

Fault detection and diagnosis in refrigeration systems using machine learning algorithms

Soltani, Zahra; Sørensen, Kresten Kjær; Leth, John; Bendtsen, Jan Dimon

Published in:
International Journal of Refrigeration

DOI (link to publication from Publisher):
[10.1016/j.ijrefrig.2022.08.008](https://doi.org/10.1016/j.ijrefrig.2022.08.008)

Creative Commons License
CC BY 4.0

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

[Link to publication from Aalborg University](#)

Citation for published version (APA):

Soltani, Z., Sørensen, K. K., Leth, J., & Bendtsen, J. D. (2022). Fault detection and diagnosis in refrigeration systems using machine learning algorithms. *International Journal of Refrigeration*, 144, 34-45.
<https://doi.org/10.1016/j.ijrefrig.2022.08.008>

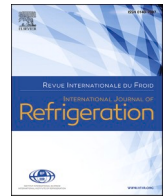
General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal -

Take down policy

If you believe that this document breaches copyright please contact us at vbn@aub.aau.dk providing details, and we will remove access to the work immediately and investigate your claim.



Fault detection and diagnosis in refrigeration systems using machine learning algorithms

Détection et diagnostic des défaillances dans les systèmes frigorifiques à l'aide d'algorithmes d'apprentissage machine

Zahra Soltani ^{*,a}, Kresten Kjær Sørensen ^b, John Leth ^a, Jan Dimon Bendtsen ^a

^a Department of electronic systems, Aalborg University, Fredrik Bajers vej 7, Aalborg, 9220, Denmark

^b Department of Transport, Bitzer Electronics, Kærvej 77, Sønderborg, 6400, Denmark

ARTICLE INFO

Keywords:

Refrigeration
Fault detection
Machine learning
Dimensionality reduction
Sensor

Mots clés:

Froid [artificiel]
Détection des défaillances
Apprentissage automatique
Réduction de la dimensionalité
Capteur

ABSTRACT

The functionality of industrial refrigeration systems is important for environment-friendly companies and organizations, since faulty systems can impact human health by lowering food quality, cause pollution, and even lead to increased global warming. Therefore, in this industry, there is a high demand among manufacturers for early and automatic fault diagnosis. In this paper, different machine learning classifiers are tested to find the best solution for diagnosing twenty faults possibly encountered in such systems. All sensor faults and some relevant component faults are simulated in a high fidelity Matlab/Simscape model of the system, which has previously been used for controller development and verification. In this work, Convolutional Neural Networks, Support Vector Machines (SVM), Principal Components Analysis-SVM, Linear Discriminant Analysis-SVM, and Linear Discriminant Analysis classifiers are compared. The results indicate that the fault detection reliability of the algorithms highly depends on how well the training data covers the operation regime. Furthermore, it is found that a well-trained SVM can simultaneously classify twenty types of fault with 95% accuracy when the verification data is taken from different system configurations.

1. Introduction

Machine Learning (ML) is a common term for many processing methods used for data-driven tasks. The main intention of ML is to enable computers to learn, predict, or decide on an unseen data without human assistance (Saravanan and Sujatha, 2018). In the 2010s, rapid development of processors, IoT, and an increasing amount of generated data paved the way for large improvements in ML capabilities. Thus, the popularity of ML increased exponentially in many industries. Machine learning is used in various contexts, such as computer vision, text classification, fault detection, language processing, image recognition, and so forth.

The idea of using ML for fault detection and diagnosis dates back to the 1980s where the existing ML methods were not as efficient as specialized experts. However, the technologies have been improved, and as of today, the availability of powerful programming tools and

algorithms for self-learning allow computers to make strategic decisions and even diagnose new events (Gaughlitz, 2019).

In particular, ML-based methods have been studied for fault detection and diagnosis (FDD) in different fields with promising results. For instance, ML is used for fault detection in brushless synchronous generators in Rahnama et al. (2019), in water distribution network (Quiñones-Grueiro et al., 2021), in age intelligence systems (Liu et al., 2021), and in high-temperature super conducting DC power cables (Choi et al., 2021). In Hajji et al. (2021), several supervised ML algorithms are compared for FDD in photovoltaic systems. In Hajji et al. (2021), data from non-faulty condition and five different faulty conditions are used both for training and test; and the results confirm that supervised learning algorithms can be used for fault detection and ease the FDD procedure. Moreover, machine learning models are compared for sensor fault detection in Sana Ullah et al. (2021), in which five types of sensor faults are emulated, namely, drift, bias, precision degradation, spike, and stuck faults.

* Corresponding author.

E-mail addresses: zns@es.aau.dk, zahra.soltani@bitzerdk.com (Z. Soltani), kresten.soerensen@bitzerdk.com (K.K. Sørensen), jjl@es.aau.dk (J. Leth), dimon@es.aau.dk (J.D. Bendtsen).

<https://doi.org/10.1016/j.ijrefrig.2022.08.008>

Received 23 December 2021; Received in revised form 12 August 2022; Accepted 17 August 2022

Available online 19 August 2022

0140-7007/© 2022 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

Nomenclature*Latin symbols*

f	activation function
n	number of samples
c	number of feature vectors
W	weight matrix in CNN
x	input sub-matrix in CNN
b	bias
S	number of the neurons in the last layer
y	output neuron
\hat{y}	predicted output
Y	output of each hidden layer
L	loss function
S	variance
V	matrix of all eigenvectors
v	eigenvectors
r	transformed sample
w	weight vector
H	hyper plane
x	input vector in SVM and LDA
C	regularization parameter for ζ
T	temperature ($^{\circ}\text{C}$)
P	pressure (Pa)
V	speed (percentage of full speed)
v_{exp}	expansion valve opening degree (%)
p	power (W)
X	input

Greek symbols

ρ	density in suction line ($\frac{\text{kg}}{\text{m}^3}$)
ν	number of classes
χ	feature vector
μ	mean
Ω	transformation matrix
λ	eigenvalue
α	number of selected eigenvectors
ζ	slack variable
Φ	GRBF function
γ	GRBF multiplier

Subscripts and superscripts

j	j_{th} sample
B	between-class
s	within-class
φ	φ_{th} vector
κ	κ_{th} class
$room$	cooling room
amb	ambient
suc	suction gas
dis	discharge
ret	returned air
sup	supplied air
0	evaporation
FC	frequency converter
C	condensing
sh	superheat ($^{\circ}\text{K}$)
cpr	compressor
$evap$	evaporator fan
$cond$	condenser fan
$real$	value before applying the fault
$fault$	value after applying the fault
$offset$	value for fault emulation

Abbreviations

GRBF	Gaussian Radial Basis Function
MLP	Multi Layer Perceptron
OVO	One Versus One
OVR	One Versus Rest
KDA	Kernel Discriminant Analysis
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis
SVM	Support Vector Machine
CNN	Convolutional Neural Network
FDD	Fault Detection and Diagnosis
ML	Machine Learning
RS	Refrigeration Systems
DC	Direct current
IoT	internet of things

For fault detection in office building systems, various data mining methods, in particular, Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Kernelized Discriminant Analysis (KDA), semi-supervised LDA, and semi-supervised KDA have been compared in Shioya et al. (2015). In Choudhary et al. (2021), different component faults in a rotating machine are classified using a Convolutional Neural Network (CNN) algorithm. According to Lo et al. (2019), in many industrial applications, good system models are difficult or even impossible to obtain due to the system's complexity or large numbers of configurations involved in the production process. The refrigeration industry is not an exception, as the system configuration varies based on different owners' demands. Hence, model based FDD is often sensitive to model parameters in such a way that small changes in the system may lead to a poor fault detection response. In such cases, ML can be a viable approach to handling unseen situations when well trained.

In Soltani et al. (2020), a CNN model is used for evaporator fan fault detection in supermarket refrigeration systems. The same system configuration and information are used in Soltani et al. (2021) to classify the same fault and investigate the robustness of the fault detection model. However, instead of CNN, shallow learning Support Vector Machines SVM and PCA-SVM classifiers are used. In Han et al. (2010), SVM and PCA-SVM are studied for the detection of 8 type of faults in a

simulated vapour-compression refrigeration system in which PCA-SVM achieved a better result compared to SVM and back-propagation neural network.

In the refrigeration industry, good performance of a fault detection algorithm can be defined as high classification accuracy, low computation time, and low false positive rate. High classification accuracy ensures an accurate fault description for the technicians for quick troubleshooting, while low computation time is important because it lowers the detection time and the hardware cost. A low false positive rate increases the reliability of the fault detection model and results in lower expenses regarding service call rate. Therefore, it is essential to evaluate the FDD algorithms based on these factors.

Because of increasing usage of digitalization in refrigeration systems (RS), many companies aim for improving existing FDD performance by utilising various data. As mentioned above, FDD algorithms perform satisfactorily in many other applications; thus, data driven FDD algorithms are selected and evaluated in this work. That is, we evaluate and optimize various FDD algorithms for the purpose of selecting the best classifier for use in RS industry applications.

The main contributions of this study is summarized below:

- A deep learning and several shallow learning classifiers are proposed for detecting and diagnosing twenty types of faults in RS.
- Importance of training data qualification regarding data variation and features selection is illustrated.
- All of the proposed classifiers are compared regarding classification accuracy, computation time and false positive rate.
- The best approach from an industrial perspective is proposed to detect a faulty system and localize the fault.

In this study, all sensor faults and some component faults are simulated using a high fidelity RS model. The model is already in use at Bitzer Electronics to develop and verify control algorithms. Notice that we will restrict our attention to steady state operating conditions, which are commonly encountered in industrial application such as reefer containers, cold storage houses and so on. It is acknowledged that transient operation is important in many applications as well, e.g., in supermarket refrigeration systems. However, transient behavior presents its own set of unique challenges, and is considered out of scope of this work.

The faults include positive and negative offsets in sensors as well as specific component faults; the faults are detailed in section 2. Three classifiers, namely CNN, SVM, and LDA, are compared to diagnose every selected fault. For pre-processing of the input data LDA and PCA are compared.

The results indicate that the SVM classifier is the superior method, being able to diagnose all classes with 100% classification accuracy except non-faulty and malfunctioning of expansion valve conditions which are diagnosed with 98% and 96% classification accuracy, respectively. The LDA and LDA-SVM classifiers are capable of detecting the faulty condition with 100% classification accuracy. However, these models have poor performance regarding robustness as a significant drop in classification accuracy is observed. Finally, CNN and PCA-SVM show a general lack in performance.

The remainder of this paper are structured as follows. First, refrigeration systems background and specification, as well as data acquisition and its specification, are introduced in section 2. Then, in section 3, the mathematical approaches of the classifiers mentioned above are explained. Afterwards, the specification of each model and the result of the classification is presented in section 4. Finally, the work is concluded in section 5.

2. Background

In general, RS are used to cool down the goods inside of an insulated

room, which is called a cold room, by transferring the heat to the environment. Fig. 1 illustrates a RS in which the refrigerant runs through the pipes. In each refrigeration cycle, heat is absorbed and dissipated. The compressor receives low pressure, low temperature refrigerant gas and releases high pressure, high temperature gas to the inlet of the condenser. The condenser is responsible for dissipating the refrigerant heat to the ambient environment, and finally gives out liquid refrigerant at high pressure while the temperature decreases. Afterwards, an expansion valve decreases the pressure of the refrigerant. Low pressure, low temperature refrigerant enters the evaporator pipes in order to absorb the heat from the cold room environment. Thus, the refrigerant changes phase from liquid to gas before reaching the compressor.

Defective components or sensors in RS lead to high power consumption, air pollution, wear and tear of the components, and/or food waste. RS have the best efficiency when everything is nominal. Thus, when faults occur, the system might deviate from the peak efficiency point. By some of the fault, the system runs outside of its permitted envelope, some of the faults lead to wear and tear of the components due to high temperature, too little lubrication, and too high pressure on the components. Late fault detection may cause the temperature of the refrigerated goods to exceed the permitted limits. Therefore, early fault detection in RS ensures maintaining the required quality of refrigerated goods such as food products or medicine, and preventing excessive maintenance and spoilage cost.

The high fidelity model used by Bitzer Electronics is presented in Fig. 2. In this model, a two-stage semi-hermetic reciprocating compressor is simulated with operating speed in the range 25–87 Hz. Here, compressor cooling capacity (V_{cpr}) is defined as compressor operating speed in percentage. Therefore, compressor speed under 25 Hz and full speed operation of 87 Hz are defined as 0% and 100% compressor cooling capacity, respectively. The refrigerant type is R134a, and an electrical expansion valve is simulated. Maximum cooling capacity of the cold room is 17 kW at 10 °C ambient temperature (T_{amb}) and 5 °C cold room temperature (T_{room}). The controller is designed so as it controls over opening degree of expansion valve (v_{exp}) using superheat temperature (T_{sh}) measurements as an input. T_{sh} is the difference between the refrigerant evaporation temperature (T_0) and suction gas temperature (T_{suc}). In addition, V_{cpr} , evaporator fan speed (V_{evap}), condenser fan speed (V_{cond}), are controlled using the mentioned controller inputs in Fig. 2. In this paper the supply temperature (T_{sup}) is the same as cold room temperature (T_{room}) and used as set point in the simulation model. Thus, set point is the temperature of the air after transferring heat to the refrigerant.

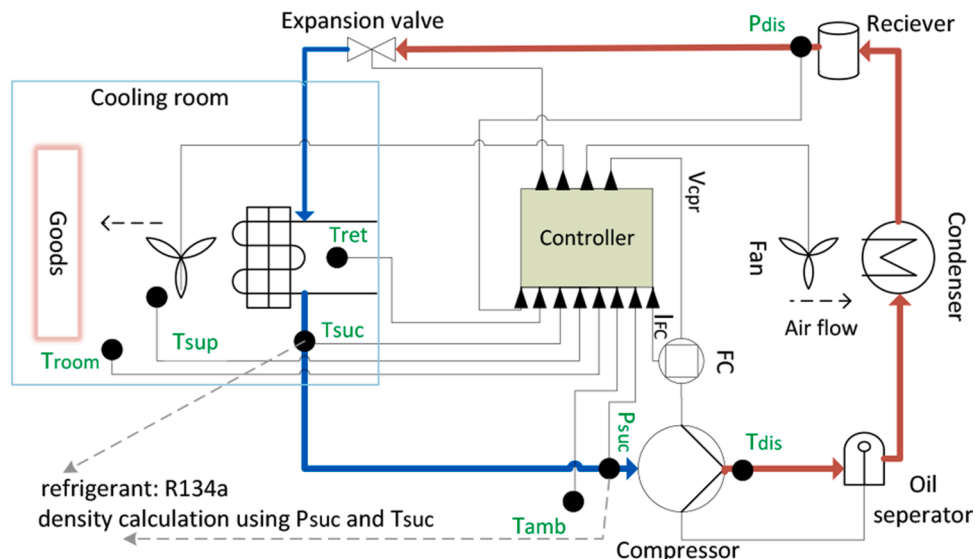


Fig. 1. Schematic of a refrigeration system.

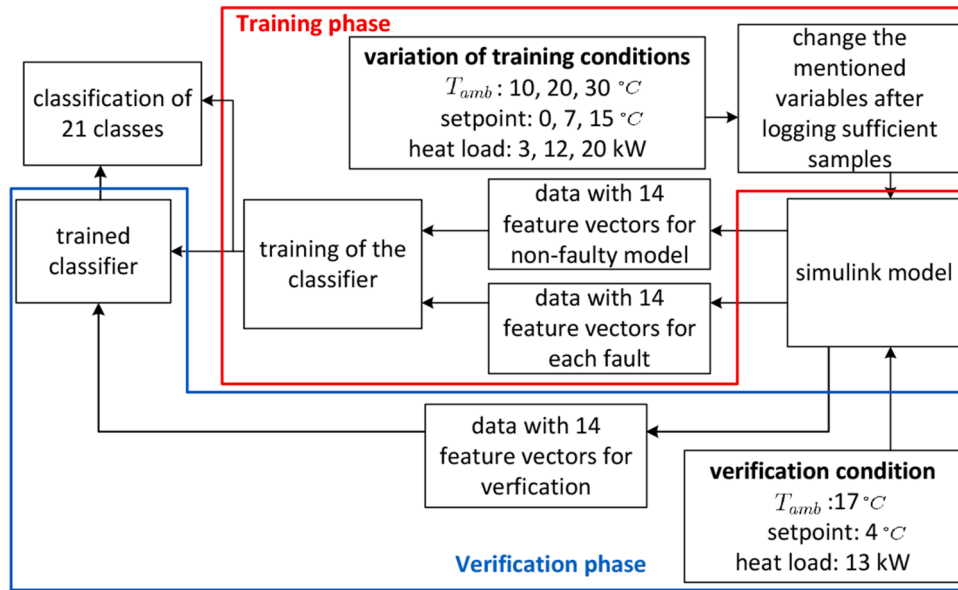


Fig. 3. An overview of data collection and ML setup. The red section indicates the training phase where data is collected and used for training of the ML model. The blue section shows verification data specification and classification.

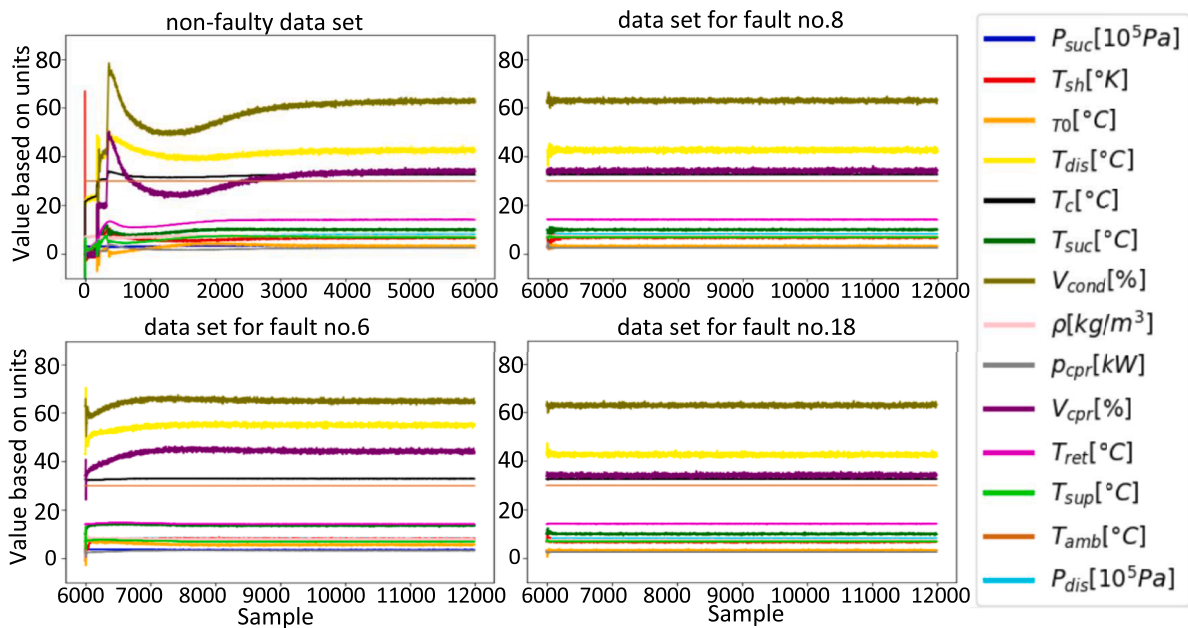


Fig. 4. Four examples of data set from different classes which have the same system configuration. The set point to $T_{sup} = T_{room}$ is set to 7 °C, heat load in the cooling room is 13 kW at the ambient temperature of 25 °C.

more than the commanded value from the controller indicating fault 8, and a blocked expansion valve that reacts 20% less than the commanded value. During data acquisition, the model is run in non-faulty condition until sample 6000. Then, each fault is introduced from sample 6001 to 12000 as seen in Fig. 4. It is observed that in some cases, such as fault 8, the data looks very similar to some of the other faulty or non-faulty data. Changes in condensing temperature (T_c) is compensated by condenser fan work, because there is a feedback control on condenser fan to keep constant pressure relative to T_{amb} and the controller controls V_{cpr} based on T_{sup} . Thus, it is hard to observe any visual changes in the data characteristics during steady state response. However, in some other cases, the fault affects the controller response immediately, and the changes can be observed in the data easily. For example, fault 6, which is shown in Fig. 4, clearly gives rise to variations in T_{dis} , and V_{cpr} . The compressor

works based on the controller command. In the case of fault 6, ρ and/or P_{suc} which are fed into the controller are measurements of the faulty sensor. Therefore, the compressor behavior is based on the faulty sensor measurement. However, as the real P_{suc} is less than required, it causes drop in mass flow rate. In Fig. 2, the P_{suc} offset is applied only to the sensor reading. The controller controls both expansion valve opening degree and compressor speed to reach a desired pressure, and when the reading is positively offset the controller must lower the actual suction pressure to reach the desired reading.

2.2. Data specification and dimensionality reduction

The idea behind dimensionality reduction techniques is to remove dependent and redundant features from original data by projecting data

to a lower-dimensional space, which holds only essential information. These approaches deal with noisy data and reduce the computation load for classification purposes (Soltani et al., 2021). In this work, the input data has 14 feature vectors or dimensions, including sensor signals, and some of the variables from RS controller, including superheat temperature, saturated evaporation temperature, compressor cooling capacity/speed, condenser fan speed, and vapour density. Statistical approaches such as PCA and LDA are used to reduce the input data dimensions before passing them through the classifiers. In this paper, all transient part of the data is removed, both for training and validation data. The 14-dimensional data is reduced to 2-dimensional data using PCA as the input to the SVM. LDA is also used for dimensionality reduction and transfers the data into a 6-dimensional data set before sending the data into the SVM classifier. Moreover, CNN and SVM are also applied to the 14-dimensional data set. For SVM and LDA classification, each class of data contains 1200 samples, and for the CNN classifier, 18000 samples with a sample rate of 1 Hz. Remark that LDA and SVM are shallow learning neural networks which, as an advantage, do not require as many samples as CNN. Too many samples result in too high computation load and low classification accuracy. As described in 2.1, the training data of each class contains various RS operating conditions. These varieties prevent overfitting and increase the model's capability for the classification of unseen operating conditions.

3. Methods

SVM, LDA, and CNN are all supervised learning methods which are sub-fields of the linear classifiers (Saravanan and Sujatha, 2018). Supervised ML classifiers categorize a new data set using a pre-trained model. Thus, the model is first trained using input data and defined labels.

CNN is a deep learning classifier commonly used for image processing purposes. A CNN is comprised of two phases of feature extraction and classification. The input data consists of feature vectors $\chi_\varphi \in \mathbb{R}^{n \times 1}$, $\varphi = 1, \dots, c$ which are gathered in data matrices $X_\kappa \in \mathbb{R}^{n \times c}$, one for each class κ , $\kappa = 1, \dots, \nu$. The numbers $n = n_\kappa$ and c quantify the number of samples in each class and the number of features, respectively. For convenience, it is assumed that all the data matrices have the same dimensions, although this is not a strict requirement.

In the feature extraction phase, the CNN makes use of so-called *neurons* which take data matrices X_κ as input and return (neuron) output $y_\kappa^k \in \mathbb{R}^{n' \times c'}$, $k = 1, \dots, S$, where S is the number of neurons (see Fig. 5). Each neuron has a *weight matrix* $W^k \in \mathbb{R}^{n \times c}$ and a bias matrix $b^k \in \mathbb{R}^{n \times c}$ associated with it. For each κ , X_κ is partitioned into $n'c'$ (possibly overlapping) submatrices $(x_\kappa)_{ij} \in \mathbb{R}^{n \times c}$, $i = 1, \dots, n'$, $j = 1, \dots, c'$. Then the neuron output y_κ^k is a matrix whose entries are defined as:

$$(y_\kappa^k)_{ij} = f\left(\mathbf{1}^T (W^k \odot (x_\kappa)_{ij} - b^k) \mathbf{1}\right) \quad (1)$$

where \odot denote element-wise multiplication of matrices, $\mathbf{1}$ denotes a

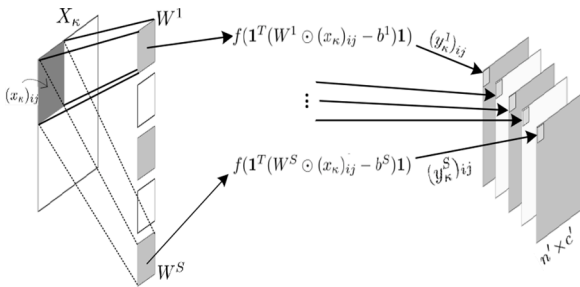


Fig. 5. A feature extraction layer of CNN, a sub-matrix $(x_\kappa)_{ij}$ is convolved with each weight matrix W^k , resulting in a number of matrices as the output of the layer.

vector of ones, and $f: \mathbb{R} \rightarrow \mathbb{R}$ is an activation function.

It is noted that the size $n' \times c'$ and number S of W^k 's are hyper-parameters, which can be tuned during the design of the CNN model to optimally filter different information of the input.

As illustrated in Fig. 6, the output of the feature extraction phase contains the essential information of the input data. This output is then vectorized as $Y^0 = \text{col}[y_\kappa^k] \in \mathbb{R}^{n' \times c' S \nu \times 1}$ before being used as input to the classification phase, which is a fully connected Multi-layer Perceptron, see Geidarov (2017), and Bishop (2006) with N_{MLP} fully connected layers. The output vector of each MLP layer $Y^l \in \mathbb{R}^{n_l \times 1}$ is computed recursively as

$$Y^l = \hat{f}(W_l Y^{l-1} + b_l), \quad l = 1, \dots, N_{\text{MLP}} \quad (2)$$

where $W_l \in \mathbb{R}^{n_l \times n_{l-1}}$ is a layer weight matrix, $b_l \in \mathbb{R}^{n_l \times 1}$ is a bias vector, $\hat{f}: \mathbb{R}^{n_l} \rightarrow \mathbb{R}^{n_l}$ is the l 'th layer's neuron activation function, and $n_{N_{\text{MLP}}} = \nu$.

The output $\hat{y} \in \mathbb{R}^\nu$ of the CNN is generated by the so-called *Softmax* activation function where the κ th coordinate of \hat{y} is given by:

$$\hat{y}_\kappa = \frac{\exp(Y_\kappa^{N_{\text{MLP}}})}{\sum_{j=1}^\nu \exp(Y_j^{N_{\text{MLP}}})} \quad (3)$$

with $Y_\kappa^{N_{\text{MLP}}}$ being the κ th coordinate of $Y^{N_{\text{MLP}}}$.

Here, it is noted that since the CNN output is normalized ($\sum_{\kappa=1}^\nu \hat{y}_\kappa = 1$), \hat{y}_κ may be considered as the probability of a new input X belonging to class κ .

During the training process, the estimation of the classes are compared with the true labels y_κ using a loss function. The loss function is also a hyper parameter that needs to be determined for the model; a common loss function is *cross entropy*:

$$L = - \sum_{\kappa=1}^\nu y_\kappa \ln(\hat{y}_\kappa). \quad (4)$$

The training process aims at adjusting the weights in such a way that better prediction of the correct class is achieved. In other words, the minimum loss is obtained.

Minimization of the loss function can be done using different optimization techniques; the most common being *Backpropagation* (Bishop, 2006), which is a variant of gradient descent. Once the weights have been adjusted to yield the optimal output for a validation data set, this model can be used to classify unlabeled, new data.

3.1. LDA classifier

Linear discriminant analysis (LDA) can be used both for dimensionality reduction and classification purposes. In LDA, as it is depicted in Fig. 7, linear separation of classes is done after projecting data onto another space. LDA seeks a large separation between transformed classes compared to the original one after the dimension of the transformed data is reduced. A transformation matrix is obtained by use of the between-classes variance and the variance within each class (Bishop, 2006).

The variance between classes $S_B \in \mathbb{R}^{c \times c}$ is calculated as follows:

$$S_B = \sum_{\kappa=1}^\nu (\mu_\kappa - \mu)(\mu_\kappa - \mu)^T \quad (5)$$

where $\mu_\kappa \in \mathbb{R}^{1 \times c}$ is the mean value of class κ , and $\mu \in \mathbb{R}^{1 \times c}$ is mean of all μ_κ . Afterwards, the within-class variance $S_s \in \mathbb{R}^{c \times c}$ is calculated by

$$S_s = \sum_{\kappa=1}^\nu \sum_{j=1}^n ((X_\kappa)_j - \mu_\kappa)^T ((X_\kappa)_j - \mu_\kappa) \quad (6)$$

where $(X_\kappa)_j$ is the j th row (or sample) in X_κ .

S_s and S_B are used to find the transformation matrix $\Omega \in \mathbb{R}^{c \times c}$ defined

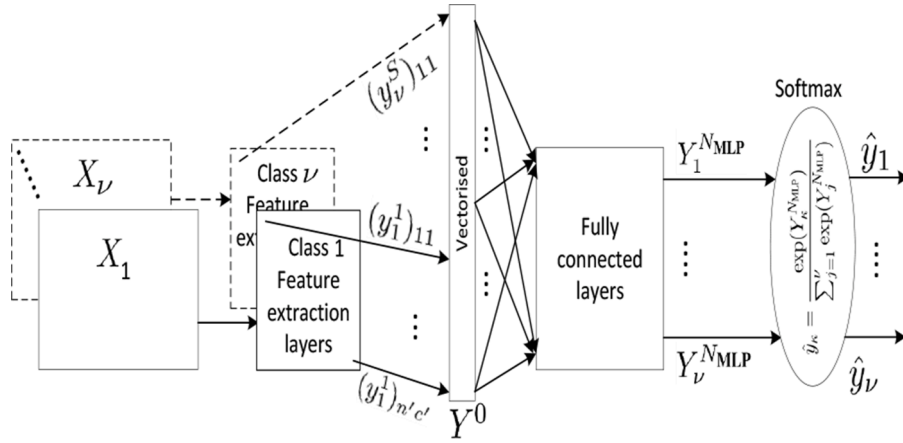
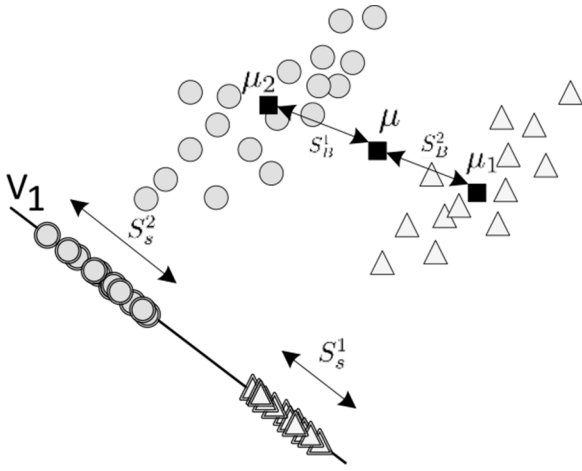
Fig. 6. General CNN structure for ν classes.

Fig. 7. LDA visualisation for dimensionality reduction from two to one-dimensional space.

as

$$\Omega = S_s^{-1} S_B \quad (7)$$

Afterwards, this transformation matrix is used to generate data in another space in which the classes are linearly separable. In order to reduce the dimensions of the data in the new space, eigenvectors and eigenvalues of Ω are obtained. The eigenvectors with higher eigenvalues carry more information of the data distribution (Tharwat et al., 2017). Order the eigenvalues of Ω in decreasing order $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_c$, choose the first $\alpha \leq c$ corresponding eigenvectors and organize them in a new matrix $V = [v_1 v_2 \dots v_\alpha] \in \mathbb{R}^{c \times \alpha}$. The lower-dimensional samples $r_j \in \mathbb{R}^{1 \times \alpha}$, $j = 1, \dots, n$ in class κ are then the rows of the matrix product $X_\kappa V$.

3.2. SVM classifier

Support vector machine (SVM) is a supervised machine learning method and linear classifier which classifies data into two or more classes. In the sequel we focus on the case of two classes.

Consider the two classes X_κ , $\kappa = 1, 2$ containing the samples as rows and set $y_j = -1$ or $y_j = 1$ if $x_j \in \mathbb{R}^{1 \times c}$ is a row in X_1 or a row in X_2 , respectively. Assume that the two classes are linearly separable, that is, the samples of each class can be separated by a (linear) hyper plane. Then there exists a hyper plane

$$H = \{x \in \mathbb{R}^c \mid xw^T + b = 0\}$$

with weight $w \in \mathbb{R}^{1 \times c}$ and bias $b \in \mathbb{R}$ such that $1/\|w\|$ is the distance from H to the nearest sample in class 1 and class 2. These nearest samples are usually called support vectors (see Fig. 8). Moreover, w and b may be found as the solution to the optimisation problem

$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (8a)$$

$$\text{s.t.} \quad y_j(x_j w^T + b) \geq 1, \quad j = 1, \dots, n \quad (8b)$$

The optimal (or hard) margin (that is, $1/\|w\|$ with w the solution to (8)) may not always lead to the best result when feeding unseen data to the model. The optimal margin might result in overfitting or margin violations. In particular, outliers can fall into the wrong class and be misclassified (Murty and Raghava, 2016). In practice, the classifier is allowed to do small misclassifications during the training, which is called *soft margin* (shown in Fig. 8). To do so, a slack variable ζ is added to the optimization problems:

$$\min_{w,b,\zeta} \frac{1}{2} \|w\|^2 + C \sum_{j=1}^n \zeta_j \quad (9a)$$

$$\text{s.t.} \quad y_j(x_j w^T + b) \geq 1 - \zeta_j \quad (9b)$$

$$\zeta_j \geq 0, \quad j = 1, \dots, n \quad (9c)$$

where C is a hyper parameter that determines the size of the allowed misclassification. The size of the parameter C is tuned by software such that the classification accuracy of unseen data is high.

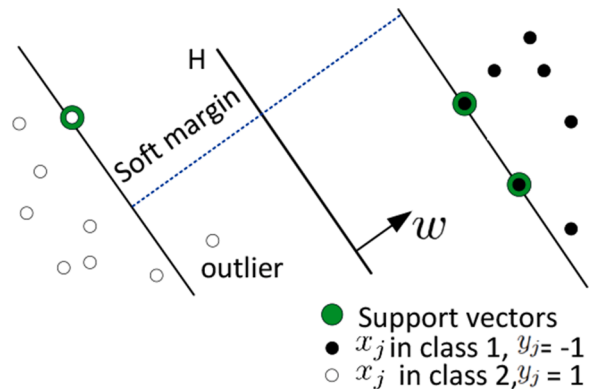


Fig. 8. Finding a classification hyperplane.

In many classification problems a linear classification is not possible. The kernel trick is a method for dealing with this case. It yields a transformation of the input space, that is the space which the samples belong to, into another higher dimensional space, in which the samples are linearly separable (Murty and Raghava, 2016). This new space is typically called the feature space. The kernel trick relies on the use of kernel function. In this work we consider a special class of kernel function, called the Gaussian Radial Basis Functions (GRBF) given by

$$\Phi(x, x') = \exp(-\gamma \|x - x'\|^2) \quad (10)$$

The hyper parameter $\gamma > 0$ determines the influence of each sample on selecting the hyper plane during training. It should be noted that choosing γ too big results in overfitting and choosing γ too small leads to under-fitting of the model (Bishop, 2006).

3.3. Multi-class classification

In the case of more than two classes, the problem can be solved using two approaches. The first one is to consider each class against the rest of the classes and is called One Versus the Rest (OVR). For the model training using OVR, one binary classifier is used for each class against all the other classes as the second category. Therefore, for a data set including ν classes, ν binary classifiers are created. For unseen data classification, each classifier is tested to determine to which class the new sample belongs. However, in many cases, the result of OVR is inconsistent as the sample can belong to either more than one class or none of them, illustrated as the gray stars in Fig. 9. Since, OVR picks one class against all other classes together, the number of samples in the corresponding class is typically a lot fewer than the rest of the classes. Therefore, the big difference between the number of samples often impacts the decision boundary (Bishop, 2006).

The second multi-class classification approach takes each class versus another and is called the One Versus One (OVO) approach. Thus, for each pair of classes, one classifier is trained. Finally, $\frac{\nu(\nu-1)}{2}$ classifiers determine each class boundaries as shown in Fig. 9.

The OVO approach is not as computationally effective as OVR due to using more classifiers. Moreover, the OVO approach has a tendency to overfit (Platt et al., 2000). However, in the end, a certain amount of trial and error is unavoidable in selecting a multi-class SVM classifier, as it depends on the input data and feature space.

4. Experiments

In this work, PCA and LDA are built in Python for dimensionality reduction purposes. It is advantageous to use lower-dimensional input data if it reduces the computation time of the classification and/or increases accuracy by removing redundant information in the data set such as noise, etc. This work tests and compares PCA-SVM and LDA-SVM models to the SVM classifier with full-dimensional data. The algorithms are built using the *scikit-learn* library in python which provides many efficient algorithms in ML, dimensionality reduction and classification. In Aurélien (2019), the ways of implementing aforementioned ML techniques in the *scikit-learn* library are described. In this

work, the label -1 is assigned to non-faulty data, while other labels are specified in Table 1. Moreover, the classifiers are fed with two sets of training data which are described in section 2, in order to evaluate the qualification of the training data.

4.1. Full-dimensional classifiers

The input data used for the SVM model includes $n = 1200$ samples of 14 feature vectors for each class. In addition, the input data contains samples from different system configurations. Each sample is labelled with one of the labels in Table 1. The SVM classifier performs OVO classification using $C = 1000$, and $\gamma = 0.01$ (see section 3.2); the hyperparameters were found by trial-and-error. The result of classification is represented in Fig. 10. *True labels* are the labels assigned to each class during the training phase, while *Predicted labels* refers to the prediction of the classifier during the training process. Thus, the diagonal values represent correct classifications. In this test, 250 samples with 1 Hz sample rate are selected for prediction.

The SVM result shows high classification accuracy for most of the classes, and there are no false positives. At 93% accuracy, the broken compressor with label 17 in Table 1 is the only fault that is misclassified.

As mentioned in section 3, LDA can be used both for dimensionality reduction and classification purposes. Here, LDA is used to classify all 21 classes of data while reducing the dimensions of the input data from 14 to 5. As shown in Fig. 10, the response of the LDA classifier is very similar to SVM classification, exhibiting 100% classification accuracy for most of the classes and no false positives. The only misclassification of about 3% is the broken compressor, which is mistaken for either P_{suc} sensor negative offset or broken evaporator fan.

CNN is a deep learning model and needs more samples compared to LDA or SVM. In the CNN model experiment, the data set for each class contains 12000 samples of all 14 feature vectors. The classification response of the training is represented in Fig. 11. The CNN classifier obtained a total accuracy of 94% and could classify most of the faults with 100% accuracy. The noticeable drawback is the false positive rate of 58%. The non-faulty condition was misclassified as classes with labels 8 and 18, which are both expansion valve malfunctions.

4.2. Reduced-dimension classifiers

In this part, PCA and LDA are used to reduce the input dimensionality. These approaches are investigated to see whether PCA or LDA can improve classification results. In addition, it is vital to study whether low dimensional inputs can reduce training computation time in the case of PCA and LDA.

After feeding data into PCA and transforming to the new space, it appears that the first two dimensions of the transformed data contain more than 80% of the variations in the new space, as seen in Fig. 12. Therefore, the first two principal components are used as the inputs to the SVM instead of 14-dimensional data. Fig. 13 shows the response of the PCA-SVM classifier with $C = 1000$, $\gamma = 0.01$, and OVO decision function.

The result of PCA-SVM shows misclassification of most of the classes. PCA causes classes to overlap as the most uncorrelated information is squeezed into the first two principal components. The result of PCA-SVM classification is not satisfactory for the multi-class classification even though it represents satisfactory results for binary classification in Soltani et al. (2021).

LDA is already used for classification, as shown in Fig. 13. However, it can also be used only for dimensionality reduction; then, the transformed lower dimensional data is used in a classification algorithm such as SVM. The first five eigenvectors corresponding to the first highest eigenvalues indicate that LDA reduces the input dimensions from eleven to five. A LDA-SVM classifier is built using $C = 1000$, $\gamma = 0.01$, and OVO decision function for the SVM part. The LDA-SVM classifier performs satisfactorily for many of the classes shown in Fig. 13. However, the

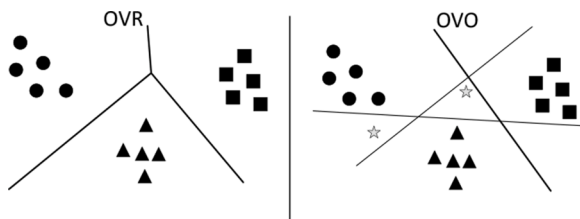


Fig. 9. multi-class data classification using OVR at the left and OVO at the right.

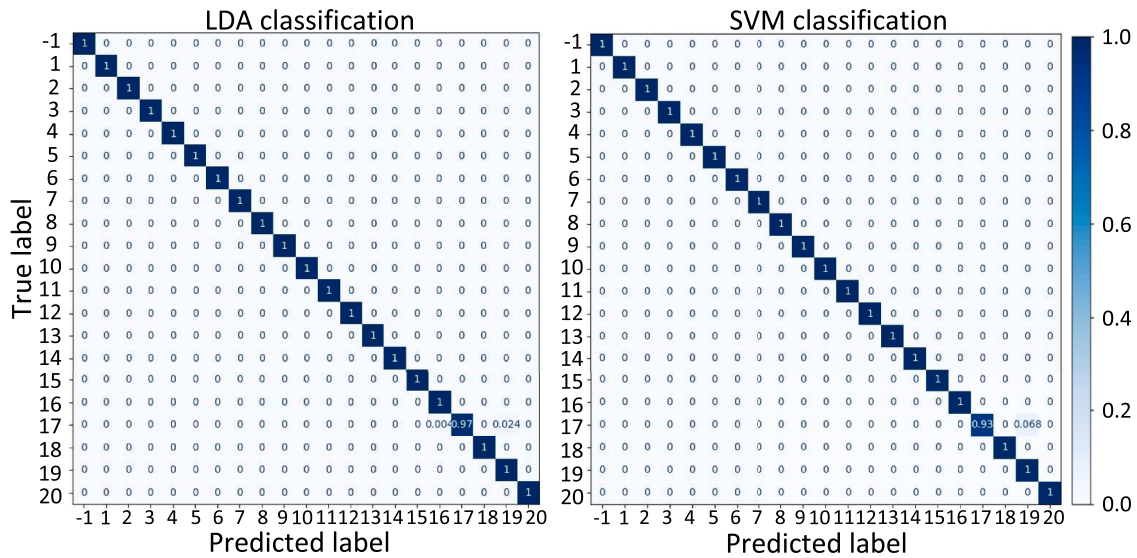


Fig. 10. LDA and SVM response for classification of 21 classes.

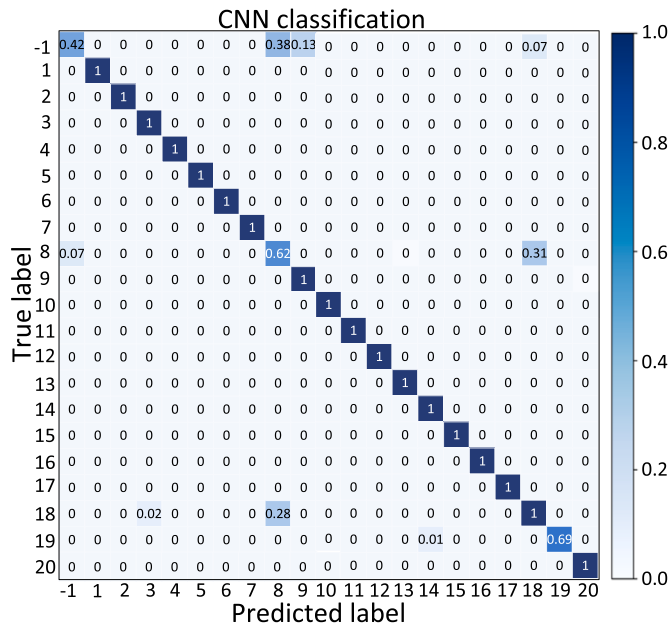


Fig. 11. Training response of the CNN classifier.

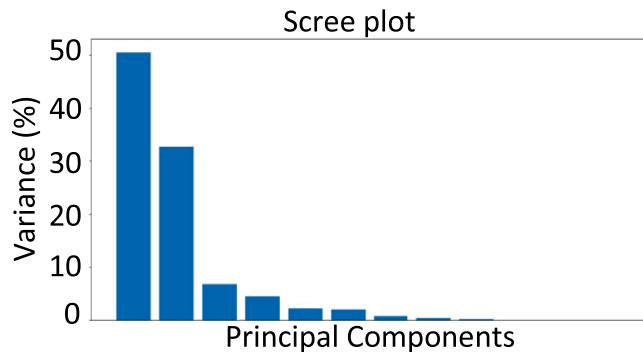


Fig. 12. The first two principle components contain the most variation among all 14 principle components.

model cannot easily distinguish the Non-faulty, loose expansion valve fault, and blocked expansion valve fault. Therefore, these three classes have low accuracy and a false positive rate of 18%, which is not a satisfactory result.

4.3. Model comparison

In this part, all five models described above are compared regarding input dimensions, total accuracy, false positive rate, training time, and prediction time. Training time is the computation time a model requires to be trained, including the time for dimensionality reduction in the cases where the input dimension is reduced. The prediction time is the computation time that a trained model uses to classify a test data set. Here, test data and training data are collected from the same system configuration and operation. The results in Table 2 represents the performance of the training process.

As seen in Table 2, SVM and LDA achieved the best results, with high accuracy and no false positives. However, the prediction time is relatively low for the LDA classifier compared to SVM, PCA-SVM, and LDA-SVM. On the other hand, the CNN classifier has the lowest prediction time, but the false positive rate is unacceptable. Therefore, LDA is found as the best model for multi-fault classification. Afterwards, more investigation is done on SVM, LDA and LDA-SVM, which perform better during the training phase.

4.4. The classifiers verification

In this part, the validation data is specified with a set point, heat load and ambient temperature which is different from what are used for the training set. In this data set, T_{set} is 4 °C, heat load is 13 kW and ambient temperature is 17 °C. Fig. 14 shows the response of SVM, LDA, and LDA-SVM classifier trained with the first training data set, with variations in set point and heat load. The overview of the results in Table 3 indicates that even though the classifiers did a good job during the training and test, they can not deal with the new data which are taken from a system in a new operating condition. Therefore, the classification results are not satisfactory, especially when looking at the false positive rate.

4.5. Effect of data variation

To deal with the challenge of misclassification of unseen data, a new training data set is fed into the same model, which contains more excitation by varying the RS operation around ambient temperature

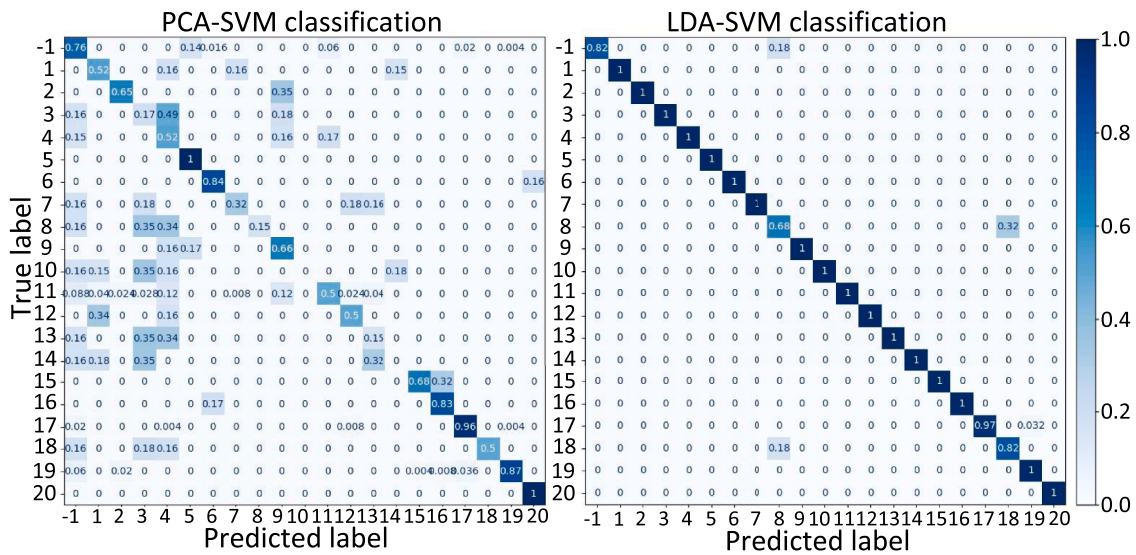


Fig. 13. PCA-SVM and LDA-SVM response for classifying 21 classes.

Table 2
Comparison of different classifiers.

model	dimensions	accuracy	false positive	training time	prediction time
SVM	14	99.6%	0%	1.1 s	1 s
LDA	14 to 5	99.8%	0%	3.2 s	0.3 s
CNN	14	94%	68%	112.5 s	0.1 s
PCA-SVM	14 to 2	55.4%	24%	7.2 s	5.6 s
LDA-SVM	14 to 5	96.6%	18%	1 s	1.1 s

Table 3
Robustness of classifiers against different operating conditions.

model	accuracy	false positive	prediction time
SVM	76%	92%	3.1 s
LDA	52%	100%	0.2 s
LDA-SVM	57%	100%	2.1 s

from 10 to 30 °C, set point from 0 to 12 °C, and heat load from 3 to 18 kW. In addition, to obtain better results, all 14 feature vectors are tested to see if one can affect misclassification. Thus, three features of input

data, namely, P_{suc} , compressor power consumption and density that were already used, are removed from the training and validation data set as they adversely affect the classification accuracy. The results are depicted in Fig. 15.

The overview of the results in Table 4 shows that the SVM classifier obtains more accurate results after training with more excited training data and removing the three mentioned feature vectors. However, for the LDA-SVM and LDA classifiers, the most accurate results are obtained when just the power consumption of the compressor and density are removed. Using this adjustment, the false positive percentage is improved a lot and SVM stands alone regarding the diagnosis of all faults simultaneously with high accuracy.

5. Conclusion

In this work, different classifiers are compared to diagnose twenty types of faults simultaneously and non-faulty condition in the industrial RS. The training data is taken from a simulation model which has been used in the development of system control in Bitzer. First, five classifiers, namely CNN, SVM, LDA, LDA-SVM, and PCA-SVM are compared. The training data contains information about the different systems operating through variation of set point and heat load of the cooling room. The test results show that CNN and PCA-SVM do not satisfactorily diagnose the

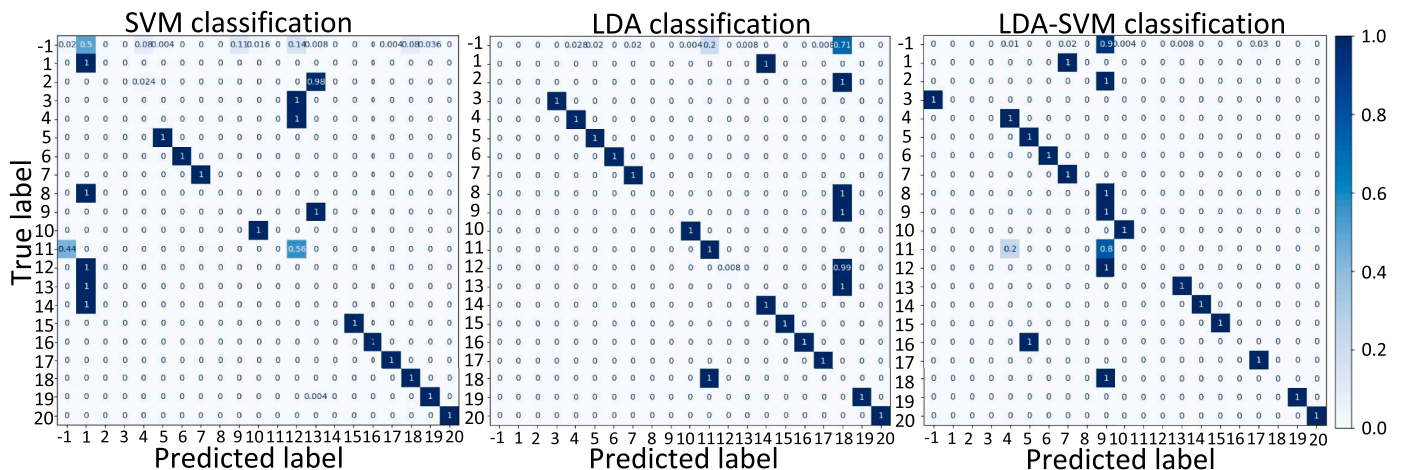


Fig. 14. Three classification responses of validation data with different system operating condition.

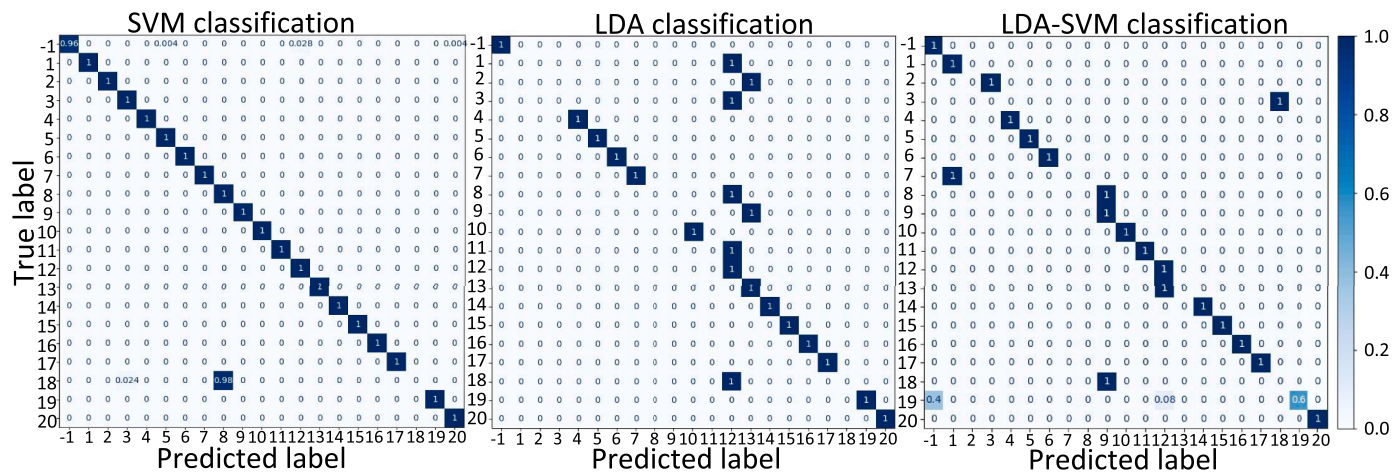


Fig. 15. Higher classification accuracy after training with new training data for all three classifiers comparing to Fig. 14.

Table 4

Robustness of classifiers after using qualified training data.

model	accuracy	false positive	prediction time
SVM	95%	4%	8.2 s
LDA	66%	0	0.2 s
LDA-SVM	69%	0%	9.1 s

faults. In addition, SVM, LDA, and LDA-SVM can not properly deal with the verification data set taken from different system operations than training data. On the other hand, training data with more excitation can help the classification results when the new training data contains the variations in ambient temperature, set point, and heat load. LDA and LDA-SVM are improved regarding false positive rate to 0%, but these classifiers do not provide the satisfactory classification results of the other classes.

It is seen that SVM has the highest classification accuracy of 95% with a 4% false positive rate. The only class which SVM does not diagnose is the blocked expansion valve, which is misclassified with the loose expansion valve. Therefore, even though this fault is misclassified, we can still trust that the malfunctioning valve needs to be checked by the technicians.

From an industrial point of view, it is very beneficial to have one classifier that can diagnose twenty one classes. Moreover, the classifiers considered in this work can be trained off-line. Off-line training may have two advantages. First, It is possible to train the classifier with simulation data and use the trained classifier for classification of real data to ensure that we do not train the classifier with the real data which are wrongly labeled. Second, the trained classifier would be computationally lighter compared if the training process were to be executed on embedded software as well. This is an advantage when the capacity of the processor of typical refrigeration systems is considered. The SVM model obtained the best classification accuracy at the algorithms tested. If a lower false positive percentage is considered, LDA can be used with a 0% false positive rate only for distinguishing the non-faulty class from the other faulty classes. Therefore, the system could benefit from having two classifiers, to make the diagnosis result more reliable. Before implementation of the classifier on real refrigeration systems, verification of the trained classifier by using real data from the field will be done in the future work.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Zahra Soltani reports financial support was provided by Innovation

Fund Denmark and Bitzer electronics A/S.

Acknowledgment

This work is funded by Innovation fund Denmark and Bitzer Electronics A/S. [fund number:9065-00010B]

References

- Aurélien, G., 2019. Hands-on Machine Learning with Scikit-Learn, Keras, and Tensorflow, 2nd edition. O'Reilly Media, Inc.
- Bishop, C.M., 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg.
- Choi, J.H., Park, C., Cheetham, P., Kim, C.H., Pamidi, S., Graber, L., 2021. Detection of series faults in high-temperature superconducting dc power cables using machine learning. IEEE transactions on applied superconductivity 31 (5), 1–9.
- Choudhary, A., Mian, T., Fatima, S., 2021. Convolutional neural network based bearing fault diagnosis of rotating machine using thermal images. Measurement : journal of the International Measurement Confederation 176.109196–
- Gauglitz, G., 2019. Artificial vs. human intelligence in analytics: Do computers outperform analytical chemists? Analytical and bioanalytical chemistry 411, 5631–5632.
- Geidarov, P.S., 2017. Clearly defined architectures of neural networks and multilayer perceptron. Optical memory & neural networks 26 (1), 62–76.
- Hajji, M., Harkat, M.-F., Kouadri, A., Abodayeh, K., Mansouri, M., Nounou, H., Nounou, M., 2021. Multivariate feature extraction based supervised machine learning for fault detection and diagnosis in photovoltaic systems. European Journal of Control 59, 313–321.
- Han, H., Cao, Z., Gu, B., Ren, N., 2010. Pca-svm-based automated fault detection and diagnosis (afdd) for vapor-compression refrigeration systems. HVAC&R research 16 (3), 295–313.
- Liu, T., Fu, Y., Xu, X., Yan, W., 2021. A cross-layer fault propagation analysis method for edge intelligence systems deployed with dnns. Journal of systems architecture 116.102057–
- Lo, N.G., Flaus, J.-M., Adrot, O., 2019. Review of machine learning approaches in fault diagnosis applied to iot systems. 2019 International Conference on Control, Automation and Diagnosis (ICCAD), pp. 1–6.
- Murty, M.N., Raghava, R., 2016. Support Vector Machines and Perceptrons: Learning, Optimization, Classification, and Application to Social Networks. Springer International Publishing AG.
- Platt, J. C., Cristianini, N., Shawe-Taylor, J., 2000. Large margin dags for multiclass classification, 547–553.
- Quiñones-Grueiro, M., Ares Milián, M., Sánchez Rivero, M., Silva Neto, A.J., Llanes-Santiago, O., 2021. Robust leak localization in water distribution networks using computational intelligence. Neurocomputing (Amsterdam) 438, 195–208.
- Rahnama, M., Vahedi, A., Alikhani, A.M., Montazeri, A., 2019. Machine-learning approach for fault detection in brushless synchronous generator using vibration signals. IET science, measurement & technology 13, 852–861.
- Sana Ullah, J., Young, D.L., In, S.K., 2021. A distributed sensor-fault detection and diagnosis framework using machine learning. Information Sciences 547, 777–796.
- Saravanan, R., Sujatha, P., 2018. A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification, pp. 945–949.
- Shioya, M., Masukawa, Y., Yairi, T., Yoshida, K., 2015. Energy fault detection in office building system by machine learning methods. ASHRAE transactions 121 (1).185–

- Soltani, Z., Soerensen, K.K., Leth, J., Bendtsen, J.D., 2020. Fault detection of supermarket refrigeration systems using convolutional neural network. IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society, pp. 231–238.
- Soltani, Z., Sørensen, K.K., Leth, J., Bendtsen, J.D., 2021. Robustness analysis of pca-svm model used for fault detection in supermarket refrigeration systems. 2021 International Conference on Electrical, Communication, and Computer Engineering (ICECCE). IEEE, pp. 1–6.
- Tharwat, A., Gaber, T., Ibrahim, A., Hassanien, A.E., 2017. Linear discriminant analysis: A detailed tutorial. AI Communications 30, 169–190.