Hindawi
Shock and Vibration
Volume 2023, Article ID 1091276, 20 pages
https://doi.org/10.1155/2023/1091276

Research Article

Prediction of Remaining Useful Life of Packing Sets in a Plunger-Type High-Pressure Compressor Based on the PCA/SVD Analysis and NN Model

Jin-Wei Liang,[1] Shyh-Chin Huang,[1] and Chun-Ling Lin[2]

[1]*Department of Mechanical Engineering, Ming Chi University of Technology, New Taipei City, Taiwan*
[2]*Department of Electrical Engineering, Ming Chi University of Technology, New Taipei City, Taiwan*

Correspondence should be addressed to Chun-Ling Lin; ginnylin@mail.mcut.edu.tw

Received 9 November 2022; Revised 26 December 2022; Accepted 31 December 2022; Published 13 January 2023

Academic Editor: Raffaella Di Sante

Copyright © 2023 Jin-Wei Liang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A machine learning-based prognostic strategy is developed in this paper for predicting the remaining useful life (RUL) of high-pressure packing in plunger-type hypercompressors. The proposed strategy applies principal component analysis (PCA) to identify the three most important sensors out of 33 possible options which seem relevant to the subject of high-pressure packing. Singular-value decomposition (SVD) is then performed with respect to chronological Hankel matrices reconstructed from one of these three pieces of sensor data, namely, leakage flow. The normalised correlation coefficient between SVD eigenvalue vectors of chronological data is defined with a view to formulate health state assessment measurement. In order to enhance the prediction accuracy of the RUL of high-pressure packing, a linear regression algorithm and a two-term power series regression algorithm are both integrated into the NN (neural network) model. The effectiveness of the method is examined using the averaged difference (over 13 data sets) between predicted and real failure events. The results showed that the maximum prediction RUL error of the model is less than 15 days, and an averaged prediction RUL error is 7.23 days for 13 run-to-failure events. Additionally, a more recent test was performed using online data to examine the health states of four identical types of packing.

1. Introduction

Machine condition monitoring is important in manufacturing factories, especially those involved in chemical reaction processes. To prevent important equipment from failing unexpectedly, an increasing amount of attention has been paid to engineering prognostics in practice. Although traditional equipment prognostics is based on the experience of personnel who are familiar with equipment, its feasibility has been diminishing due to improved asset reliability and equipment complexity. On the other hand, the implementation of machine monitoring takes the analysis of the relevant information acquired from various sensors and uses it to determine the health condition of the system, or, in some circumstances, one of the important components. Benefitting from the progress of computational ability, prognostics, as an advanced maintenance technique, has been a popular topic of research. Nevertheless, reliable prognostics remains as the state-of-the-art technology in most cases due to many difficulties involved in attacking real problems. One of the primary difficulties in implementing effective machine health prognostics lies in the fact that the nature of defect growth is highly stochastic. Therefore, it is challenging to come up with an effective health indicator for the online quantification of machine health degradation. Recently, certain methods have been recommended for machine health assessment and prediction to prevent unexpected machine downtime [1–3]. Jianbo suggested that the hybrid feature selection strategy is capable of choosing the representation feature for machine health assessment without human intervention [4]. Moreover, Atamuradov et al. constructed a machine health predictor framework for failure diagnostics and prognostics [5].

The predictor approaches are generally categorised into three types, including model-based, data-driven, and hybrid methods. Since most engineering systems are complicated, it is difficult to establish an accurate model of the system or the component degradation process. In contrast, data-driven approaches model the degradation characteristics of the system based on historical run-to-failure sensor data. These approaches can infer correlations and causalities hidden in data while learning underlying trends. Most current sensor-based, data-driven methods for RUL prediction are rooted in statistics and fall into two categories, namely, stochastic process techniques [6, 7] and artificial intelligence tools such as neural networks [8], recurrent neural networks [9], and long short-term memory (LSTM) networks [10]. The former rely on statistical models to determine the RUL in a probabilistic way, whereas the latter rely on machine learning tools and are nonprobabilistic in nature.

In the current study, a data-driven and stochastic-based prognostic strategy was developed to predict the RUL of milling machine cutting tools [6], which involved the application of the autoregressive-integrated moving average (ARIMA) method. The RUL prediction indicated that 25% (approximately) extra tool usage can be achieved. On the other hand, Li et al. [7] developed a systematic methodology focusing on ball screw failure. The approach consisted of fault diagnosis, early diagnosis, health assessment, and RUL prediction. Gaussian process regression was adopted to predict the trend of degradation behaviour, while principal component analysis (PCA) was utilised to determine the optimal feature sets of ball screw failures in [5]. The results indicated that the built-in sensor data were valuable in implementing fault diagnosis, whereas additional sensor data seemed to be needed to address the RUL prediction problems given the complex behaviour involved in ball screw failures. Zhang et al. proposed a multiobjective deep belief networks ensemble (MODBNE) method [8]. The method employed a multiobjective evolutionary algorithm integrated into traditional DBN training techniques. The resultant DBNs were combined in order to formulate an ensemble model used for RUL estimation. The combination weights in the ensemble model were optimised via a single-objective differential evolution algorithm using a task-oriented objective function. Yu et al. [9] proposed a sensor-based data-driven algorithm combined with a deep learning tool and the similarity oriented matching technique to estimate the RUL of a system. The approach can be divided into two steps. The first step applied a bidirectional recurrent neural network-based autoencoder to convert multisensor readings collected from historical run-to-failure instances to low-dimensional embedding; in contrast, the latter were used to construct the one-dimensional health index (HI) values to reflect different health degradation patterns of run-to-failure instances. In the second step, the online HI curve obtained from sensor readings in real-time data was compared with the degradation patterns built in the offline phase. The similarity-based curve matching technique was adopted in this stage, from which the real-time RUL of the test unit can be obtained. Galli et al. [10] proposed an LSTM-based model combining self-monitoring analysis and reporting technology (SMART) attributes and temporal analysis for estimating health status of a hard disk drive (HDD) according to its time to failure. The methodology was grounded in three main steps: health degree definition, sequence extraction, and health status assessment through LSTM. Indeed, LSTM networks were interesting in the context of HDD failure prediction, as they took advantage of the highly sequential nature of the information available to the model. The experimental results showed that the proposed method can predict hard drive health status up to 45 days before failure. Ding et al. [11] addressed the in-depth study of autoregression-based prognostics and proposed the stationary subspaces-vector autoregressive with exogenous terms (SSVARX) methodology to estimate the degradation trend estimation (DTE) of rotating machinery. The authors selected the multichannel vibration signals and converted nonstationary signals into time and frequency domain-based weak-stationary degradation indicators. Subsequently, they adopted the proposed DTE models to evaluate domain variables after the stationarity test, order determination, and impulse response analysis. Following two run-to-failure life tests of rolling and slewing bearings, it was clear that the SSVARX produced highly accurate prediction results. The data recorded from real industrial scenes may not be sufficient and could lead to negative impacts, such as overfitting problems. Peng proposed [12] the unsupervised meta gated recurrent unit (UMGRU) containing a dual-cycle learning architecture with the designed clustering assignment module to estimate the few-shot prognostics under unlabelled historical data. The UMGRU provided reliable RUL predictions for both experimental and industrial scenarios. Roughly speaking, in numerous data-driven RUL prediction studies, either statistical approaches [6, 7, 13–16] or artificial intelligence methods [8–12, 17–19] were applied.

Although various prognostic approaches have been proposed for different applications, few existing schemes provide promising performance across these applications. The current study combined PCA-based order reduction, singular-value decomposition (SVD)-based health state assessment, neural network (NN)-based prediction models, and regression algorithms to predict the RUL of high-pressure packing in hypercompressors. Due to the importance of the equipment, more than 190 sensors, including those for pressure, temperature, vibration, and leakage flow, were used to monitor the operation conditions of the plunger-type high-pressure compressor. To simplify the problem, domain expertise knowledge was first applied to reduce the number of relevant sensors to 33. Subsequently, PCA, a common method used by researchers for order reduction [20–23], was adopted to further reduce the number of relevant sensors to three. The analysis conducted using PCA indicated that the leakage flow rate through the high-pressure packing set and the X- and Y-directional vibrations measured on the plunger were the three most critical pieces of data related to the failure events observed in high-pressure packing. Following this, raw signals obtained from the leakage flow were used to conduct phase-space reconstruction and obtain the so-called Hankel matrix [17, 18]. SVD-based analysis was conducted with respect to the chronological Hankel matrices, from which the SVD-normalised correlation coefficients were determined [15]. A total of 37 features, including 11 associated with each of the three relevant sensors and four other features obtained from the actual health degradation curve, were chosen as inputs to a feedforward neural network scheme to develop a packing health state assessment model. Features associated with leakage flow and vibration data included mean, skewness, standard deviation, and RMS. The health degradation curve of packing was defined by using the SVD-normalised correlation coefficients obtained from the leakage data. The resultant neural network model consisted of two hidden layers, with ten and five hidden nodes, while the linear and two-term power series algorithms were both applied to regress degradation trends. The model was verified by using 13 run-to-failure data sets to illustrate its effectiveness in predicting the RUL of high-pressure packing.

The motivation of this study lies mainly in the fact that the plunger-type high-pressure compressor is an extremely crucial machine in the ethylene-vinyl acetate (EVA) production line such that any unscheduled downtime would result in significant financial loss [24]. To avoid this, an overly conservative maintenance strategy has long been applied. Therefore, an accurate RUL algorithm applicable to this compressor would be cost-effective and beneficial to the petrochemical industry.

Regarding the difference between our derivation and most data-driven approaches, the derived packing RUL algorithm in the current paper is a combination of data-driven knowledge and domain knowledge, and this appears to be necessary and crucial. For instance, because packing is composed of six sealing rings, low readings of the leakage sensor by no means imply healthy status, since some rings may change their orientations during operation and seal the leakage temporarily. Therefore, knowledge-based techniques for failure criterion setup, feature selection, etc., are very helpful. To this end, the algorithm has been tested online and proven to be effective on the real compressor.

2. Methodology

This study combines PCA, SVD, NN, and regression algorithms to develop a health prognostic method for predicting the RUL of high-pressure packing in a plunger-type hypercompressor. The flowchart of the proposed method is shown in Figure 1. Details regarding the different algorithms involved in Figure 1 are elaborated in the following subsections.

2.1. Principal Component Analysis (PCA). The most popular technique for dimensionality reduction in machines is PCA [20–23]. PCA is an unsupervised linear transformation method, the aim of which is to find the directions of maximum variance in high-dimensional data and project that variance into a new subspace with fewer dimensions. The minimum set of the largest principal components (PCs), which accounts for at least some predefined variance threshold (usually in the range of 80%–95% of original data variance), is considered for further analyses [25]. Theoretically, PCA can identify (from a black box containing many uncorrelated measured variables) the related sensors that have a strong correlation with normal and abnormal operation conditions in a complicated system such as the high-pressure packing facility.

Details regarding the processing of PCA and the procedures involved are elaborated as follows:

(a) Normalisation of raw data: Suppose we have raw data matrix $\mathbf{X}$ comprising a set of n observations of p variables. A standard deviation normalisation process can be applied to the raw data matrix to acquire resultant data matrix $\mathbf{B}$, which possesses unit variance and zero mean [26].

(b) We obtain $p \times p$ empirical covariance matrix $\mathbf{C}$ from normalised matrix $\mathbf{B}$, which is achieved in step A as follows:

$$\mathbf{C} = \frac{1}{n-1}\mathbf{B}^T\mathbf{B}, \tag{1}$$

where n is the number of observations.

(c) We find the eigenvectors and eigenvalues of the covariance matrix. Matrix $\mathbf{V}$ of the eigenvectors can be used to diagonalise covariance matrix $\mathbf{C}$ as follows:

$$\mathbf{V}^{-1}\mathbf{C}\mathbf{V} = \mathbf{D}, \tag{2}$$

where $\boldsymbol{D}$ is the diagonal matrix of the eigenvalues of $\mathbf{C}$. Matrix $\mathbf{D}$ takes the form of a $p \times p$ diagonal matrix as follows:

$$D_{kl} = \begin{cases} \lambda_k, & k = l, \\ 0, & k \neq l, \end{cases} \tag{3}$$

where λ_k is the jth eigenvalue of covariance matrix $\mathbf{C}$. Matrix $\mathbf{V}$, also of dimension $p \times p$, contains p column vectors. The aforementioned vectors are the eigenvectors of covariance matrix $\mathbf{C}$.

2.2. Singular Value Decomposition (SVD)-Normalised Correlation Coefficient. The SVD-normalised correlation coefficient is the method used to automatically assess the health state [18]. The theory behind the SVD-normalised correlation coefficient is that the correlation between the singular-value vector and the normal state data is higher than the correlation which exists between the normal state and fault state data. The procedures adopted to calculate the SVD-normalised correlation coefficient, R_j, for health state assessment are as follows:

(a) Normalisation of the selected imported raw data: The selected imported raw data are normalised by min-max normalisation within the range of [−1, 1].

(b) Calculate the singular-value matrix of each subsequence by using SVD: In the current study, the window size of each subsequence is 1024 points,

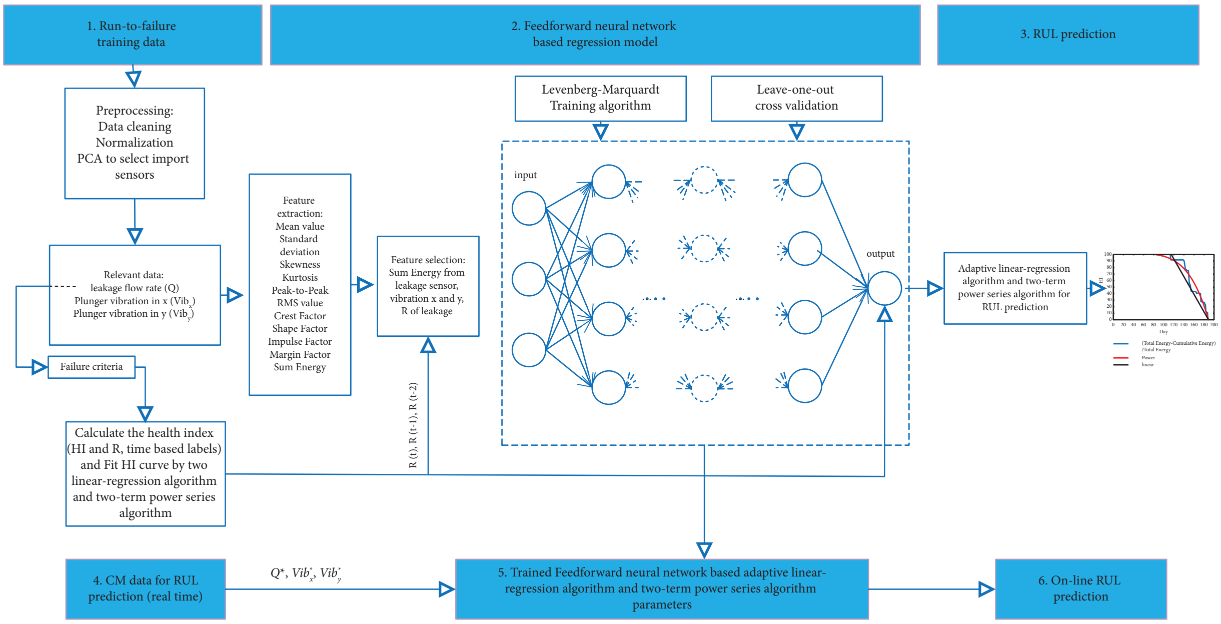

1. Run-to-failure training data
2. Feedforward neural network based regression model
3. RUL prediction
Preprocessing: Data cleaning Normalization PCA to select import sensors
Relevant data: leakage flow rate (Q) Plunger vibration in x (Vib$_x$) Plunger vibration in y (Vib$_y$)
Failure criteria
Calculate the health index (HI and R, time based labels) and Fit HI curve by two linear-regression algorithm and two-term power series algorithm
Feature extraction: Mean value Standard deviation Skewness Kurtosis Peak-to-Peak RMS value Crest Factor Shape Factor Impulse Factor Margin Factor Sum Energy
Feature selection: Sum Energy from leakage sensor, vibration x and y, R of leakage
R (t), R (t-1), R (t-2)
Levenberg-Marquardt Training algorithm
Leave-one-out cross validation
input
output
Adaptive linear-regression algorithm and two-term power series algorithm for RUL prediction
HI
Day
(Total Energy-Cumulative Energy) /Total Energy
Power
linear
4. CM data for RUL prediction (real time)
Q^*, Vib_x^*, Vib_y^*
5. Trained Feedforward neural network based adaptive linear-regression algorithm and two-term power series algorithm parameters
6. On-line RUL prediction

Figure 1: The flowchart of the proposed method.

which is equivalent to one-day operation. In other words, the 1024-point-normalised data, y_i, corresponding to one-day operation, are adopted to carry out phase-space reconstruction so as to obtain the so-called Hankel matrix as follows:

$$\mathbf{H} = \begin{bmatrix} y_1 & y_2 & \cdots & y_n \\ y_2 & y_3 & \cdots & y_{n+1} \\ \vdots & \vdots & \vdots & \vdots \\ y_m & y_{m+1} & \cdots & y_N \end{bmatrix}, \tag{4}$$

where $n = 720, m = 305$, and $N = 1024$ in this study. By using SVD, $\mathbf{H} = \mathbf{U}_{m\times k}\Lambda_{k\times k}\mathbf{W}^T_{k\times n}$ is obtained, where $\mathbf{U}$ and $\mathbf{W}$are both orthogonal matrices, while $\Lambda = \begin{bmatrix} S & 0 \\ 0 & 0 \end{bmatrix}, 1 < n < N$ and $\mathbf{S} = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_q)$. In this study, the value of q is 720. $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_q \geq 0$ are nonzero singular values of $\mathbf{H}$. Then, singular-value matrix $\mathbf{M}$ of each subsequence is defined as follows:

$$\mathbf{M} = \begin{bmatrix} \sigma_1^1 & \sigma_1^2 & \cdots & \sigma_1^d \\ \sigma_2^1 & \sigma_2^2 & \cdots & \sigma_2^d \\ \vdots & \vdots & \vdots & \vdots \\ \sigma_q^1 & \sigma_q^2 & \cdots & \sigma_q^d \end{bmatrix}, \tag{5}$$

where d is the total number of subsequences and q is the number of nonzero singular values in the Hankel matrix of each subsequence.

(c) Calculation of correlation coefficient R_j from singular-value matrix $\mathbf{M}$: The correlation coefficient with the singular value associated with each subsequence is calculated using Pearson's linear correlation coefficient formula as follows:

$$R_j = \frac{\sum_{i=1}^{q}\left(x_i - \overline{x}\right)\left(y_{j,i} - \overline{y_j}\right)}{\sqrt{\sum_{i=1}^{q}\left(x_i - \overline{x}\right)^2 \sum_{i=1}^{q}}}\left(y_{j,i} - \overline{y_j}\right)^2, \tag{6}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the singular-value vectors corresponding to different operating days. Vector $\mathbf{R}$ is the SVD-normalised correlation coefficient. When the system had high numerical stability, the change in the singular value R_j was small because there was slight perturbation of the signal.

2.3. Regression Model. A regression model describes the relationship between dependent variable (response variable) $\mathscr{Y}$ and one or more independent variables (explanatory or predictor variables) $\mathscr{X}$. The regression model adopts a straight line to approximate n data points, while it makes the sum of the squared residuals of the model minimal [27]. The simple linear regression model can be described by the following polynomial equation:

$$\mathscr{Y} = \alpha_1 + \beta_1 \mathscr{X}, \tag{7}$$

where β_1 is the coefficient and α_1 is the constant term in the model.

In contrast, the two-term power series algorithm approximates the relationship amongst a variety of pieces of data using the following analytical expression:

$$\mathscr{Y} = \alpha_2 \mathscr{X}^{\beta_2} + c_2, \tag{8}$$

where α_2 is the coefficient, β_2 is the power, and α_1 is the constant term in the model.

2.4. Feature Extraction and Selection Method. In order to construct an effective feedforward neural network (FNN) model, relevant features are required. Many previous studies have suggested that deep learning techniques could be good candidates when it comes to extracting discriminative features directly from raw data [28, 29]. It is known that feature representation or engineering can be used to transform raw data into features that better represent the underlying problem of predictive models. This could also result in improved model accuracy when it comes to unseen data. However, selecting appropriate features is mainly problem-oriented. The ReliefF algorithm is a filter-based feature selection method that helps choose a subset of features from the original feature set [30]. ReliefF is employed for classification and finds the weights of predictors in the case where y is a multiclass categorical variable. Similar to ReliefF, RReliefF (relief for regression) is utilised for regression and works with continuous y [30]. RReliefF penalises the predictors that give different values to neighbours with the same response values and rewards predictors that give different values to neighbours with different response values. RReliefF obtains the predictor weights based on intermediate weights as follows:

(a) We set weights W_{dy}, W_{dj}. W_{dyvdj}, and W_j equal to 0.

(b) We select random observation x_r iteratively and find the k-nearest observations to x_r and update W_{dy}, W_{dj}, and W_{dyvdj} for each nearest neighbour x_q:

$$\begin{aligned} W^{i}_{dy} &= W^{i-1}_{dy} + \Delta_y(x_r, x_q) \times d_{rq}, \\ W^{i}_{dj} &= W^{i-1}_{dj} + \Delta_j(x_r, x_q) \times d_{rq}, \\ W^{i}_{dyvdj} &= W^{i-1}_{dyvdj} + \Delta_y(x_r, x_q) \times \Delta_j(x_r, x_q) \times d_{rq}, \end{aligned} \tag{9}$$

where W_{dy} is the weight of different values for response y, W_{dj} is the weight of different values for predictor F_j, W_{dyvdj} is the weight of different response values y and different values for predictor F_j, i is the iteration step number, and $\Delta_y(x_r, x_q)$ is the difference in the value of continuous response y between observations x_r and x_q.

(c) Let y_r be the value of the response for observation x_r and y_q be the value of the response for observation x_q:

$$\begin{aligned} \Delta_y(x_r, x_q) &= \frac{|y_r - y_q|}{\max(y) - \min\ (y)}, \\ \Delta_j(x_r, x_q) &= \begin{cases} 0, & x_r = x_q, \\ 1, & x_r \neq x_q, \end{cases} \\ d_{rq} &= \frac{\widetilde{d_{rq}}}{\sum_{l=1}^{k} \widetilde{d_{rl}}}. \end{aligned} \tag{10}$$

(d) We calculate predictor weight W_j after updating all the intermediate weights as follows:

$$W_j = \frac{W_{dyvdj}}{W_{dy}} = \frac{W_{dj} - W_{dyvdj}}{m - W_{dy}}. \tag{11}$$

2.5. Neural Network. The FNN consists of an input layer, one or several hidden layers, and an output layer, as shown in Figure 2. The first layer has a connection from the network input. Each subsequent layer has a connection from the previous layer. The final layer produces the network's output. A feedforward network with one hidden layer and enough neurons in the hidden layers can fit any finite input-output mapping problem [19].

In the FNN, the output can be expressed for neuron k as follows:

$$O_k = g_2\left(\sum_{j=1}^{M} W_{kj} g_1\left(\sum_{i=1}^{N} W_{ij} x_i + W_{j0}\right) + W_{k0}\right), \tag{12}$$

where x_i is the input value to input node i, O_k is the output at output node k, j is the number of hidden nodes, g_1 is the nonlinear activation function for the hidden layer and g_2 is the linear activation function for the output layer, N and M are the number of neurons in the input and hidden layers, respectively, W_{j0} and W_{k0} are the biases of the jth neuron in the hidden layer and the kth neuron in the output layer, respectively, and W_{ij} and W_{kj} represent the weight between different layers.

3. Proposed Method for Prediction of the Remaining Useful Life of Packing Sets in Plunger-Type High-Pressure Compressors

3.1. Selecting Relevant Sensors Using Principal Component Analysis (PCA). It is challenging to predict the RUL of high-pressure packing using 33 sensors due to the fact that the prediction model might be easily disrupted by irrelevant sensors. With regard to implementation, the recorded signals collected from 33 sensors under normal and abnormal operation conditions of packing were included in two data sets, which were then merged into one. With PCA, the raw data of 33 sensors can be reduced to three important PCs. The aforementioned PCs can be used to approximately describe the normal and abnormal operation conditions of packing.

In the current study, the eigenvector and eigenvalue were determined via the MATLAB function "pca" (MATLAB statistics and machine learning toolbox). By default, the "pca" command centres the data and uses the SVD algorithm to deal with the eigenvalue analysis (function "svd" in the MATLAB symbolic math toolbox). SVD is a more general solution than PCA [31, 32].

3.2. Health State Assessment Using Singular Value Decomposition (SVD)-Normalised Correlation Coefficient. To predict the remaining life of high-pressure packing accurately, the SVD-based health state assessment criterion [10, 12, 13] was applied in this study. The method first divides the packing's life into several subsequences. For each of these subsequences, a phase-reconstruction process is adopted to construct the so-called Hankel matrix, which, in most cases, is a real symmetric matrix. Subsequently, SVD analysis is performed with respect to these sequential Hankel matrices to come up with the health state of packing. Due to the robustness associated with the SVD vectors, when data contain a small amount of perturbation, it is believed that the normalised correlation coefficient of the SVD vectors

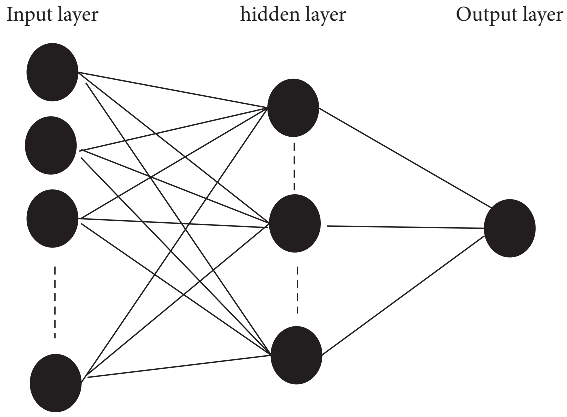


Figure 2: Structure of the FNN model.

between normal conditions is higher than that between normal and fault conditions. Based on the criterion described above, when packing is in good condition, the changes in the signal are small and the changes in singular values are also small. Along the course of the packing operation, greater variance appears between signals related to packing conditions, and the singular value and the value of the SVD correlation coefficient will scale up accordingly. With this definition of the SVD-normalised correlation coefficient, the negative effect of local noise and small perturbation on health state assessment can be avoided. In this study, the SVD was determined by the MATLAB function "svd" (MATLAB symbolic math toolbox).

In Equation (6), $\mathbf{x}$ represents the starting part of the raw signal, especially those few days immediately after the new packing component was installed in machinery. Therefore, $\mathbf{x}$ can be considered the base value (or normal state) of intrinsic system characteristics. On the other hand, $j = 1, \ldots, d$ represents the day-count index. In that regard, $\mathbf{y}$ denotes the nonzero singular-value vector corresponding to operating days other than the reference period. Each R_j implies that a new 1024-point raw data set is included in the computation process involving Equations (4)–(6). Wear takes place along the course of operation, and one can therefore observe that the R_j values of high-pressure packing decrease in the long-time base. Thus, vector $\mathbf{R}$ consisting of R_j can be obtained to represent the health state over the operation time of high-pressure packing.

3.3. A Novel Energy-Based Health Index (HI). Due to the uncertainty of the operation process, the actual health degradation curve of packing will never resemble a monotonically decreasing function. To that end, a novel criterion is proposed here to assess the health index of packing using the SVD correlation coefficient defined in Equation (6). The steps involved in calculating this new criterion are illustrated in Figure 3 and are as follows:

(1) According to the procedures adopted to perform SVD and compute Pearson's linear correlation coefficient, correlation coefficient vector $\mathbf{R}$ can be obtained from the chronological Hankel matrix.

(2) We calculate the sum energy as follows:

$$\text{sum energy} = \sum_{i=1}^{n} \left(1 - R_i\right)^2, \tag{13}$$

where n is the last day. Referring to the red area indicated in Figure 3, the "sum energy" represents the squared area between the R curve and constant 1. Here, the sum energy is more like the total degradation of the high-pressure packing in a run-to-failure event.

(3) We accumulate the energy of each day as follows (Figure 3):

$$\text{Energy}_m = \sum_{i=1}^{m} \left(1 - R_i\right)^2. \tag{14}$$

For any specific day count, this term denotes the accumulated degradation or the partial sum of energy.

(4) We calculate the health index using the sum energy and accumulated energy as follows (Figure 3):

$$\text{HI}_m\,(\%) = \frac{\text{sum energy} - \text{energy}_m}{\text{sum energy}} * 100, \tag{15}$$

where m is the day count.

3.4. Feature Extraction and Selection of Neural Networks. It is worth noting that, in the current study, only 13 data sets of packing with the abnormal operation were observed, while the sampling rate involved in relevant sensors was far too low. As a result of this, deep learning may be ineffective in finding, selecting, or extracting relevant features related to the health state of packing [33]. As the time resolution of data is one minute in this study, the sample rate is very low, and therefore, extraction of frequency features is unavailable. Considering the feasibility and computational burden, traditional statistical features in the time domain, such as the mean, peak-to-peak, and root mean square, are adopted in this study [18]. However, not all-time features are responsive to HI. Thus, the current study ranks the importance of predictors for HI using the RReliefF algorithm [30]. The weights of predictors are determined by the MATLAB function "relieff," with 10 nearest neighbours (MATLAB statistics and machine learning toolbox).

3.5. HI Prediction by Neural Networks. This paper proposes an FNN to build an HI-predicted model that describes the relationship between HI or the fitted HI curve using a linear regression model and the two-term power series algorithm in Equations (7) and (8), as well as the temporal information hidden in the selected feature after the RReliefF algorithm [34]. The FNN model uses the selected feature of packing after the RReliefF algorithm as an input. Meanwhile, amongst HI and two fitted HI curves, one of them will be selected as the output of FNN. This study designs one or two

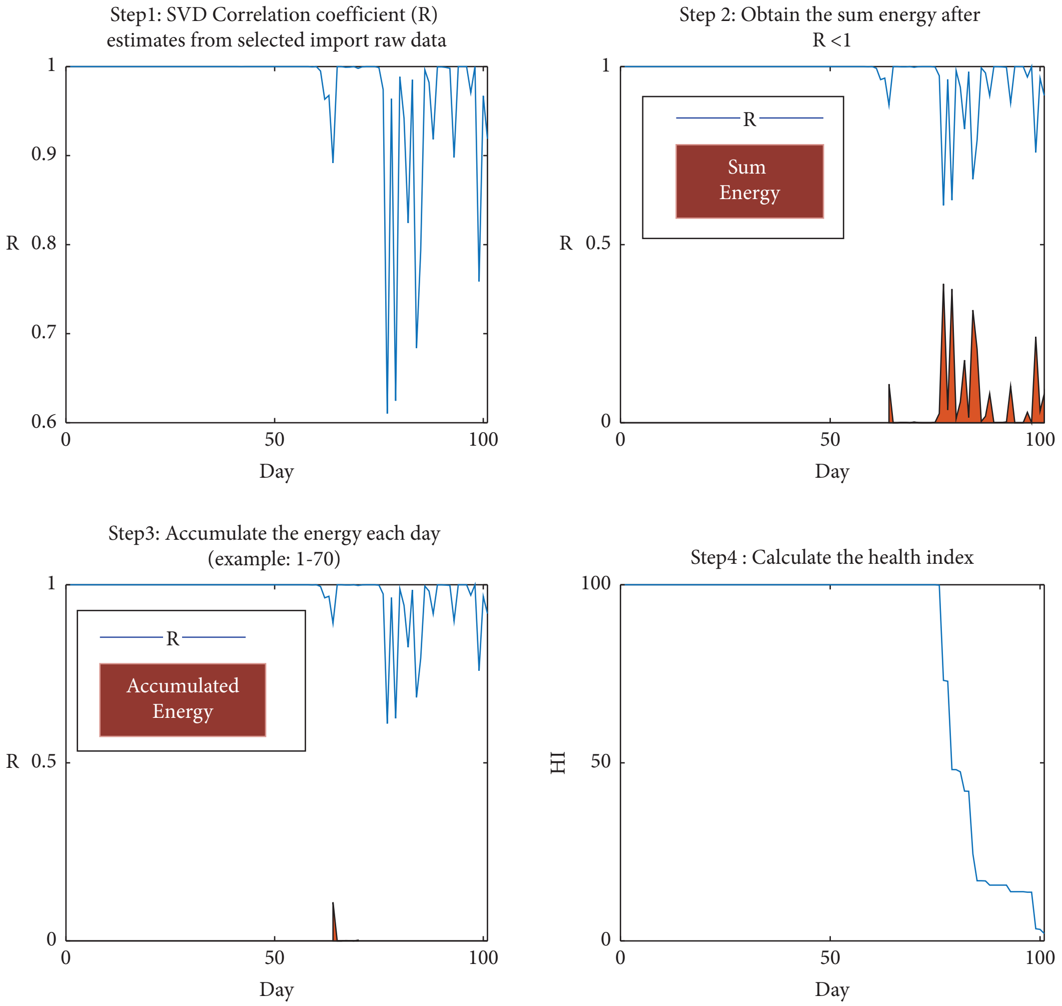


Figure 3: Steps related to the assessment of the health index.

hidden layers with different nodes to evaluate which structure is suitable for developing an HI-predicted model. In the current study, the FNN can be constructed by the MATLAB function “train” with “feedforwardnet” (MATLAB deep learning toolbox, Levenberg–Marquardt algorithm is the default training function).

A leave-one-out cross-validation scheme (LOOCV) was applied to examine the effectiveness of the HI-predicted model [35]. The LOOCV was used to make predictions on data not used to train the model, which utilises each individual data set as a “test” set. After satisfactory results were achieved, the linear regression algorithm and the two-term power series algorithm were integrated into the FNN model in order to predict the RUL of high-pressure packing.

3.6. HI-Based RUL Prediction of High-Pressure Packing.

When it comes to deriving data-driven, RUL-predicting techniques, approaches can be divided into two strategies, depending on whether an HI is used: (1) direct RUL predictions that model the relationship between input signals and RUL and (2) HI-based RUL predictions that build the model of input signals against HI and then map the estimated HI to RUL [36]. The relationship between HI and RUL is as follows: RUL is equal to 0 when HI is 0. In this study, the “HI-based prognostic” method was adopted to predict RUL. In step 4, shown in Figure 3, the characteristic of HI depends on the degradation features of the mechanical wearing process and seems nonlinear and involved. Thus, it is difficult to estimate the last day where $\mathrm{HI}=0$ using a universal nonlinear function. To that end, the current study adopted a linear regression algorithm and a two-term power series algorithm to fit the HI and then mapped the HI to RUL easily, as shown in Figure 4.

Based on Figure 4, $\mathscr{Y}$ and $\mathscr{X}$ refer to the health index and elapsed operation time, respectively. To that end, $\{\mathscr{Y}_i, \mathscr{X}_i\}$ denote data points involved in the calculations of Equations (7) and (8). Meanwhile, regression coefficients α_1 and β_1 are determined by the MATLAB function “regress” (MATLAB statistics toolbox), and in the two-term power series algorithm, α_2, β_2, and c_2 are determined by the MATLAB function “fit with option power 1” (MATLAB curve fitting toolbox), which minimises the sum of squared residuals of the model.

In the current work, the fitting target, namely, HI curves, can be the predicted or responding output of the NN model.

The latter is used to mimic the global behaviour between the elapsed operation time and the health index of high-pressure packing. The word “global” means that many run-to-failure data sets obtained from different (but identical) plunger-type compressors were used to train the NN model. The resultant NN model can be applied to input new raw data and obtain a predicted HI value. The predicted HI values obtained from the NN model are short-term predictions; namely, the predicted HI is close to the instant current time in this case. In order to forecast the HI and RUL values in the longer future time, we introduce the predicted HI into Equations (7) and (8) to obtain related parameters (α_1 and β_1 and α_2, β_2, and c_2). Then, one can forecast the future HI values by using the analytical expressions shown in Equations (7) and (8), from which RUL can be obtained accordingly.

4. Case Study and Results

4.1. Plunger-Type Hypercompressor. The plunger-type hypercompressor is an important piece of equipment and is adopted in chemical processes involving ethylene gas used in continuous flow procedures. Because the plunger-type high-pressure compressor (red square in Figure 5) was considered a critical item of equipment in the reaction process of a local chemical factory, a total of 192 sensors were set up to record and monitor real-time operation conditions. However, the sample rate of these sensors was one per minute, which is far too low if frequency features of the data are to be considered for use in machining learning approaches. When the compressor is subject to abnormal signal or operation conditions, a shutdown is needed. According to maintenance reports, high-pressure packing is the component that is responsible for most abnormal shutdown events. The packing cost is high enough to motivate the factory to fund this investigation. The aim of the current study is to come up with a machine learning-based prognostic strategy.

By ruling out sensors that do not correlate to packing or are outliers in the eyes of domain experts, the number of sensors considered for deriving the machine learning-based prognostic approach was reduced from 192 to 33. The latter includes the leakage flow rate through the high-pressure packing set, X- and Y-directional vibrations measured on the plunger, and the difference in temperature between the gas inlet and outlet.

In addition, a total of 13 abnormal operation data sets of high-pressure packing in plunger-type hypercompressors were observed and collected by the local chemical factory. These constituted run-to-failure data, which were applied to train and verify the NN model.

4.2. Selecting Important Sensors Using PCA. Despite the fact that domain expert knowledge was applied to reduce the sensor number from 192 to 33, the remaining sensor number was still too large to cope with. As such, the PCA method was applied here to further reduce the order of the problem. To achieve this, the recorded signals from the 33 sensors under normal and abnormal operation conditions in packing were merged. By using PCA, the raw data of 33 sensors were reduced to three important principal components (PCs), as shown in Table 1. These components, originating from important sensors, can be used to approximately describe the normal and abnormal operation conditions of packing. The features of these operation conditions are illustrated in Figure 6. Indeed, Figure 6 shows that the normal and failure operation data sets can be well discriminated in three PCs. The developmental sequences of PCA change in response to operating time. Meanwhile, Table 1 shows that PC1 can largely explain the total variability because it has a high explained variance ratio. The variance ratio represents the percentage of variance that is attributed by each of the selected components. It can be observed, in Table 1, that with three PCs, the cumulative variance can reach 99.85%. In addition, according to the value of the eigenvector of PC1, it can be observed that the leakage flow rate and the X/Y vibrations, which relate to the plunger motion, have higher values than those of other 30 sensors. It turns out that the eigenvalue of leakage flow is greater than 75% of the total value. Thus, leakage flow is the most important of the 33 sensors when the abnormal operation of high-pressure packing is considered. Figure 7 shows time-domain features in the raw data of the leakage flow rate and the X/Y vibrations under abnormal operations.

4.3. Health Index (HI) Assessment Using the SVD Correlation Coefficient Estimates, Linear Regression Algorithm, and Two-Term Power Series Algorithm. A total of 13 abnormal operation data sets were obtained from eight identical plunger-type hypercompressors installed in a local chemical factory. The results of the PCA showed that leakage flow is the most important sensor. Thus, leakage flow data are suitable for being used for assessing the HI of high-pressure packing in plunger-type hypercompressors. In order to obtain the run-to-failure data on leakage flow, failure criteria were defined; the first of which was if the three conditions shown in Table 2 happen simultaneously, the packing failure will be declared and a termination of operation will be needed. Figure 8 illustrates four sets of run-to-failure raw data of leakage flow. As mentioned in the previous sections, this study adopted leakage flow to assess HI using SVD correlation coefficient estimates and obtained fitted HI curves using the linear regression algorithm and the two-term power series algorithm, as shown in Figure 9. Amongst HI and the two fitted HI curves, one of them will be adopted as the output of NN, and this will be followed by evaluation of which is suitable for the output.

4.4. Feature Extraction and Selection Using the RReliefF Algorithm. The results of the PCA showed that the leakage flow rate and the X/Y vibrations are sensors that possess high values in eigenvectors. Thus, the current study picked time-domain statistics such as the mean, peak-to-peak, and root mean square of the three sensors as problem features. In total, 13 time-domain features are selected in this study and are listed in Table 3, where x_i can be referred to as the readings of the leakage flow rate and the X/Y vibrations. Indeed, 37 features were defined, including 11 features

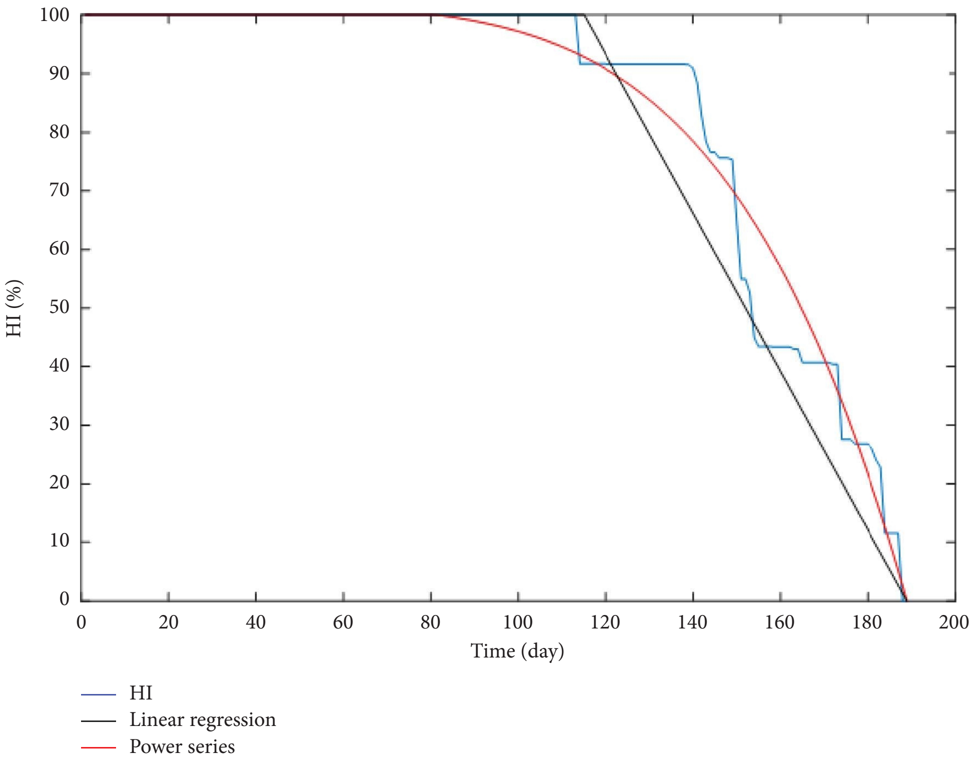


Figure 4: Fitting features appeared in the linear algorithm and the two-term power series regression algorithm.

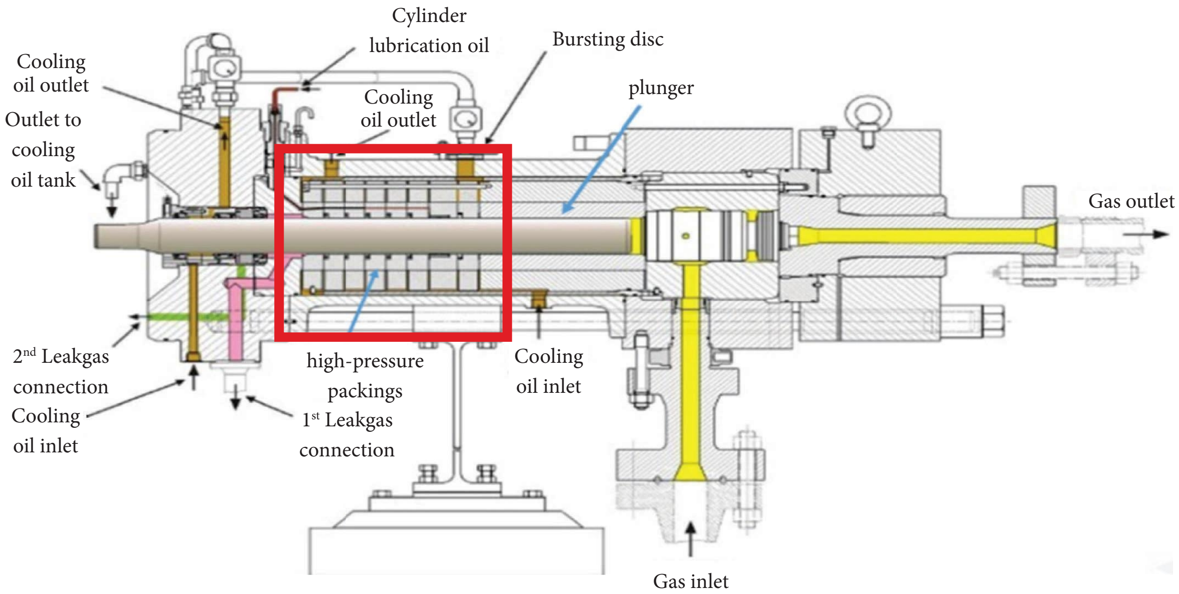


Figure 5: High-pressure packing in plunger-type hypercompressors.

associated with each of the three relevant sensors, three SVD correlation coefficient estimates (R_i, R_{i-1}, R_{i-2}), and sum energy of R_i. The result of the RReliefF algorithm analysis showed that the sum energy of all leakage flow rates, the X/Y vibrations, and R_i were all related to HI. Thus, those four features are the inputs of NN. The present study defined one hidden layer with ten hidden nodes and two different layers with ten and five hidden nodes to evaluate which structure is suitable for predicting HI. Finally, we introduce predicted HI into Equations (7) and (8) to obtain related parameters (α_1 and β_1 and α_2, β_2, and c_2) so that long-term forecasting is possible. The latter can be achieved by introducing future day (i) into Equations (10) and (11) to obtain the future HI and to forecast the time when HI = 0, from which RUL is obtained. Supplementary Tables 1–3 show the performances of RUL amongst the different outputs of NN, the different

Table 1: Eigenvalue, explained variance, and cumulative variance of three important PCs.

Number of components	Eigenvalue	Explained variance %	Cumulative variance %
PC1	206.5608	98.7937	98.7937
PC2	1.911	0.95228	99.746
PC3	0.2311	0.11053	99.8566

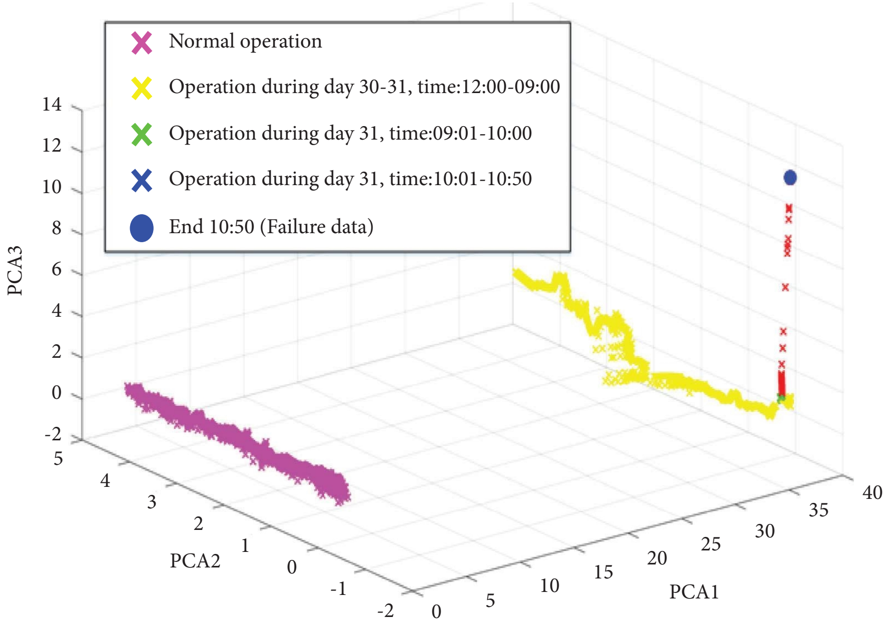


Figure 6: Three PC results after PCA and developmental sequences in PCA change in response to operating time.

structures of hidden layers, the number of nodes, and different prediction methods for the RUL of high-pressure packing. When the FNN had one hidden layer, the number of hidden nodes was ten. When the FNN had two hidden layers, the numbers of hidden nodes were ten and five, respectively.

In Supplementary Tables 1 and 2, in order to fit HI, which is fitted by using Equations (10) or (11), as the NN output (yellow line and red line in Figure 9), the adopted method is the same as that used for predicting the relationship between predicted HI and RUL. Then, the FNN has one (Supplementary Table 1) or two hidden layers with different nodes (Supplementary Table 2) to evaluate which structure is suitable for developing the HI-predicted model. In Supplementary Table 3, the NN output is real HI, which is not fitted by using Equations (10) or (11) (blue line in Figure 9). Then, the FNN has two hidden layers with different nodes to develop the HI-predicted model. The findings showed the performance of the testing results, while the other 12 data sets served as the training data.

After comparing the results in Supplementary Tables 1–3, it was found that better performance is shown in Supplementary Table 2. Of note, many data sets provide accurate prediction of RUL. Indeed, RUL can estimate the time of HI = 0 using (8), while the value of HI is still 60. Thus, the current study suggested that FNN had “two” hidden layers and that the numbers of hidden nodes were ten and five, respectively. Then, fitting HI was used as the NN output to predict RUL by also using Equation (8). This structure will be suitable for predicting the RUL of high-pressure packing in plunger-type hypercompressors.

In order to prove the validity of the proposed method in this study, we compared the proposed method using long short-term memory (LSTM) [37, 38], linear regression [39], robust linear regression [40], linear support vector machine (SVM), and quadratic SVM [41, 42]. Table 4 shows the structure and hyperparameter settings of all the above-mentioned methods. The optimal parameters of the algorithms could be selected from the MATLAB function. LSTM used the training options for the Adam (adaptive moment estimation) optimiser, including learning rate information, *L*2 regularisation factor, and minibatch size. Robust linear regression turned on the robust option, which adopted robust fitting using the “bisquare” weight function with the default tuning constant. SVM set the related options, including kernel scale, box constraint, and epsilon, to automatic.

This study also calculated the sum of absolute errors (SAEs) between the predicted HI and the actual HI, as shown in Figure 10. The results of Figures 10(a)–10(d) show that the predicted HI of the proposed FNN, which consisted of two hidden layers, with the numbers of hidden nodes being ten and five, respectively, was similar to the actual HI because of the lowest SAE. In Figure 10(a), robust linear regression is the second best. In Figure 10(b), LSTM is the second best. In Figure 10(c), the Quadratic SVM is the second best. In

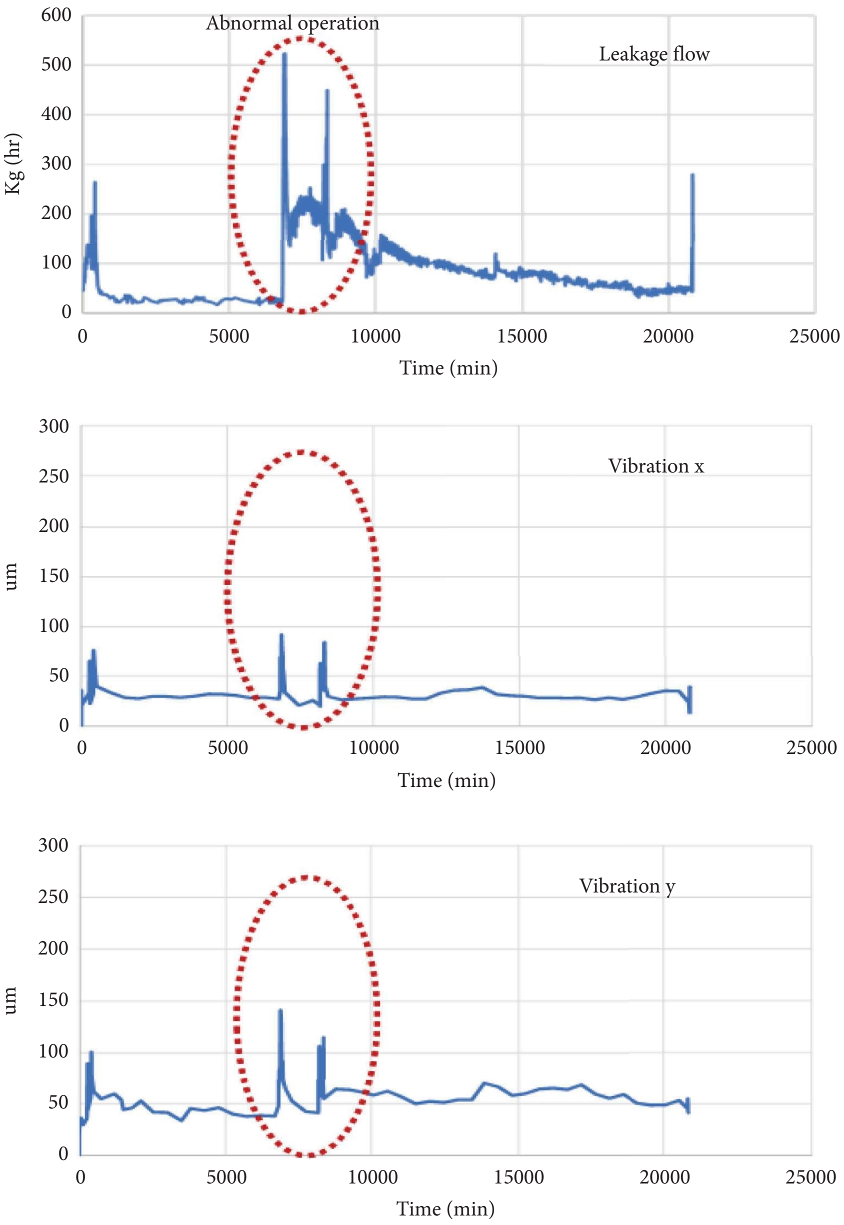


Figure 7: Raw data of the leakage flow rate and the X/Y vibrations under abnormal operations.

Table 2: Three conditions for failure criteria.

Conditions	Value of leakage flow	Occupation ratio within 7 days
1	>300	>40%
2	>400	>30%
3	>500	>20%

Figure 10(d), robust linear regression is the second best. Thus, the results of Figure 10 show that the proposed FNN structure can more accurately predict HI and RUL than other methods. Compared with other methods, the results of LSTM had greater variation. The reason was the small number of trained data sets in this study. Indeed, there were only 13 run-to-failure data sets obtained from eight identical hypercompressors installed in a local chemical factory, with an average run-to-failure life of 233 days. In addition, the results of Figure 10 show that most algorithms cannot predict the zero HI, while the actual HI reached zero. It was found that the proposed HI-based RUL prediction of high-pressure packing in this study could solve the above-mentioned issues, and subsequently, the follow-up-predicted HI is estimated using the two-term power series algorithm in Equation (8).

5. The Proposed Model for Predicting RUL in the Real Case

The proposed model was further tested using online data to examine the health states of four identical kinds of packing (2A1, 2A2, 2B1, and 2B2). Since these online data never appeared in the training or verification processes of the

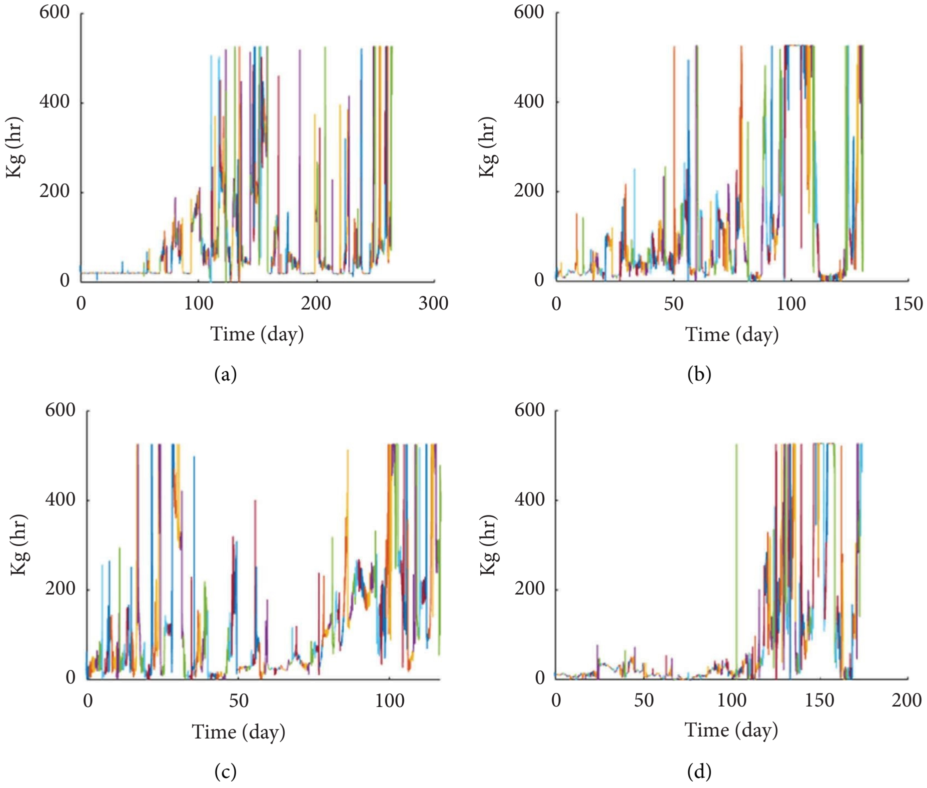


FIGURE 8: Four datasets (a–d) of run-to-failure data of leakage flow. Different colour lines in (a–d) denote leakage flow on different days.

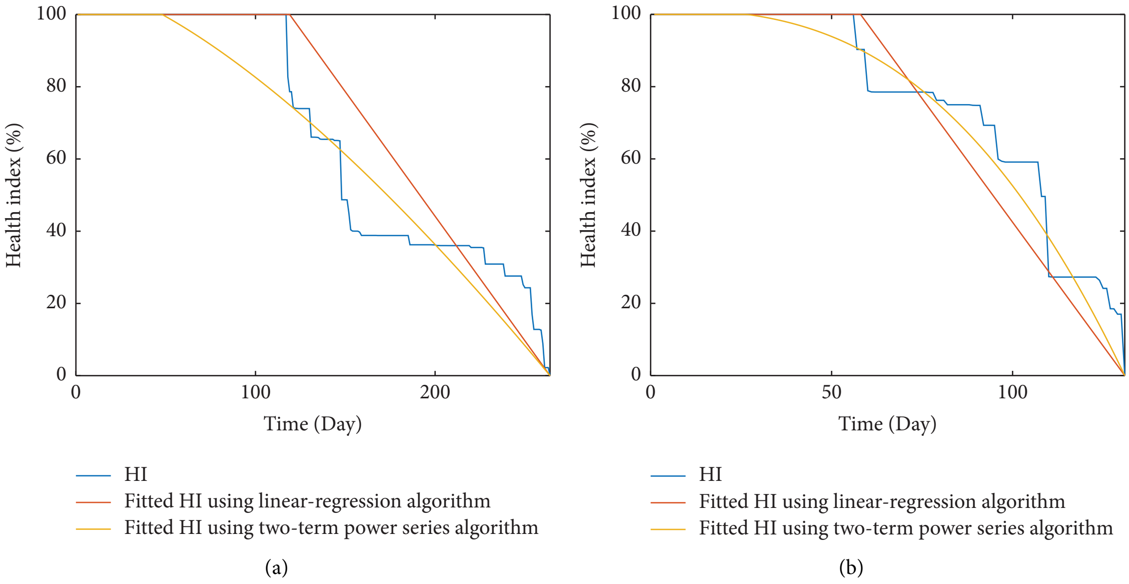
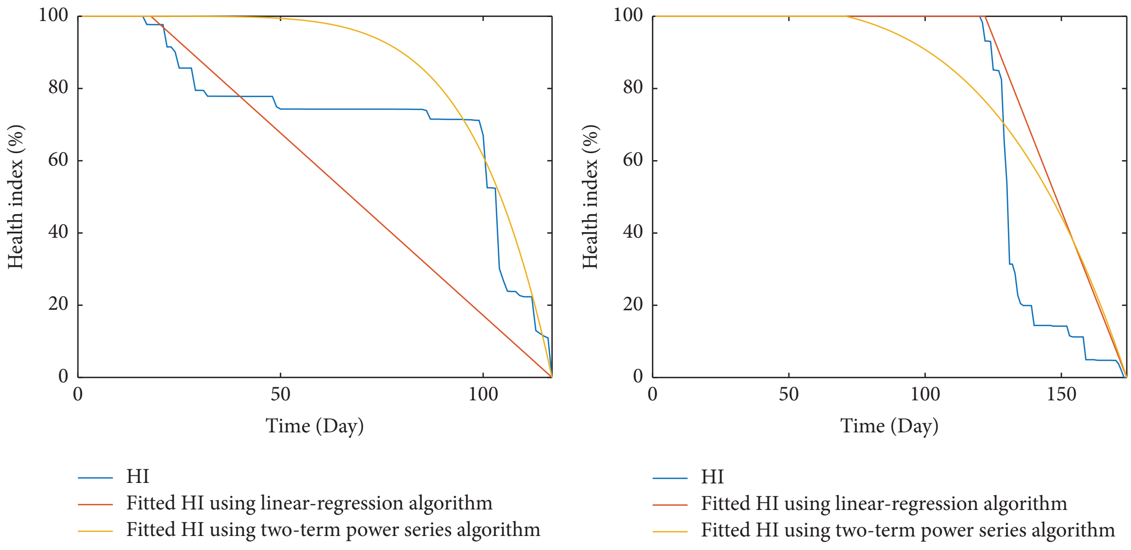


FIGURE 9: Continued.



(c)

(d)

Figure 9: Real HI (blud line) using SVD correlation coefficient estimates and fitted HI curve using linear regression algorithm (red line) and two-term power series algorithm(orange line) from four datasets (a–d) in Figure 8.

Table 3: Time-domain features applied in this study.

Feature	Formula
Mean value	$1/n \sum_{i=1}^{n} x_i$
Standard deviation (SD)	$\sqrt{1/ \sum_{i=1}^{n} (x_i - \overline{x})^2}$
Skewness	$(1/n) \sum_{i=1}^{n} (x_i - \overline{x})^3 / ((1/n) \sum_{i=1}^{n} (x_i - \overline{x})^2)^{3/2}$
Kurtosis	$(1/n) \sum_{i=1}^{n} (x_i - \overline{x})^4 / ((1/n) \sum_{i=1}^{n} (x_i - \overline{x})^2)^2$
Peak-to-peak	$\max(x) - \min\ (x)$
Root mean square (RMS)	$\sqrt{1/n \sum_{i=1}^{n} x_i^2}$
Crest factor (peak-to-average ratio)	$\max(x)/x_{rms}$
Shape factor	$x_{rms}/ (1/n) \sum_{i=1}^{n} \lvert x_i \rvert$
Impulse factor	$\max(x)/ (1/n) \sum_{i=1}^{n} \lvert x_i \rvert$
Margin factor	$\max(x)/ ((1/n) \sum_{i=1}^{n} \lvert x_i \rvert)^2$
Sum energy	$\sum_{i=1}^{n} (x_i)^2$
R_i (SVD correlation coefficient estimates)	Calculate from Equation (6)
Sum energy of R_i	Calculate from Equation (8)

Table 4: The structure and hyperparameter settings of proposed FNN, LSTM, linear regression, robust linear regression, linear SVM, and quadratic SVM.

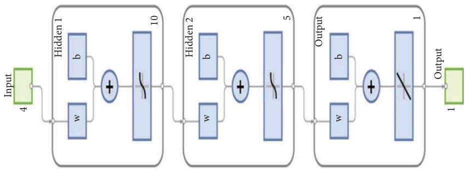
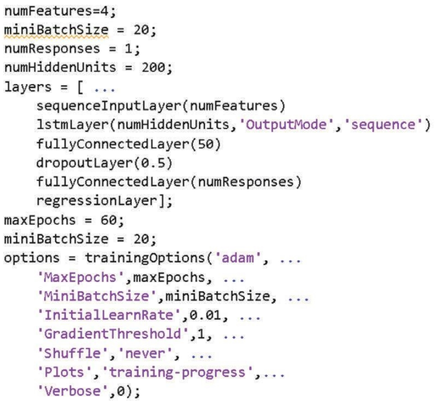
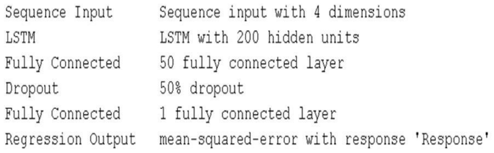
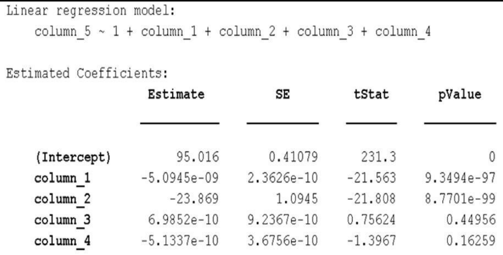
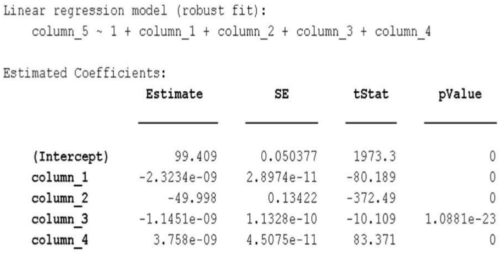
Method	Set parameters	Structure
FNN in this study	HI predicted: Function: feedforwardnet Hidden size: [10 5] Data division: random Training: Levenberg–Marquardt Performance: mean squared error Output: fitting HI using Equation (11) (deep learning toolbox)	 Input 4 Hidden 1 10 b w Hidden 2 5 b w Output 1 b w Output 1 Activation function of hidden layers: hyperbolic tangent sigmoid transfer function (tansig) Activation function of output layers: linear transfer function (purelin)

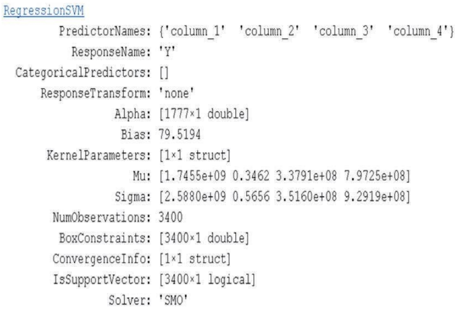
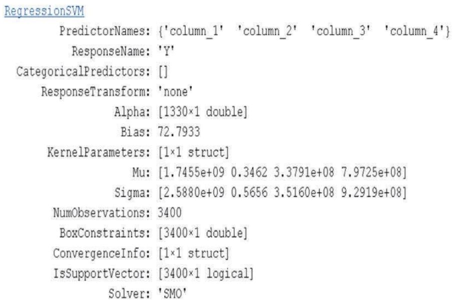
Table 4: Continued.

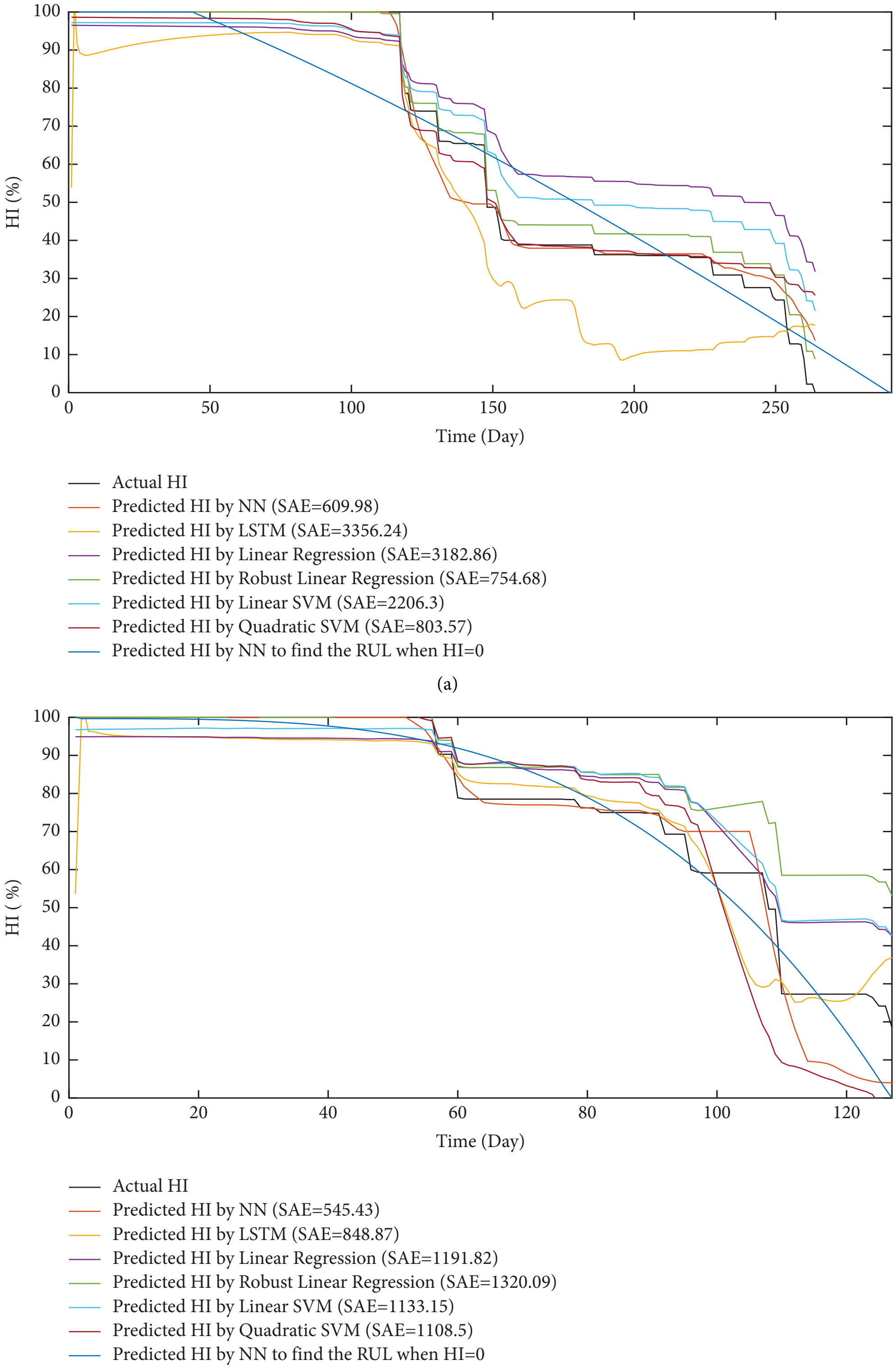
<table>
<tr><th>Method</th><th>Set parameters</th><th>Structure</th></tr>
<tr><td>LSTM</td><td>
<pre>
numFeatures=4;
miniBatchSize = 20;
numResponses = 1;
numHiddenUnits = 200;
layers = [...
    sequenceInputLayer(numFeatures)
    lstmLayer(numHiddenUnits,'OutputMode','sequence')
    fullyConnectedLayer(50)
    dropoutLayer(0.5)
    fullyConnectedLayer(numResponses)
    regressionLayer];
maxEpochs = 60;
miniBatchSize = 20;
options = trainingOptions('adam', ...
    'MaxEpochs',maxEpochs, ...
    'MiniBatchSize',miniBatchSize, ...
    'InitialLearnRate',0.01, ...
    'GradientThreshold',1, ...
    'Shuffle','never', ...
    'Plots','training-progress',...
    'Verbose',0);
</pre>
(Deep learning toolbox)</td><td>
<pre>
Sequence Input   Sequence input with 4 dimensions
LSTM   LSTM with 200 hidden units
Fully Connected   50 fully connected layer
Dropout   50% dropout
Fully Connected   1 fully connected layer
Regression Output mean-squared-error with response 'Response'
</pre></td></tr>
<tr><td>Linear regression</td><td>Function: fitlm
Preset: linear
Terms: linear
Robust option: off
(Statistics and machine learning toolbox)</td><td>
<pre>
Linear regression model:
    column_5 ~ 1 + column_1 + column_2 + column_3 + column_4

Estimated Coefficients:
                    Estimate   SE   tStat   pValue
                   __________   __________   _______   __________

    (Intercept)   95.016   0.41079   231.3   0
    column_1   -5.0945e-09   2.3626e-10   -21.563   9.3494e-97
    column_2   -23.869   1.0945   -21.808   8.7701e-99
    column_3   6.9852e-10   9.2367e-10   0.75624   0.44956
    column_4   -5.1337e-10   3.6756e-10   -1.3967   0.16259
</pre></td></tr>
<tr><td>Robust linear regression</td><td>Function: fitlm
Preset: linear
Terms: linear
Robust option: on
(Statistics and machine learning toolbox)</td><td>
<pre>
Linear regression model (robust fit):
    column_5 ~ 1 + column_1 + column_2 + column_3 + column_4

Estimated Coefficients:
                    Estimate   SE   tStat   pValue
                   __________   __________   _______   __________

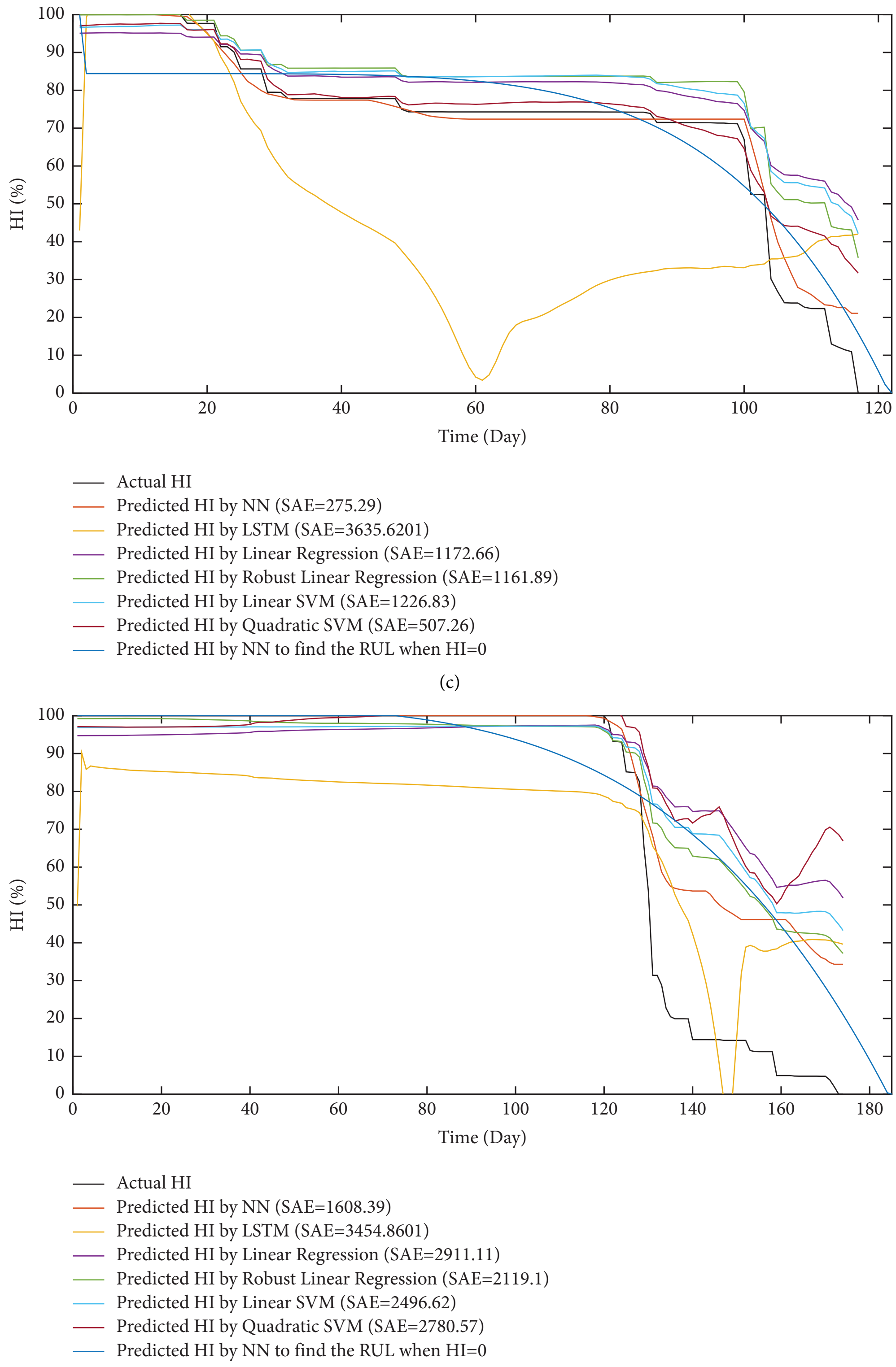
    (Intercept)   99.409   0.050377   1973.3   0
    column_1   -2.3234e-09   2.8974e-11   -80.189   0
    column_2   -49.998   0.13422   -372.49   0
    column_3   -1.1451e-09   1.1328e-10   -10.109   1.0881e-23
    column_4   3.758e-09   4.5075e-11   83.371   0
</pre></td></tr>
<tr><td>Linear SVM</td><td>Function: fitrsvm
Preset: linear SVM
Kernel function: linear
Kernel scale: automatic
Box constraint: automatic
Epsilon: auto
Standardize data: yes
(Statistics and machine learning toolbox)</td><td>
<pre>
  RegressionSVM
           PredictorNames: {'column_1' 'column_2' 'column_3' 'column_4'}
             ResponseName: 'Y'
    CategoricalPredictors: []
        ResponseTransform: 'none'
                    Alpha: [1777×1 double]
                     Bias: 79.5194
         KernelParameters: [1×1 struct]
                       Mu: [1.7455e+09 0.3462 3.3791e+08 7.9725e+08]
                    Sigma: [2.5880e+09 0.5656 3.5160e+08 9.2919e+08]
          NumObservations: 3400
           BoxConstraints: [3400×1 double]
          ConvergenceInfo: [1×1 struct]
          IsSupportVector: [3400×1 logical]
                   Solver: 'SMO'
</pre></td></tr>
<tr><td>Quadratic SVM</td><td>Function: fitrsvm
Preset: quadratic SVM
Kernel function: linear
Kernel scale: automatic
Box constraint: automatic
Epsilon: auto
Standardize data: yes
(Statistics and machine learning toolbox)</td><td>
<pre>
  RegressionSVM
           PredictorNames: {'column_1' 'column_2' 'column_3' 'column_4'}
             ResponseName: 'Y'
    CategoricalPredictors: []
        ResponseTransform: 'none'
                    Alpha: [1330×1 double]
                     Bias: 72.7933
         KernelParameters: [1×1 struct]
                       Mu: [1.7455e+09 0.3462 3.3791e+08 7.9725e+08]
                    Sigma: [2.5880e+09 0.5656 3.5160e+08 9.2919e+08]
          NumObservations: 3400
           BoxConstraints: [3400×1 double]
          ConvergenceInfo: [1×1 struct]
          IsSupportVector: [3400×1 logical]
                   Solver: 'SMO'
</pre></td></tr>
</table>



(a)

(b)

FIGURE 10: Continued.



(c)

(d)

Figure 10: The predicted HI by different methods (FNN in this study, LSTM, linear regression, robust linear, linear SVM, and quadratic SVM), actual HI (black line), and predicted RUL (blue line) using equation (11).

proposed scheme, the results roughly indicate the effectiveness and reliability of the method. Figure 11 shows these four sets of online leakage flow data. In the figure, the number of days represents the elapsed operation time of high-pressure packing, while the ordinate value indicates the leakage flow rate. After the proposed scheme is applied to Figure 11, the predicted results of HI and RUL are shown in Figure 12. It can be seen that, in Figure 12, the packing in

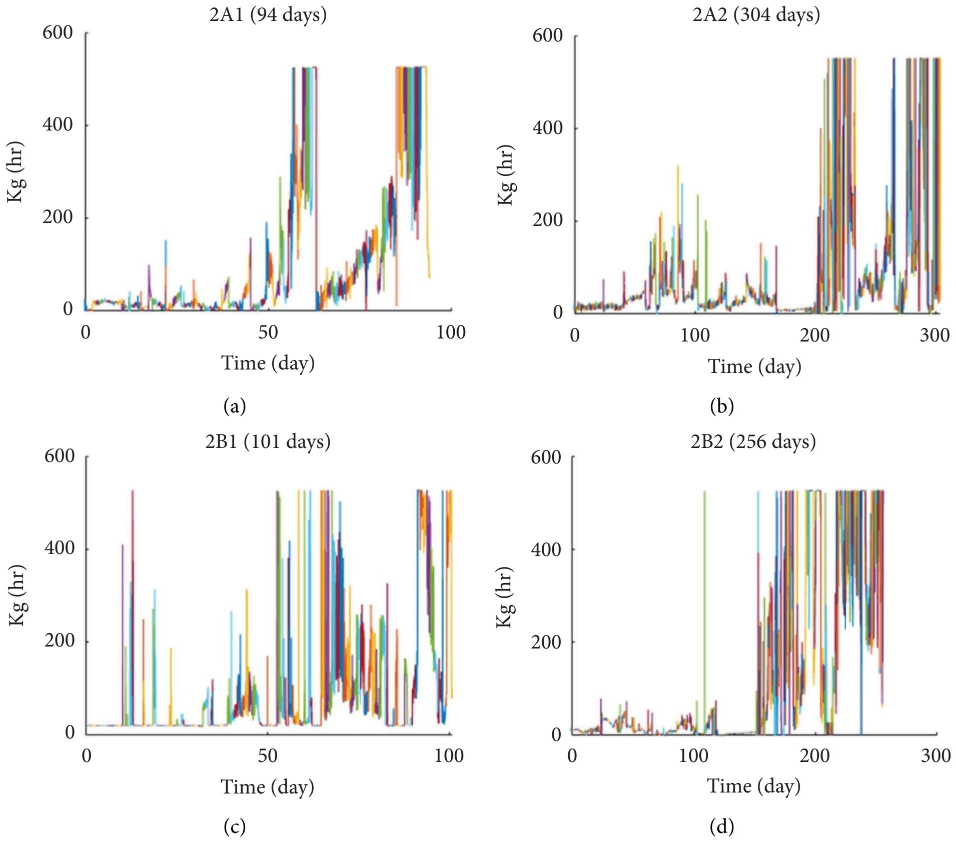


Figure 11: Four datasets (a–d) of run-to-failure data of leakage flow in real case. Different colour lines in (a–d) denote leakage flow on different days. The title is the name of high-pressure packing and the number of day is the duration of run-to-failure data.

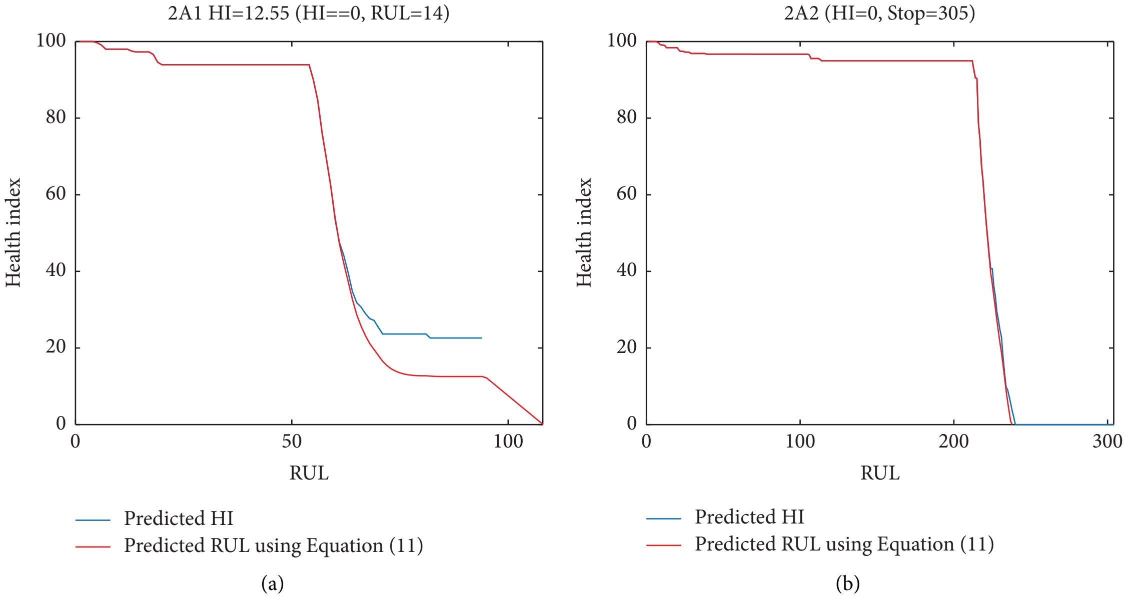
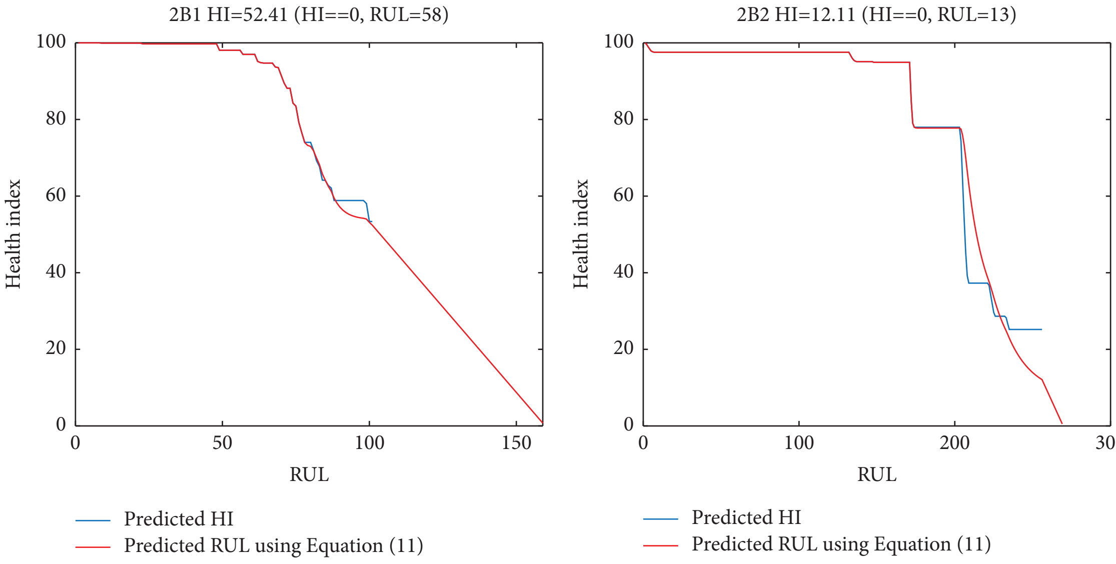


Figure 12: Continued.



(c)

(d)

Figure 12: Predicted HI (blue line) and predicted RUL (red line) of four high-pressure packing (a–d) in Figure 11. If title is HI==0 and RUL=predicted value of RUL, the meaning is that the high-pressure packing still can work predicted value of RUL day. That is, predicted value of RUL occurs after the current data. If title is HI==0 and Stop=Stop day of operating high-pressure packing, the meaning is that the high-pressure packing should stop because of failure. That is, predicted value of RUL occurs before the current data.

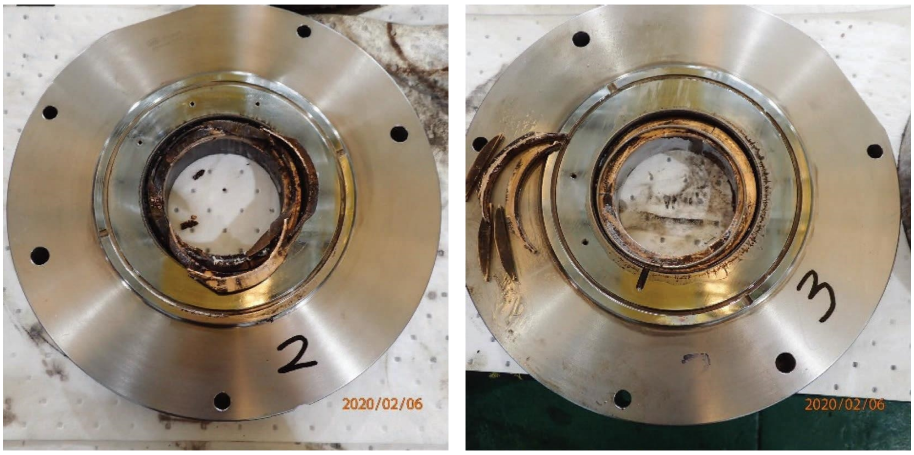


Figure 13: The failure in packing of 2A2.

2A1 can operate for 14 more days, while the corresponding packing components in 2B1 and 2B2 can run for 58 and 13 more days, respectively. In contrast with these three cases, the results obtained in the 2A2 machine indicate that the predicted RUL reached 0 at 305 operation days. Thus, suspected failure was predicted. This prediction of failure was later confirmed by the factory maintenance team when the equipment was disassembled, as shown in Figure 13.

6. Conclusion

Some popular machine and deep learning architectures (linear and robust linear regression, linear and quadratic SVM, and LSTM) are presented in this study. As the amount of data surpasses a certain size, deep learning accuracy increases incrementally with respect to the amount of data. However, there were only 13 run-to-failure data sets obtained from eight identical hypercompressors installed in a local chemical factory, with an average run-to-failure life of 233 days. This study proposed a machine learning-based prognostic strategy for predicting the RUL of high-pressure packing in plunger-type hypercompressors when the number of data sets was small. According to the results, the best performance can be achieved when the NN has two hidden layers and the numbers of hidden nodes are ten and five, respectively. The proposed NN scheme is combined with a two-term power series algorithm to regress the degradation trends. The maximum prediction RUL error is less than 15 days, and the average prediction RUL error is less than 7.23 days by using the proposed scheme. Based on the results, the proposed approach can provide sufficient information to the manufacturer, thus allowing the manufacturer to plan maintenance in advance. Finally, the proposed model was used to predict HI and RUL in real cases where the online data were never involved in the training and verification process. The predicted RUL of one high-pressure packing was 0. After disassembling the equipment, the failure of packing was found and confirmed. This real-test result further proves the effectiveness of the proposed method.

Data Availability

The data used to support this study are currently under embargo, while the research findings are commercialized.

Disclosure

A preprint of this study was previously published in Authorea, 2022.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported, in part, by the Aiming for the Talent Cultivation Project of the Ministry of Education of Taiwan and the Program funded by the National Science and Technology Council of Taiwan under grant number MOST 111-2221-E-131-030.

Supplementary Materials

There were three supplementary Tables in this study to show the performances of RUL amongst the different outputs of NN, the different structures of hidden layers, and the number of nodes and different prediction methods for the RUL of high-pressure packing. When the FNN had one hidden layer, the number of hidden nodes was ten. When the FNN had two hidden layers, the numbers of hidden nodes were ten and five, respectively. Supplementary Table 1 shows that NN output is used as the target HI curve and fitted by using equations (10) or (11) (yellow line and red line in Figure 9), and RUL prediction is done accordingly with equations (10) or (11) when NN has one hidden layer and the number of hidden node is 10. Supplementary Table 2 shows that NN output is used as the target HI curve and fitted by using equations (10) or (11) (yellow line and red line in Figure 9), and RUL prediction is done accordingly with equations (10) or (11) when NN has two hidden layers and the number of hidden nodes are 10 and 5, respectively. Supplementary Table 3 shows that NN output is real HI which did not fit by using equations (10) or (11) (blue line in Figure 9), and RUL prediction is done with equations (10) or (11) when NN has two hidden layers and the number of hidden nodes are 10 and 5, respectively. (*Supplementary Materials*)

References

[1] H. Hanachi, W. Yu, I. Y. Kim, J. Liu, and C. K. Mechefske, “Hybrid data-driven physics-based model fusion framework for tool wear prediction,” *International Journal of Advanced Manufacturing Technology*, vol. 101, no. 9, pp. 2861–2872, 2019.

[2] J. Lee, F. Wu, W. Zhao, M. Ghaffari, L. Liao, and D. Siegel, “Prognostics and health management design for rotary machinery systems—reviews, methodology and applications,” *Mechanical Systems and Signal Processing*, vol. 42, no. 1-2, pp. 314–334, 2014.

[3] D. An, N. H. Kim, and J.-H. Choi, “Practical options for selecting data-driven or physics-based prognostics algorithms with reviews,” *Reliability Engineering & System Safety*, vol. 133, pp. 223–236, 2015.

[4] J. Yu, “A hybrid feature selection scheme and self-organizing map model for machine health assessment,” *Applied Soft Computing*, vol. 11, no. 5, pp. 4041–4054, 2011.

[5] V. Atamuradov, K. Medjaher, F. Camci, N. Zerhouni, P. Dersin, and B. Lamoureux, “Machine health indicator construction framework for failure diagnostics and prognostics,” *Journal of Signal Processing Systems*, vol. 92, no. 6, pp. 591–609, 2020.

[6] Y.-C. Liu, Y.-J. Chang, S.-L. Liu, and S.-P. Chen, “Data-driven prognostics of remaining useful life for milling machine cutting tools,” in *Proceedings of the 2019 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 1–5, San Francisco, CA, USA, June 2019.

[7] P. Li, X. Jia, J. Feng et al., “Prognosability study of ball screw degradation using systematic methodology,” *Mechanical Systems and Signal Processing*, vol. 109, pp. 45–57, 2018.

[8] C. Zhang, P. Lim, A. K. Qin, and K. C. Tan, “Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2306–2318, 2017.

[9] W. Yu, I. Y. Kim, and C. Mechefske, “Remaining useful life estimation using a bidirectional recurrent neural network based autoencoder scheme,” *Mechanical Systems and Signal Processing*, vol. 129, pp. 764–780, 2019.

[10] A. Galli, M. Gravina, V. Moscato, and G. Sperli, “Deep Learning for HDD health assessment: an application based on LSTM,” *IEEE Transactions on Computers*, vol. 71, no. 1, pp. 69–80, 2020.

[11] P. Ding, M. Jia, and X. Yan, “Stationary subspaces-vector autoregressive with exogenous terms methodology for degradation trend estimation of rolling and slewing bearings,” *Mechanical Systems and Signal Processing*, vol. 150, Article ID 107293, 2021.

[12] P. Ding and M. Jia, “Mechatronics equipment performance degradation assessment using limited and unlabeled data,” *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2374–2385, 2022.

[13] D. E. Acuña and M. E. Orchard, “Particle-filtering-based failure prognosis via sigma-points: application to lithium-ion battery state-of-charge monitoring,” *Mechanical Systems and Signal Processing*, vol. 85, pp. 827–848, 2017.

[14] D. An, J.-H. Choi, T. L. Schmitz, and N. H. Kim, “In situ monitoring and prediction of progressive joint wear using Bayesian statistics,” *Wear*, vol. 270, no. 11-12, pp. 828–838, 2011.

[15] X.-S. Si, W. Wang, C.-H. Hu, and D.-H. Zhou, “Remaining useful life estimation–a review on the statistical data driven

approaches," *European Journal of Operational Research*, vol. 213, no. 1, pp. 1–14, 2011.

[16] A. Mosallam, K. Medjaher, and N. Zerhouni, "Data-driven prognostic method based on Bayesian approaches for direct remaining useful life prediction," *Journal of Intelligent Manufacturing*, vol. 27, no. 5, pp. 1037–1048, 2016.

[17] W. Mao, J. He, and M. J. Zuo, "Predicting remaining useful life of rolling bearings based on deep feature representation and transfer learning," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 4, pp. 1594–1608, 2020.

[18] W. Mao, J. He, J. Tang, and Y. Li, "Predicting remaining useful life of rolling bearings based on deep feature representation and long short-term memory neural network," *Advances in Mechanical Engineering*, vol. 10, no. 12, Article ID 168781401881718, 2018.

[19] E. Mohamed, M. Elsayed, A. Hassan, M. Mahmoud, and A. El-Husseiny, "A machine learning approach to predict the permeability from nmr t2 relaxation time distribution for various reservoir rock types," in *Proceedings of the Abu Dhabi International Petroleum Exhibition and Conference*, Abu Dhabi, UAE, November 2022.

[20] X. Xu, F. Xiao, and S. Wang, "Enhanced chiller sensor fault detection, diagnosis and estimation using wavelet analysis and principal component analysis methods," *Applied Thermal Engineering*, vol. 28, no. 2-3, pp. 226–237, 2008.

[21] Q. He, R. Yan, F. Kong, and R. Du, "Machine condition monitoring using principal component representations," *Mechanical Systems and Signal Processing*, vol. 23, no. 2, pp. 446–466, 2009.

[22] H. Lahdhiri, M. Said, K. B. Abdellafou, O. Taouali, and M. F. Harkat, "Supervised process monitoring and fault diagnosis based on machine learning methods," *International Journal of Advanced Manufacturing Technology*, vol. 102, no. 5-8, pp. 2321–2337, 2019.

[23] G. T. Reddy, M. P. K. Reddy, K. Lakshmanna et al., "Analysis of dimensionality reduction techniques on big data," *IEEE Access*, vol. 8, pp. 54776–54788, 2020.

[24] J.-W. Liang, S.-C. Huang, and C.-L. Lin, "Prediction of remaining useful life of packing sets in plunger-type high-pressure compressor based on PCA/SVD analysis and NN model," 2022, https://www.authorea.com/doi/full/10.22541/au.165535744.49181450.

[25] F. Artoni, A. Delorme, and S. Makeig, "Applying dimension reduction to EEG data by Principal Component Analysis reduces the quality of its subsequent Independent Component decomposition," *NeuroImage*, vol. 175, pp. 176–187, 2018.

[26] S. Patro and K. K. Sahu, "Normalization: a preprocessing stage," 2015, https://arxiv.org/abs/1503.06462#:~:text=As%20we%20know%20that%20the,becomes%20used%20for%20further%20stage.

[27] S. Weisberg, *Applied Linear Regression*, John Wiley & Sons, Hoboken, NJ, USA, 2005.

[28] L. Guo, Y. Lei, N. Li, T. Yan, and N. Li, "Machinery health indicator construction based on convolutional neural networks considering trend burr," *Neurocomputing*, vol. 292, pp. 142–150, 2018.

[29] H. Shao, H. Jiang, H. Zhang, and T. Liang, "Electric locomotive bearing fault diagnosis using a novel convolutional deep belief network," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 3, pp. 2727–2736, 2018.

[30] M. Robnik-Šikonja and I. Kononenko, "Theoretical and empirical analysis of ReliefF and RReliefF," *Machine Learning*, vol. 53, no. 1/2, pp. 23–69, 2003.

[31] M. Navas and C. Ordonez, "Efficient computation of PCA with SVD in SQL," in *Proceedings of the 2nd Workshop on Data Mining Using Matrices and Tensors*, pp. 1–10, Paris, France, September 2009.

[32] T. D. Lagerlund, F. W. Sharbrough, and N. E. Busacker, "Spatial filtering of multichannel electroencephalographic recordings through principal component analysis by singular value decomposition," *Journal of Clinical Neurophysiology*, vol. 14, no. 1, pp. 73–82, 1997.

[33] S. Ravi and H. Larochelle, *Optimization as a Model for Few-Shot Learning*, OpenReview.net, 2016.

[34] A. K. Mahamad, S. Saon, and T. Hiyama, "Predicting remaining useful life of rotating machinery based artificial neural network," *Computers & Mathematics with Applications*, vol. 60, no. 4, pp. 1078–1087, 2010.

[35] Z. Shao, M. J. Er, and N. Wang, "An efficient leave-one-out cross-validation-based extreme learning machine (ELOO-ELM) with minimal user intervention," *IEEE Transactions on Cybernetics*, vol. 46, no. 8, pp. 1939–1951, 2016.

[36] F. Yang, M. S. Habibullah, T. Zhang, Z. Xu, P. Lim, and S. Nadarajan, "Health index-based prognostics for remaining useful life predictions in electrical machines," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 4, pp. 2633–2644, 2016.

[37] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *Proceedings of the 2008 International Conference on Prognostics and Health Management*, pp. 1–9, Denver, CO, USA, December 2008.

[38] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in *Proceedings of the 2017 IEEE International Conference on Prognostics and Health Management (ICPHM)*, pp. 88–95, IEEE, Dallas, TX, USA, August 2017.

[39] V. Mathew, T. Toby, V. Singh, B. M. Rao, and M. G. Kumar, "Prediction of Remaining Useful Lifetime (RUL) of turbofan engine using machine learning," in *Proceedings of the 2017 IEEE International Conference on Circuits and Systems (ICCS)*, pp. 306–311, IEEE, Thiruvananthapuram, Kerala, India, December 2017.

[40] C. Yu and W. Yao, "Robust linear regression: a review and comparison," *Communications in Statistics - Simulation and Computation*, vol. 46, no. 8, pp. 6261–6282, 2017.

[41] S.-X. Du and T.-J. Wu, "Support vector machines for regression," *Acta Simulata Systematica Sinica*, vol. 11, no. 022, 2003.

[42] B. Schölkopf, A. J. Smola, and F. Bach, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and beyond*, MIT press, Cambridge, MS, USA, 2002.