per the literature, are sensitive to data on different scales, and thus transforming the data is essential to obtain good results. In fact, as better detailed in the Appendix section, in the case of KNN, which relies on distance metrics to determine proximity between data points, features with larger scales can disproportionately influence the calculation of distances. Consequently, features with smaller scales may contribute negligibly to the distance computation, leading to a biased influence of certain variables in the classification process. Similarly, SVC, particularly when employing radial basis function (rbf) kernels, is sensitive to input feature scales. The rbf kernel considers the distance

**Fig. 5** Cross-correlation matrix of feature extracted by using tsfresh
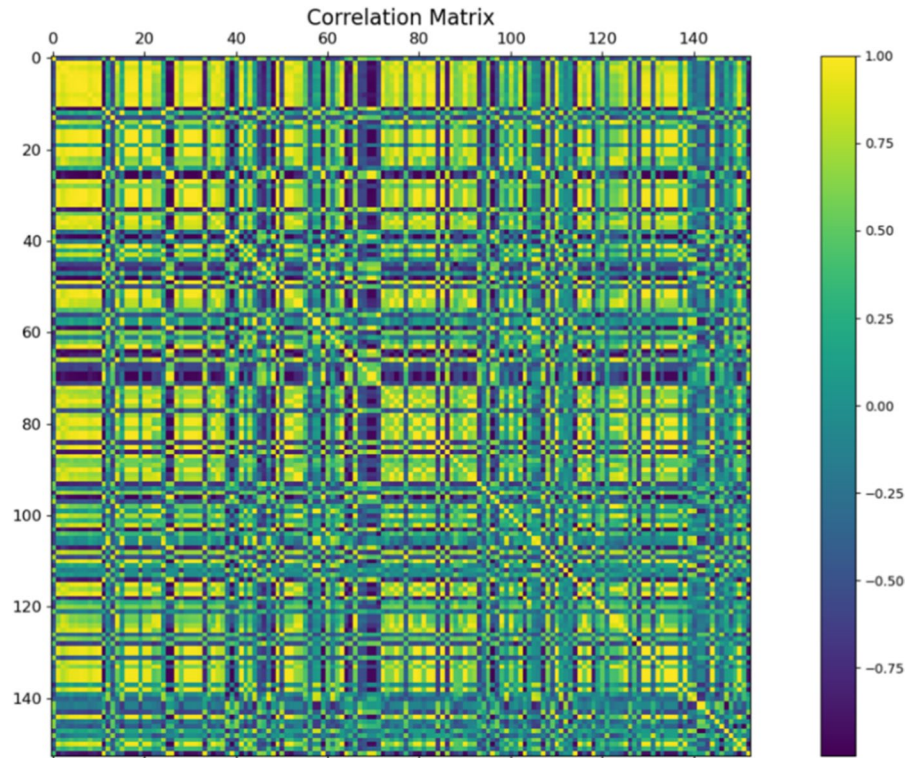
**Fig. 6 a** VMGears® data acquisition and analysis workflow: **b** example of the kinematic chain setup
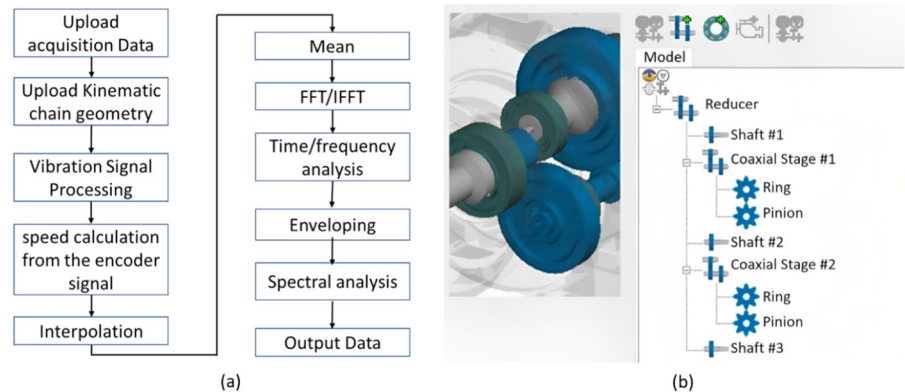
between data points in feature space, and when features are on different scales, the impact of features with larger scales dominates the computation of the decision function. This dominance may lead to suboptimal performance and adversely affect the model's ability to generalize to unseen data.

The effect of hyperparameter tuning is also noticeable, but transforming the data properly has the most significant impact when using such models. Conversely,

when looking at RF results, two insights can be drawn: scaling is not effective—as expected, and hyperparameter tuning worsens the accuracy on the training set. This can be attributed to the RF model's high power, which in this case overfits the training data and loses accuracy after optimizing their hyperparameters. Indeed, RF builds multiple decision trees during training, each contributing to the final outcome. For its nature, RF shines in handling noisy features, implicitly selecting
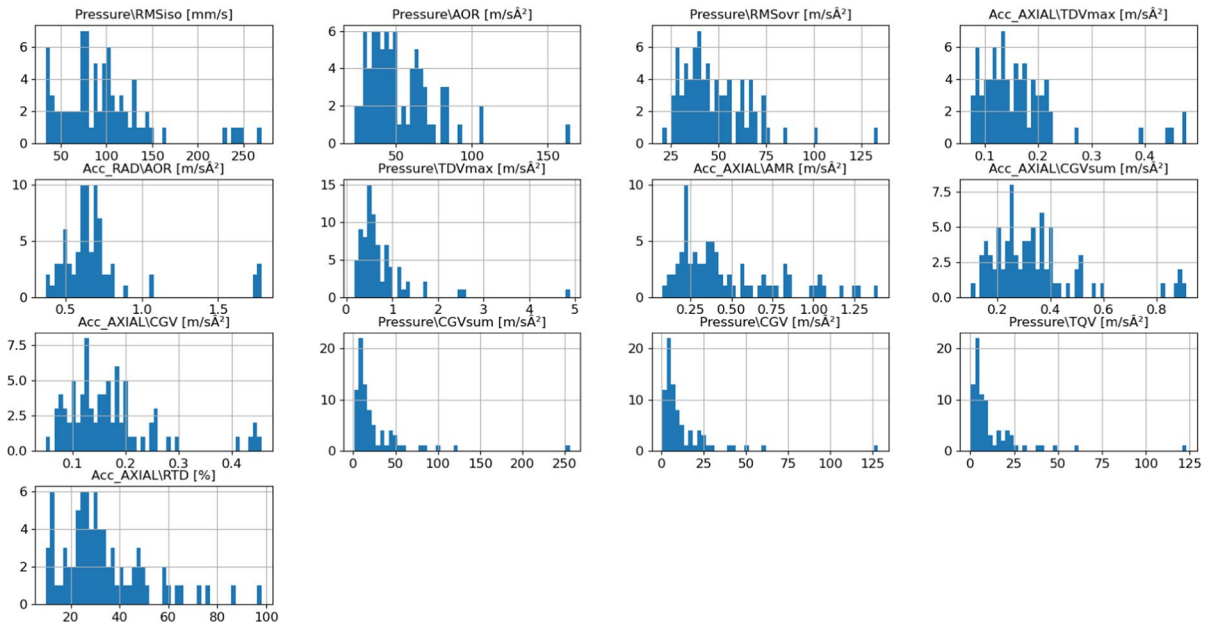
Fig. 7 Histogram of the most correlated features generated using PeiVMGears®
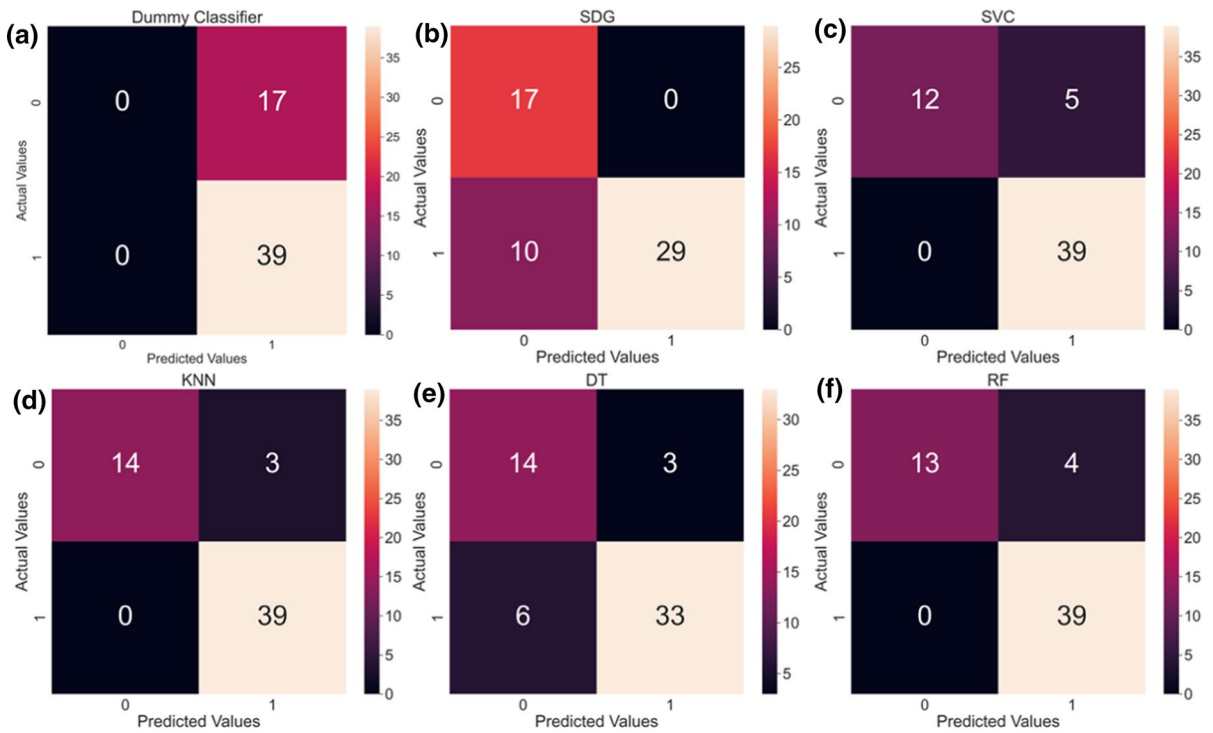
Fig. 8 Confusion matrix by **a** Dummy Classifier, **b** SDG, **c** SVM-Lin, **d** KNN, **e** DT, **g** RF

important ones, and decoding complex non-linear data relationship. However, the risk of overfitting is high, causing the model to learn training data details and noise rather than catching the actual pattern. This has the risk of poorly performing on unseen data. As better detailed in the Appendix section, the numbers of trees, and primarily the number of leaves and depth of trees have to be carefully set and tuned.

One of the significant advantages of using a classifier such as RF is the possibility to examine the most important features for the model. Among some well-established features for fault detection, such as maximum, minimum, root mean square, mean, kurtosis, absolute energy, and standard deviation—as described in Sect. 1, there are some peculiar ones specific of the feature extraction methodologies proposed in this study.

Features extracted using PeiVMGears®, named *"AOR", "TDV", and "AMR",* are among the most important features for the model and are further detailed:

"AOR"—Amplitude of Orders of Rotation [m/s^2]—represents the cumulative amplitude of the first 10 rotation orders of the electric motor, given by:

Z stands for the number of gear teeth, and Ampl_RotX is the amplitude associated with the examined gear's Xth rotational frequency.

"AMR"—Amplitude of Modulation at Rotation Frequency [m/s^2]—is the amplitude of the modulating effect produced by gear rotation frequency in the range (1000:6000) Hz.
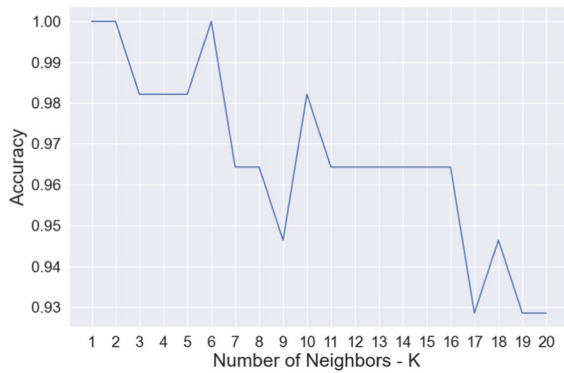
Similarly, among the most important features, the first three features extracted using TsFresh are:

"Approximate_Entropy" (ApEn): a probabilistic evaluation that assesses the regularity or predictability of a temporal series of data. It calculates the logarithmic probability that similar patterns of data points will remain similar when the data is expanded by one additional data point. ApEn is used to identify changes in the pattern or complexity of the time series data, which can indicate the presence of faults or other anomalies in the system. The vibration signal will have a rise in the total amount of frequency components as the machine deteriorates as a result of the development and/or progression of structural faults, which will cause a drop in the signal's regularity and an increase in the

$$AOR = \sqrt{Ampl\_Ord\_1^2 + Ampl\_Ord\_2^2 + \cdots + Ampl\_Ord\_10^2} \qquad (3)$$

"TDV"—Tooth Defect Vibration [m/s^2]—represents the maximum vibration peak in the angle-base averaged vibration signal, once removed the meshing frequencies, detailed in Eq. 4:

related ApEn value. It has been demonstrated to be a useful tool for determining a machine's condition

$$TDV = (Ampl_{Rot1}^2 + Ampl_{Rot2}^2 + \ldots + +Ampl_{Rot(Z-1)}^2 + Ampl_{Rot(Z+1)}^2 + + \ldots \\ + Ampl_{Rot(10Z-1)}^2 + Ampl_{Rot(10Z+1)}^2)^{1/2} \qquad (4)$$

	Hyper-parameters		Optimized value
SVC	C	[0.1, 1, 10, 100, 1000]	10
	Gamma	[1, 0.1, 0.01, 0.001, 0.0001]	0.001
	Kernel	['rbf', 'poly', 'sigmoid']	'rbf'
KNN	n_neighbors	[3, 5, 7, 9, 11, 13, 15]	25
	weights	['uniform','distance']	'uniform'
	metric	['minkowski','euclidean','manhattan']	'minkowski'
RF	Ntree	[100, 250, 500, 1000, None]	500
	Max_depth	[3, 6, 9]	3
	Mtry	[None, 'sqrt', 'log2']	'log2'
	Max_leaf_nodes	[3, 6, 9]	6

Table 4 Hyperparameters considered in the optimization problem for SVC, KNN and RF

Fig. 9 KNN accuracy on train data for different K values

of health, moreover concerning bearing defects [77, 78].

Recent research has shown that related entropy theories, including "Permutation_Entropy" (PE), which may reflect the dynamic characteristics of the equipment under observation by measuring the complexity of the chaotic noise, are useful in monitoring the state of rotating machinery [79]. PE's key benefits are great calculation efficiency and resistance to noise, which can be utilized to detect changes in the system's underlying dynamics that may be signs of failures or other anomalies [80].

- *"spkt_welch_density"* calculates the time series' cross-power spectral density at various frequencies. By analyzing the power spectral density of vibration or current signals, it has proven effective in detecting faults and condition monitoring of rotary machines to find specific faults like bearing wear, imbalance, misalignment, and permanent magnet failures [81–83].
- *"partial_autocorrelation"* is a statistical technique for figuring out the relationship between a variable and its lagged values while accounting for the impact of other variables. It assesses the degree of correlation between a variable and its prior values after factoring in for the effects of the other variables. Partial autocorrelation has been used in fault detection and condition monitoring of rotary machines to identify the presence of specific types of faults that can affect the machine's performance. For example, in a healthy rotary machine, the Partial AutoCorrelation Function (PACF) will generally show a slow decay in corre-

lation over time, reflecting the natural damping of vibrations in the machine. However, in a machine with a fault, the PACF may show sharp peaks or dips at specific lag times, indicating the presence of resonance or other fault-related vibration patterns [84–86].

The proposed workflow's reliability and robustness are supported by the use of the most important features for the RF classifier, which have been applied to fault prognosis and condition monitoring of rotary machines. This demonstrates that the described approach can capture the physical nature of the analyzed faults. Additionally, the extraction of three features using PeiVMGears® provides valuable insight into the significance and potential benefits of utilizing a tool that incorporates prior knowledge of the machine.

In general, by analyzing the nature of the most important feature for the model, insightful consideration concerning the failure mode could be drained. For example, one of the most important feature for the model is the TDV, as discussed above. This feature, could uniquely lead to a defectivity concerning one or several defective teeth, giving to the manufacturer precise indications. Actually, since the models were trained to distinguish between healthy and faulty machines, broader observations can be derived. Through the analysis of features such as partial autocorrelation, spkt welch density, and entropy-based metrics, as extensively supported in the literature, this study confirms that faulty machines manifest distinct components in the frequency domain and power spectrum. Perhaps, a notable advantage of the proposed methodology lies in its capability to analyze the most crucial features for the specific machine and issue post-training. This enables the machine learning model to discern patterns and rationalize the factors contributing to the faults.

This approach can aid in phenomena understanding and causal analysis, thereby overcoming the primary limitation of a purely data-driven approach.

5.2 Ensemble learning

Ensemble learning is a technique that aggregates the forecasts made by a group of predictors, referred to as an ensemble, to form a more accurate prediction than any single predictor could achieve. This approach is
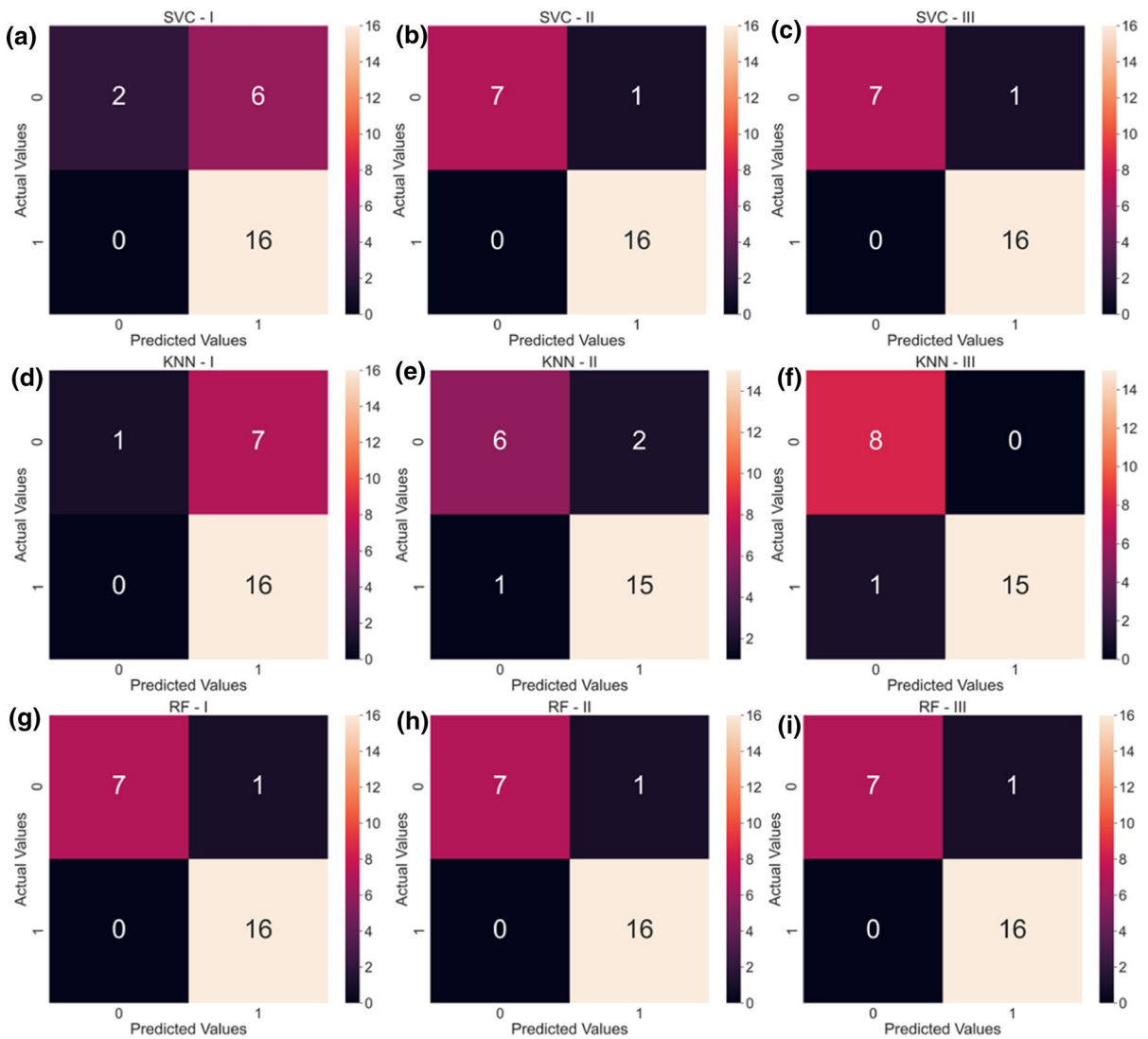
Fig. 10 Confusion matrix for **a** SVC—I, **b** SVC—II, **c** SVC—III, **d** KNN—I, **e** KNN—II, **f** KNN—III, **e** RF—I, **f** RF—II, **g** RF—III

based on the wisdom of the crowd concept, where the collective intelligence of a group can be more effective than that of just one expert [52].

In the context of ML, ensemble methods are often used to improve the performance of classifiers. One common approach involves aggregating the predictions of individual classifiers through a voting process. A hard-voting classifier selects the class with the most votes (as shown in Fig. 11), while a soft-voting classifier estimates class probabilities and uses the highest average probability across all classifiers to predict the class. Soft-voting is typically more

effective than hard voting because it provides votes from voters who are confident with greater weight.

Given the high accuracy of the individual models, as shown in Table 5, an in-depth investigation was conducted to prove the overall better efficacy of the ensemble learning technique over the single classifiers. This ablation study consisted of intentionally excluding one at time from the training set all samples related to a unique defect (A, B, C, D), as specified in Table 2, and five healthy samples (H) The dataset was then divided into 70–30% and the removed samples were added to the test set. The performance of each

Table 5 Metrics concerning SVC, KNN and RF models

| Model | Data | Accuracy | | Precision | | Recall | | Fl-Score | | CM | | Cost [sec] |
		Training	Test	Training	Test	Training	Test	Training	Test	Training	Test	Test
SVC	1	[0.79, 0.68, 0.78]	0.75	0.72	0.86	0.64	0.63	0.65	0.62	[[611] [3 36]]	[[2 6] [0 16]]	0.231
	II	[0.89, 0.89, 0.94]	0.96	0.94	0.97	0.85	0.94	0.88	0.95	[[12 5] [0 39]]	[[70] [016]]	0.201
	III	[0.95, 0.90, 0.94]	0.96	0.97	0.97	0.94	0.94	0.95	0.95	[[13 4] [0 39]]	[[70] [016]]	0.227
KNN	1	[0.68, 0.68, 0.83]	0.71	0.68	0.85	0.64	0.56	0.65	0.52	[[710] [5 34]]	[[9] [016]]	0.245
	II	[0.95, 0.95, 0.94]	0.87	0.96	0.84	0.91	0.84	0.93	0.85	[[14 3] [0 39]]	[[6 2] [115]]	0.268
	III	[0.90, 0.90,1]	0.96	0.89	0.94	0.9	0.97	0.9	0.96	[[152] [3 36]]	[[8 0] [115]]	0.216
RF	1	[0.95, 0.90, 0.94]	0.96	0.95	0.97	0.88	0.94	0.91	0.95	[[13 4] [0 39]]	[[70] [016]]	1.208
	II	[0.95, 0.90, 0.94]	0.96	0.95	0.97	0.88	0.94	0.91	0.95	[[13 4] [0 39]]	[[70] [016]]	1.05
	III	[0.84, 1, 0.94]	0.96	0.94	0.97	0.85	0.94	0.88	0.95	[[12 5] [0 39]]	[[70] [016]]	1.062
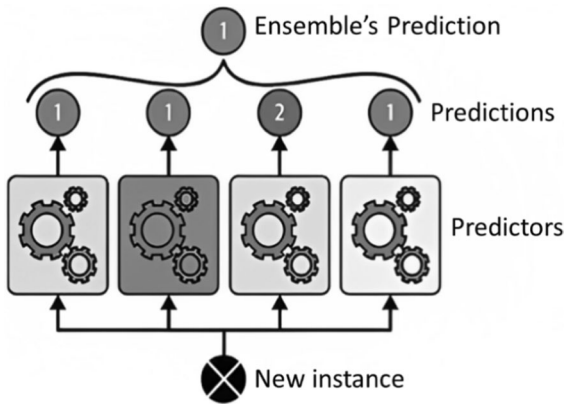
Fig. 11 Hard voting classifier predictions

Table 6 Accuracy of the ensemble learning algorithm and individual predictors' performances on different dataset.

Data	Ensemble	cm	KNN	SVC	RF
All	0.96	[7 1] [0 16]	0.88	0.96	0.96
A	0.93	[8 0] [2 18]	0.86	0.89	0.89
B	0.93	[7 1] [1 19]	0.86	0.89	0.89
C	0.93	[6 2] [0 20]	0.93	0.93	0.89
D	0.79	[6 2] [4 15]	0.71	0.75	0.75
H	0.93	[10 1] [1 16]	0.86	0.89	0.86

(All) whole data set; (A) all five samples that have A defect only removed from training set; (B) all five samples that have B defect only removed from training set; (C) all five samples that have C defect only removed from training set; (D) all five samples that have D defect only removed from training set; (H) five samples among the 25 healthy ones removed from training set

model and the ensemble model were evaluated, and the results are presented in Table 6. The ensemble algorithm outperforms the individual models in most cases.

Overall, the models performed well in individuating unseen defects, but a few issues were observed. Notably, the model performed poorly on data "D," which corresponds to a magnet flaw. This misprediction could be due to the defect's low impact on the pressure and acceleration data. The absence of bushings on either the front or rear body is being captured well, not producing any differences in the models' predictions. The results of this ablation study demonstrated that the algorithms were capable of maintaining a high overall accuracy, even when a particular defect was removed from the training dataset. Furthermore, Table 5 give emphasis on how each model perform singularly giving insight on their contribution to the ensemble model for each case.

This finding underscores the robustness of the algorithms in capturing the underlying physical phenomena, indicating their ability to generalize beyond specific training samples.

6 Extended results and analysis

This section utilizes the proposed methodology to address the issue of dimensionality reduction, which arises from the high number of features generated in the pre-processing phase (discussed in Sects. 4.1 and 4.2), and examines the impact of utilizing a limited number of sensors to render the procedure both affordable and scalable across diverse contexts.

6.1 Principal component analysis: PCA

A large number of features per training instance can make many ML problems slow to train and difficult to solve. This problem, also referred to as the "curse of dimensionality," presents significant difficulties in numerous applications. Dimensionality reduction techniques can be employed to compress high-dimensional data into a lower-dimensional space to solve this problem [87]. Additionally, training cases are frequently not distributed across all dimensions, particularly when multiple variables are strongly associated [88], and the high degree of correlation observed in Figs. 5 and 7 justify the employment of such tool. Furthermore, dimensionality reduction can be a useful tool to visualize high-dimensional data in a condensed form and gain insights into patterns.

Linear and nonlinear methods are the two main approaches to dimensionality reduction. Linear algorithms like principal component analysis (PCA), are simpler and faster but may be limited in their ability to capture complex patterns. Nonlinear methods, such as t-distributed Stochastic Neighbor Embedding (t-SNE) and Locally Linear Embedding (LLE), can capture more complex patterns, but are usually more computationally intensive and require more tuning.

PCA, very popular for dimensionality reduction [11], is an unsupervised ML algorithm that works by identifying the axes that account for the most variance in the training set and finding orthogonal axes that account for the remaining variance. In other words, PCA reduces the dataset to a smaller number of dimensions while retaining as much variation as it can by projecting the dataset onto the hyperplane defined by the first n principal components.

Figure 12 shows an example of using PCA to reduce and plot the training set down to two and three dimensions, respectively. The explained variance by reducing the dimensionality to two components is 42.5%, while it is 49.3% at three dimensions. The two classes of healthy and faulty pumps are clearly distinguishable despite some overlap.

Rather than arbitrarily imposing the number of dimensions that the dataset will be reduced to, finding the number of dimensions with the most variability is an alternate strategy. Figure 13 shows the explained variance plotted against the number of dimensions. As shown, it is possible to keep 95% of the variance by reducing the dimensionality to 23, which consists of 11% of the starting dataset. No dimensions above 55 have been plotted due to the constancy of the explained variance.

In Table 7, results in terms of accuracy are shown by comparing the standard case which consists of 202 features over the total number of dimensions that retain 95% of the variance, 23D and 2D and 3D. As visible, reducing the dimensionality of a dataset can result in a loss of information, similar to how compressing an image can degrade its quality. Nevertheless, dimensionality reduction is a powerful tool in ML and its use may be acceptable in specific scenarios where computational efficiency is a critical consideration. The choice to use dimensionality reduction should ultimately be based on a detailed analysis of the trade-off between computational efficiency and the potential degradation in performance.

6.2 Sensors' sensibility study

Ultimately, a sensitivity study was conducted to evaluate the effectiveness of sensors used to collect data. In an industrial context, minimizing the number of sensors is essential to reduce costs associated with hardware and computational effort required for real-time operations. This also makes it possible to implement such methodologies in fields like mobility machinery. In addition to the former argument, the ablation study presented in Table 8, by explicating single predictor's performance for each sensors configuration further demonstrate the overall better accuracy enhanced by the employment of the ensemble method over the single predictors.

Table 8 displays the accuracy and computational costs for various sensor combinations. The computational costs are defined as the ratio between the computational time the specific scenario and the computational time with all three sensors. The first column specifies the sensors used for each trial. The ensemble
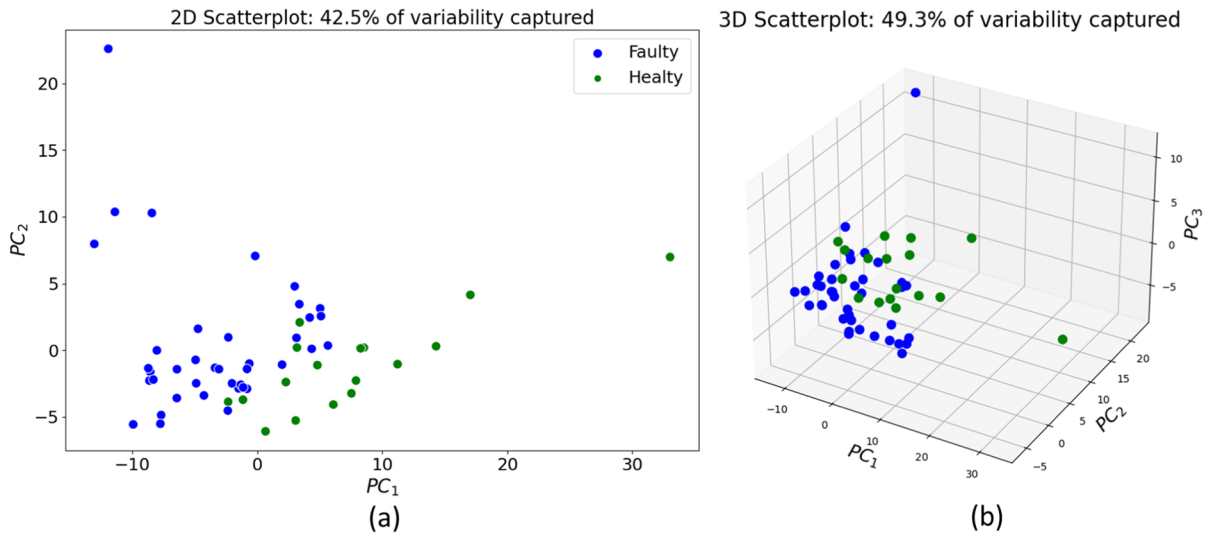
Fig. 12 Scatterplot of the transformed training dataset reduced up to two dimensions (**a**) and three dimensions (**b**) using PCA
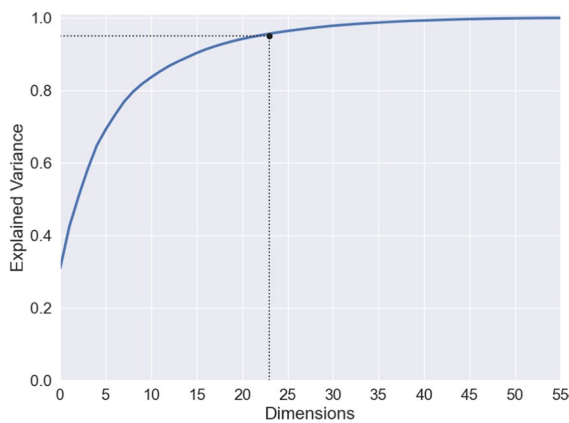
learning algorithm, composed of KNN, SVC, and RF, was trained and tested on the test set data after removing all features related to one or two sensors. Note that the hyperparameters for each model were kept constant at default values as proposed by scikit-learn [71] over the different trials.

The findings showed that just only using data extracted by the radial accelerometer preserves high performance while reducing computational time. Conversely, using data from the pressure sensor only produces the worst performance. This can be possibly explained due to the high noise carried by such signal resulting in inadequate data for the ML models to differentiate between the two categories. These

Fig. 13 Explained variance versus the number of dimensions

findings carry significant implications for upscaling these methodologies, especially in fields where spatial and computational limitations hold paramount importance, such as the automotive and aerospace industries. Additionally, constraining computational time enables more real-time operations, aligning with the prevailing trend in the mobility sector's evolution, characterized by an increasing number of connected devices and a growing demand for edge computing capabilities.

7 Conclusion

In summary, this research offers an innovative fault detection method for ePumps that can identify faulty

Table 7 Accuracy of the ensemble learning model and individual models for different dimensions using PCA

Dimensions	Ensemble	cm	KNN	SVC	RF
203	0.96	[7 1] [0 16]	0.88	0.96	0.96
23	0.92	[6 2] [0 16]	0.92	0.96	0.92
3	0.88	[6 2] [1 15]	0.88	0.92	0.79
2	0.83	[5 3] [1 15]	0.83	0.88	0.71

Table 8 Accuracy and cost comparison of the ensemble learning model and individual models using different sensor combinations

Data	Ensemble	cm	cost	KNN	SVC	RF	Dimensions
P,R,A	0.96	[7 1] [0 16]	100.0%	0.88	0.96	0.96	204
P,R	0.92	[6 2] [0 16]	6.8%	0.92	0.92	0.92	182
P,A	0.88	[5 3] [0 16]	5.7%	0.88	0.88	0.88	147
R,A	1.00	[8 0] [0 16]	8.3%	1.00	1.00	0.96	80
P	0.88	[5 3] [0 16]	18.5%	0.83	0.88	0.88	123
R	0.96	[7 1] [0 16]	16.2%	1.00	0.92	0.96	58
A	0.92	[6 2] [0 16]	25.7%	0.96	0.92	0.88	23

P = dynamic pressure sensor; R = radial accelerometer sensor; A = axial accelerometer sensor

components that would pass a traditional EOL test. The methodology utilizes a combination of an innovative knowledge-based vibroacoustic tool with prior knowledge of the kinematic chain and ML techniques to extract discriminatory features from time series data of pressure and acceleration signals. This work overcomes the major limitation of the purely data-driven approach discussed in the literature regarding the lack of causality analysis by combining the feature generation step with a vibroacoustic tool that utilizes prior knowledge of the kinematic chain, and secondly by examining the key features that the Random Forest algorithm used when solving the classification task. The results show how specific features such as "AOR", "TDV, and "AMR" are used by the ML classifiers to correctly identify the faulty ePumps with respect to the healthy ones. Parallelly, features extracted using a data-driven approach have been proven explanatory for this specific problem, by empowering features well-known in the literature concerning the field of condition monitoring related to rotary machines. Results from the experimental test campaign show that the proposed model, which includes several ML classifiers in an ensemble learning algorithm, accurately detects faulty units with a precision higher than 95% on 80 experimental samples.

Moreover, a comprehensive study aimed at dimensionality reduction has been executed using the PCA method. The outcomes demonstrate the potentiality of this methodology, which allows the preservation of 95% of the explained variance by reducing the dataset up to 89% from the initial one with limited degradation of predictions' accuracy.

In the end, a sensor sensitivity analysis has been conducted by considering different sensors' combinations in an attempt to decrease the number of sensors to the minimum to make this procedure affordable and scalable even in fields with limited resources due to cost or space constraints. According to the findings of the study, it was determined that an accuracy of 96% can be achieved by using only the radial accelerometer, resulting in a substantial reduction in computational time by 83.8%.

While the proposed approach demonstrates considerable success in the context of ePumps, it is essential to delineate the scope of its applicability. The methodology's effectiveness may vary depending on the specific characteristics of electric pumps, and further research is needed to ascertain its adaptability to different pump types and potentially extend its utility to other domains. Notably, while the methodology conceptually scales to other hydraulic machines, factors such as the positioning of acceleration and pressure sensors, sampling frequencies, and the specific features extracted from raw data are inherently specific to each machine and application, and require specific consideration.

In conclusion, the implications of the proposed methodology are particularly significant for the pump manufacturing industry, offering the prospect of enhanced product quality without an associated increase in costs. Furthermore, this research holds promise for broader applications across diverse

industries and machine types, presenting a comprehensive framework for fault detection that seamlessly integrates machine learning and causal analysis. Subsequent research endeavors can delve into the application of the proposed methodology with larger sample sizes and the exploration of new fault types.

Overall, this paper presents a significant contribution to the field of fault detection in ePumps by providing a novel methodology that achieves high precision and interpretability. By combining DOE, ML techniques, and causal analysis, this research can lead to more efficient, reliable, and safe systems in the pump manufacturing industry and beyond.

Data availability The datasets generated during the current study are not publicly available due to confidentiality for the pump manufacturer but are available from the corresponding author on reasonable request.

Declarations

Appendix

Dummy classifier

A classifier model that produces predictions without looking for patterns in the data is known as a "dummy classifier." The training data's most prevalent class is essentially what the default model predicts. This classifier is useful for establishing a benchmark for the performance of more advanced classifiers, since in general, any other should perform better on the provided dataset.

Stochastic gradient descent: SDG

Stochastic Gradient Descent is an optimization methodology used in ML to optimize the model's weights. At each step, a random instance from the training set is chosen, and the gradients are computed using only that one instance. SGD is a popular option for large-scale ML issues because of how much faster it is than other optimization strategies thanks to this method.

However, this approach is considerably less regular than batch gradient descent because of its stochastic (i.e., random) nature. Once there, it will continue to bounce around, never settling down, until it eventually ends up very close to the minimum (see Fig. 14). The final parameter values once the process terminates will be good but not ideal.

The effectiveness and simplicity of SGD's implementation are among its benefits. The regularization value and iterations' number are just two of the hyperparameters that are needed. SGD is additionally sensitive to feature scaling, which can have an impact on the speed of convergence and the caliber of the model that is produced.

Support vector machine: SVM

SVM is a strong and adaptable ML algorithm that can perform regression, novelty detection, and linear or nonlinear classification. Although SVMs scale poorly to very large datasets, they operate well with small to medium-sized nonlinear datasets. By employing the support vectors to generate a separation hyperplane, SVMs establish a boundary between data classes. The gamma and regularization parameters can be adjusted, and various kernels can be used for the decision function.

The ideal decision boundary for dividing data types A (squares) and B (circles) so that the two classes may be differentiated clearly without overlapping is shown in Fig. 15. The regularization parameter, which can be adjusted depending on experience, determines the separation between the decision boundaries. The benefits of SVMs include their efficacy in high-dimensional domains, memory efficiency because only a portion of the training points

are used in the decision function, and flexibility in the specification of various kernel functions. Overfitting when there are many more features than samples, as well as not directly supplying probability estimates, are drawbacks.

K-nearest neighbor: *KNN*

The K-Nearest Neighbor technique is a straightforward but flexible supervised ML approach that can be applied to both classification and regression tasks [57]. It is predicated on the idea that data points with comparable features also have similar feature values. The algorithm uses the k closest training examples in the feature space as input. KNN computes the distance between a new data point and each training sample using metrics such the Euclidean, Hamming, or Manhattan distance in order to categorize the new data point. The method then chooses the k database entries that are most similar to the new instances and uses a majority vote to establish the new data point's class.

It is easy to use the example in Fig. 16 to demonstrate this. The fact that there are more triangles than squares inside the circle indicates that a new data point represented by the star can be categorized as a triangle. However, the categorization may be affected by altering the value of the k parameter,

which stands for the distance in Euclidean space. For instance, the star would be categorized as a square if the circle were to extend enough to contain three squares and two triangles. It is crucial to normalize the data by converting it to a specific value between 0 and 1, using the average value, or using the standard deviation in order to assure proper classification.

Decision tree: DT

A common ML method for classification and regression functions is the decision tree. Starting from a set of training data, the algorithm constructs a model resembling a tree of decisions and potential outcomes. A decision or feature is represented by each node in the tree, and each branch denotes a potential result or value. The leaf nodes provide the final classification or regression value, whereas the root node represents the most important feature.
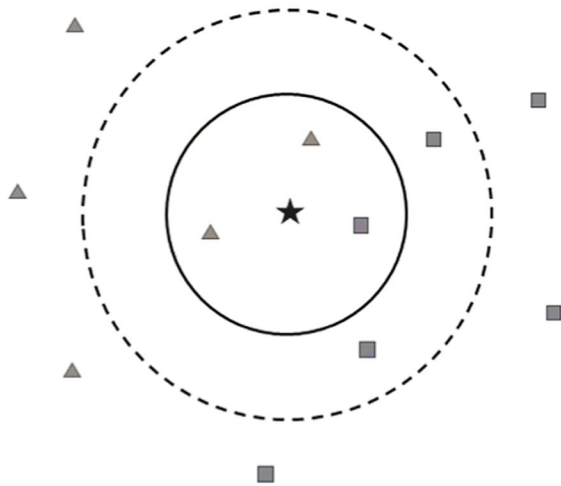
Starting with a training dataset, the DT algorithm chooses the optimum feature to divide the data depending on a certain measure (such as information gain or Gini impurity). Each subset of the data is subjected to this procedure repeatedly until a stopping
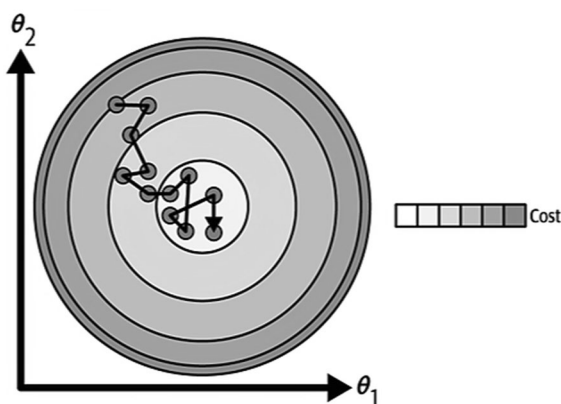
Fig. 14 Stochastic gradient descendant techniques
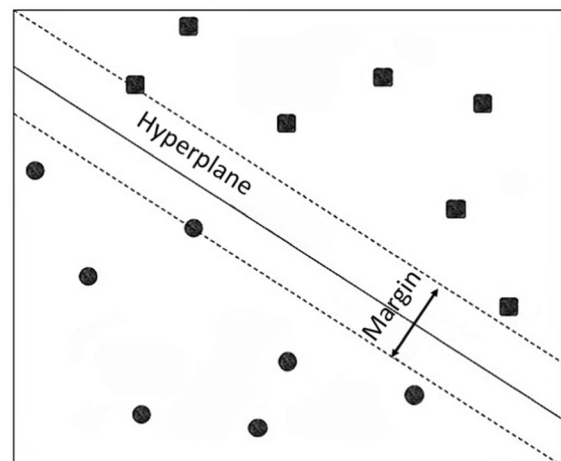
Fig. 15 Illustration of a typical SVM architecture

Fig. 16 Illustration of a typical KNN architecture

requirement, such as a maximum depth or a minimum number of samples per leaf, is met.

Since the resulting tree may be easily comprehended and described by non-experts, one benefit of DTs is that they are simple to visualize and interpret. Decision trees can also handle missing values, numerical data, and both category and categorical data. But when the tree gets too deep or complicated, decision trees are vulnerable to overfitting. Several methods, like pruning, restricting the tree depth, or using ensemble methods like RFs, can be utilized to address this problem.

Random forest: RF

RFs are a well-liked ensemble learning technique that combines several DTs to increase prediction stability

and accuracy [58]. A random forest, as opposed to a single DT, is trained using random batch of the input features and training data. By lowering the chance of overfitting and strengthening the model's robustness, this method makes it suitable for high-dimensional and noisy data. According to Fig. 17, the random forest algorithm operates in four steps:

From the dataset, random samples are chosen, decision trees are built for each sample, predictions are obtained from each tree, predictions are voted on, and the prediction with the most votes is chosen as the final prediction. By combining the predictions of multiple DTs through majority voting or weighted averaging, RF improves the model's accuracy. RF has been widely used in various domains, including bioinformatics, finance, and image recognition often outperforming other popular ML algorithms, such as DT and SVM, concerning accuracy and computational efficiency [58]. However, the performance of RF depends on several factors, such as the number of trees in the forest, the size of the random subsets, and the quality of the input features. Therefore, careful tuning of these parameters is necessary to achieve optimal performance.

Overall, RF is a powerful and flexible ML method that can handle a variety of tasks, including classification, regression, feature selection, and anomaly detection. They are particularly well-suited for large and complex datasets with high-dimensional input spaces and noisy or missing data. However, like other ensemble methods, RF can be considered a black-box type classifier, making it challenging to interpret the model and explain its predictions.

Fig. 17 Framework of RF

References

1. Hughes A, Drury B (2019) Electric motors and drives: fundamentals, types and applications. Newnes

2. International Energy Agency (2021) Global EV Outlook 2021: scaling up the transition to electric mobility

3. Yu J, Zhang Y (2023) Challenges and opportunities of deep learning-based process fault detection and diagnosis: a review. Neural Comput Appl 35(1):211–252

4. Lee CKM, Lv Y, Hong Z (2013) Risk modeling and assessment for distributed manufacturing system. Int J Prod Res 51(9):2652–2666

5. Pichler K, Haas R, Putz V, Kastl C (2021) Degradation detection for internal gear pumps using pressure reduction time maps. In: Annual conference of the PHM society, Vol 13, No 1

6. Rituraj F, Vacca A, Morselli MA (2019) Modeling of manufacturing errors in external gear machines and experimental validation. Mech Mach Theory 140:457–478

7. Vasiliev I, Frangu L, Cristea ML (2022) Pump fault detection using autoencoding neural network. In: 2022 26th international conference on system theory, control and computing (ICSTCC) (pp 426–431). IEEE

8. Siano D, Panza MA (2018) Diagnostic method by using vibration analysis for pump fault detection. Energy Procedia 148:10–17

9. Ma J, Li JC (1995) Detection of localized defects in rolling element bearings via composite hypothesis test. Mech Syst Signal Process 9(1):63–75

10. "Pump tests," Intro to Pumps. https://www.introtopumps.com/pump-terms/pump-tests

11. Lee GH, Akpudo UE, Hur JW (2021) FMECA and MFCC-based early wear detection in gear pumps in cost-aware monitoring systems. Electronics 10(23):2939

12. Jardine AK, Lin D, Banjevic D (2006) A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mech Syst Signal Process 20(7):1483–1510

13. Gertler J (1998) Fault detection and diagnosis in engineering systems. CRC Press

14. Simani S, Fantuzzi C, Patton RJ (2003) Model-based fault diagnosis techniques. Springer, London, pp 19–60

15. Zhang Y, Vacca A, Gong G, Yang H (2023) Quantitative fault diagnostics of hydraulic cylinder using particle filter. Machines 11:1019. https://doi.org/10.3390/machines11111019

16. Hu Q, Si XS, Zhang QH, Qin AS (2020) A rotating machinery fault diagnosis method based on multi-scale dimensionless indicators and random forests. Mech Syst Signal Process 139:106609

17. Van Tung T, Yang BS (2009) Machine fault diagnosis and prognosis: The state of the art. Int J Fluid Mach Syst 2(1):61–71

18. Wuest T, Weimer D, Irgens C, Thoben KD (2016) Machine learning in manufacturing: advantages, challenges, and applications. Prod Manuf Res 4(1):23–45

19. Lakshmanan K (2021) Predictive maintenance of an external gear pump using machine learning algorithms (Doctoral dissertation, Swansea University)

20. Xu W, Zhou Z, Li T, Sun C, Chen X, Yan R (2022) Physics-constraint variational neural network for wear state assessment of external gear pump. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2022.3213009

21. Soares SG, Araújo R (2015) An on-line weighted ensemble of regressor models to handle concept drifts. Eng Appl Artif Intell 37:392–406

22. Kumar A, Kumar R (2017) Time-frequency analysis and support vector machine in automatic detection of defect from vibration signal of centrifugal pump. Measurement 108:119–133

23. Samanta B (2004) Gear fault detection using artificial neural networks and support vector machines with genetic algorithms. Mech Syst Signal Process 18(3):625–644

24. Farokhzad S (2013) Vibration based fault detection of centrifugal pump by fast Fourier transform and adaptive neuro-fuzzy inference system. J Mech Eng Technol 1(3):82–87

25. Farokhzad S, Ahmadi H, Jafary A (2013) Fault classification of centrifugal water pump based on decision tree and regression model. J Sci Today's World 2(2):170–176

26. Barszcz T, Randall RB (2009) Application of spectral kurtosis for detection of a tooth crack in the planetary gear of a wind turbine. Mech Syst Signal Process 23(4):1352–1365

27. Wang Y, Xiang J, Markert R, Liang M (2016) Spectral kurtosis for fault detection, diagnosis and prognostics of rotating machines: A review with applications. Mech Syst Signal Process 66:679–698

28. Mollazade K, Ahmadi H, Omid M, Alimardani R (2008) An intelligent combined method based on power spectral density, decision trees and fuzzy logic for hydraulic pumps fault diagnosis. Int J Intell Syst Technol 3(4):251–263

29. Peng ZK, Chu FL (2004) Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography. Mech Syst Signal Process 18(2):199–221

30. ALTobi MAS, Bevan G, Wallace P, Harrison D, Ramachandran KP (2019) Fault diagnosis of a centrifugal pump using MLP-GABP and SVM with CWT. Eng Sci Technol Int J 22(3):854–861

31. Bagheri B, Ahmadi H, Labbafi R (2011) Implementing discrete wavelet transform and artificial neural networks

for acoustic condition monitoring of gearbox. Elixir Mech Eng 35:2909–2911

32. Yang Y, He Y, Cheng J, Yu D (2009) A gear fault diagnosis using Hilbert spectrum based on MODWPT and a comparison with EMD approach. Measurement 42(4):542–551

33. Yu X, Ren X, Wan H, Wu S, Ding E (2019) Rolling bearing fault feature extraction and diagnosis method based on MODWPT and DBN. In: 2019 11th international conference on wireless communications and signal processing (WCSP) (pp 1–7). IEEE

34. Shalev-Shwartz S, Ben-David S (2014) Understanding machine learning: From theory to algorithms. Cambridge University Press

35. John GH, Langley P (2013) Estimating continuous distributions in Bayesian classifiers. arXiv preprint arXiv:1302.4964

36. Wang W, Yang Y, Zhou Z (2015) Naive Bayes for big data classification: an empirical study. Expert Syst Appl 42(11):4970–4980

37. Friedman N, Geiger D, Goldszmidt M (1997) Bayesian network classifiers. Mach Learn 29:131–163

38. Cover T, Hart P (1967) Nearest neighbor pattern classification. IEEE Trans Inform Theory 13(1):21–27

39. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20:273–297

40. Chang CC, Lin CJ (2011) LIBSVM: a library for support vector machines. ACM Trans Intell Syst Technol (TIST) 2(3):1–27

41. Vapnik VN (1999) An overview of statistical learning theory. IEEE Trans Neural Netw 10(5):988–999

42. Breiman L (2001) Random forests. Mach Learn 45:5–32

43. Rumelhart DE, Hinton GE, Williams RJ (1986) Learning representations by back-propagating errors. Nature 323(6088):533–536

44. Graves A, Schmidhuber J (2005) Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Netw 18(5–6):602–610

45. Gong W, Chen H, Zhang Z, Zhang M, Wang R, Guan C, Wang Q (2019) A novel deep learning method for intelligent fault diagnosis of rotating machinery based on improved CNN-SVM and multichannel data fusion. Sensors 19(7):1693

46. AlShorman O, Irfan M, Saad N, Zhen D, Haider N, Glowacz A, AlShorman A (2020) A review of artificial intelligence methods for condition monitoring and fault diagnosis of rolling element bearings for induction motor. Shock Vib 2020:1–20

47. Bin GF, Gao JJ, Li XJ, Dhillon BS (2012) Early fault diagnosis of rotating machinery based on wavelet packets—empirical mode decomposition feature extraction and neural network. Mech Syst Signal Process 27:696–711

48. Montavon G, Samek W, Müller KR (2018) Methods for interpreting and understanding deep neural networks. Digit Signal Process 73:1–15

49. Liu R, Yang B, Zio E, Chen X (2018) Artificial intelligence for fault diagnosis of rotating machinery: a review. Mech Syst Signal Process 108:33–47

50. Zhou ZH (2012) Ensemble methods: foundations and algorithms. CRC Press

51. Ivantysyn J, Ivantysynova M (2003) Hydrostatic pumps and motors: principles, design, performance, modelling, analysis, control and testing

52. Géron, A. (2022).Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow. "O'Reilly Media, Inc."

53. https://www.fluidotech.it/en/products/technologies/external-gear-pumps/

54. Aliyu R, Mokhtar AA, Hussin H (2022) Prognostic health management of pumps using artificial intelligence in the oil and gas sector: a review. Appl Sci 12(22):11691

55. Zhao X, Vacca A (2019) Theoretical investigation into the ripple source of external gear pumps. Energies 12(3):535

56. Osman AH, Gobran MH, Mahmoud FF (2019) Vibration signature of normal and notched tooth gear pump. Eur Sci J 15:64–75

57. Benhabib B (2003) Manufacturing: design, production, automation, and integration. CRC Press

58. Kudelina K, Asad B, Vaimann T, Rassolkin A, Kallaste A, Lukichev DV (2020) Main faults and diagnostic possibilities of BLDC motors. In: 2020 27th international workshop on electric drives: MPEI department of electric drives 90th anniversary (IWED) (pp. 1–6). IEEE

59. Montgomery DC (2017) Design and analysis of experiments. Wiley

60. Khan A, Hwang H, Kim HS (2021) Synthetic data augmentation and deep learning for the fault diagnosis of rotating machines. Mathematics 9(18):2336

61. Li X, Zhang W, Ding Q, Sun JQ (2020) Intelligent rotating machinery fault diagnosis based on deep learning using data augmentation. J Intell Manuf 31:433–452

62. Marinaro G, Frosina E, Senatore A, Stelson KA (2021) A fast and effective method for the optimization of the valve plate of swashplate axial piston pumps. J Fluids Eng 143(9):091203

63. Christ M, Braun N, Neuffer J, Kempa-Liehr AW (2018) Time series feature extraction on basis of scalable hypothesis tests (tsfresh–a python package). Neurocomputing 307:72–77

64. Christ M, Kempa-Liehr AW, Feindt M (2016) Distributed and parallel time series feature extraction for industrial big data applications. arXiv preprint arXiv:1610.07717

65. Kempa-Liehr AW, Oram J, Wong A, Finch M, Besier T (2020) Feature engineering workflow for activity recognition from synchronized inertial measurement units. In: Pattern recognition: ACPR 2019 workshops, Auckland, New Zealand, November 26, 2019, Proceedings 5 (pp 223–231). Springer Singapore

66. Kennedy A, Nash G, Rattenbury NJ, Kempa-Liehr AW (2021) Modelling the projected separation of microlensing events using systematic time-series feature engineering. Astron Comput 35:100460

67. Teh HY, Kevin I, Wang K, Kempa-Liehr AW (2021) Expect the unexpected: unsupervised feature selection for automated sensor anomaly detection. IEEE Sens J 21(16):18033–18046

68. Boslaugh S (2012) Statistics in a nutshell: a desktop quick reference. "O'Reilly Media, Inc."

69. https://www.peivm.it/wp-content/uploads/2023/01/Brochure_PEI-VM_01-2023_EN_web.pdf

70. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Duchesnay E (2011) Scikit-learn: Machine learning in Python. J Mach Learn Res 12:2825–2830

71. Agrawal T, Agrawal T (2021) Hyperparameter optimization using scikit-learn. Hyperparameter optimization in machine learning: make your machine learning and deep learning models more efficient, pp 31–51

72. Maxwell AE, Warner TA, Fang F (2018) Implementation of machine-learning classification in remote sensing: an applied review. Int J Remote Sens 39(9):2784–2817

73. Belgiu M, Drăguţ L (2016) Random forest in remote sensing: a review of applications and future directions. ISPRS J Photogramm Remote Sens 114:24–31

74. Guan H, Li J, Chapman M, Deng F, Ji Z, Yang X (2013) Integration of orthoimagery and lidar data for object-based urban thematic mapping using random forests. Int J Remote Sens 34(14):5166–5186

75. Gislason PO, Benediktsson JA, Sveinsson JR (2006) Random forests for land cover classification. Pattern Recogn Lett 27(4):294–300

76. Ghosh A, Fassnacht FE, Joshi PK, Koch B (2014) A framework for mapping tree species combining hyperspectral and LiDAR data: Role of selected classifiers and sensor across three spatial scales. Int J Appl Earth Obs Geoinf 26:49–63

77. Yan R, Gao RX (2007) Approximate entropy as a diagnostic tool for machine health monitoring. Mech Syst Signal Process 21(2):824–839

78. Ma J, Li Z, Li C, Zhan L, Zhang GZ (2021) Rolling bearing fault diagnosis based on refined composite multi-scale approximate entropy and optimized probabilistic neural network. Entropy 23(2):259

79. Liu L, Zhi Z, Zhang H, Guo Q, Peng Y, Liu D (2019) Related entropy theories application in condition monitoring of rotating machineries. Entropy 21(11):1061

80. Li H, Huang J, Yang X, Luo J, Zhang L, Pang Y (2020) Fault diagnosis for rotating machinery using multiscale permutation entropy and convolutional neural networks. Entropy 22(8):851

81. Yoon J, He D, Van Hecke B, Nostrand TJ, Zhu J, Bechhoefer E (2016) Vibration-based wind turbine planetary gearbox fault diagnosis using spectral averaging. Wind Energy 19(9):1733–1747

82. Jin Z, Han Q, Zhang K, Zhang Y (2020) An intelligent fault diagnosis method of rolling bearings based on Welch power spectrum transformation with radial basis function neural network. J Vib Control 26(9–10):629–642

83. Zerdani S, Elhafyani ML, Zouggar S (2022) Application of power spectral density and the support vector machine to fault diagnosis for permanent magnet synchronous motor. SN Appl Sci 4:245

84. Rafiee J, Tse PW (2009) Use of autocorrelation of wavelet coefficients for fault diagnosis. Mech Syst Signal Process 23(5):1554–1572

85. Dou D, Zhou S (2016) Comparison of four direct classification methods for intelligent fault diagnosis of rotating machinery. Appl Soft Comput 46:459–468

86. Li Q, Ji X, Liang SY (2017) Incipient fault feature extraction for rotating machinery based on improved AR-minimum entropy deconvolution combined with variational mode decomposition approach. Entropy 19(7):317

87. Bishop CM, Nasrabadi NM (2006) Pattern recognition and machine learning. Springer, New York, p 738

88. Hastie T, Tibshirani R, Friedman JH, Friedman JH (2009) The elements of statistical learning: data mining, inference, and prediction. Springer, New York, pp 1–758