A Systematic Semi-Supervised Self-adaptable Fault Diagnostics Approach in an Evolving Environment

Yang Hu[1], Piero Baraldi[1], Francesco Di Maio[1], Enrico Zio[1,2]

[1] *Politecnico di Milano, Department of Energy, Via La Masa, 20156 Milan, Italy*

[2]*Chair on Systems Science and the Energetic challenge, Fondation EDF, Centrale Supélec, Paris, France*

ABSTRACT

Fault diagnostic methods are challenged by their applications to industrial components operating in evolving environments of their working conditions. To overcome this problem, we propose a Systematic Semi-Supervised Self-adaptable Fault Diagnostics approach (4SFD), which allows dynamically selecting the features to be used for performing the diagnosis, detecting the necessity of updating the diagnostic model and automatically updating it. Within the proposed approach, the main novelty is the semi-supervised feature selection method developed to dynamically select the set of features in response to the evolving environment. An artificial Gaussian and a real world bearing datasets are considered for the verification of the proposed approach.

Key words: Evolving Environment, Feature Selection, Concept Drift, Drift Detection, Fault Diagnostics, Bearing Faults

1. INTRODUCTION

In industry, components and equipment operate in evolving environments characterized by working conditions that change often in an unpredictable way. A possible approach for fault diagnostics in variable operating conditions consists in training a dedicated diagnostic model for each possible set of operating conditions. Then, the model trained using the operating conditions most similar to that of the test pattern is selected for the diagnosis [1,2]. In the case in which the information on the operating condition experienced by components and equipment is not available, the use of ensemble of diagnostic models has been proposed. In practice, different diagnostic models are trained, considering diagnostic data collected at different operating conditions, and the individual model outcomes are

properly aggregated by using a majority voting approach in order to provide the final classification of the test pattern [3–6].

In case of evolving environment, one of the major challenges of performing fault diagnostics is that the training data available to build the diagnostic model typically do not include all possible operating and environmental conditions that the component will experience during its life. As a result, if the diagnostic model is used in working conditions different from those considered during the model training, its performance may be unsatisfactory [7,8]. Furthermore, in many industrial applications, collecting data for training a diagnostic model is a difficult, time consuming and very expensive task which requires the collection and analysis of sensor data for many years or performing ad-hoc experimental tests. In many cases, the only available training data are acquired in a laboratory environment in operating conditions very different from those that will be experienced by the component during its life.

The objective of this work is to develop a systematic framework for performing fault diagnostics in evolving environments, given that the training set are not representative for all the operating and environmental conditions that the component will experience during its life. The diagnostic approach, which will be referred to as "Systematic Semi-Supervised Self-adaptable Fault Diagnostic (4SFD)", is able to deal with the evolving environments in all phases of development of the diagnostic model:

1) selection of the feature set to be used by the diagnostic model;
2) development of the diagnostic model;
3) detection of the occurrence of a concept drift, i.e. an unforeseen modification of the statistical properties of the data which indicates the occurrence of evolving environments [9];
4) update of the diagnostic model, in case of concept drift detection; it can include a modification of the feature set (step 1) above) and/or a modification of the diagnostic model (step 2) above).

With respect to step 1), feature extraction methods are typically applied to raw signal measurements [10]. For example, statistical indicators, such as mean, kurtosis and skewness [11–13], wavelet transforms [14,15] and entropy [16] are commonly extracted from vibrational signals.

Although the number of features that can be potentially extracted from a raw signal using the above methods is very large, a lot of them are not useful in fault diagnostics and can degrade the performance of the classification model

[17–25]. This is due to the fact that i) irrelevant, non-informative features result in a classifier model which is not robust, ii) when the model handles many features, a large number of observation data is required to properly span the high-dimensional feature space for accurate multivariable interpolation iii) many input features unnecessarily increase the complexity of the classification model. Furthermore, studies have shown that for the success of the classification it is necessary to remove highly correlated features [24]. Thus, a feature selection algorithm is typically used to select the most representative features, which allows improving the performance of diagnostic model [14], and, at the same time, remarkably reduce its computational burden [16].

The objective of feature selection methods is to identify a subset of the available features such that the diagnostic model provides the most satisfactory performance. A feature selection algorithm is based on the definition of a procedure for searching a feature set in the space of all possible combinations of features. Then, the expected diagnostic performance of the proposed feature set is evaluated. In filter approaches, the evaluation considers statistical properties of the features and is independent from the diagnostic algorithm. The wrapper approach, on the other hand, uses the diagnostic algorithm as a part of the evaluation: the feature set performance is the classification accuracy obtained by training the classifier with the selected features [26,27]. Filter approaches are computationally simpler, faster, and easier to implement in high-dimensional feature sets, but they neglect the dependencies between feature sets and classifier. In many cases, this causes worse performance than wrapper approaches, which, on the contrary, use the classification accuracy as selection criterion. Notice, however, that the optimal feature set selected by a wrapper approach strongly depends on the classification algorithm, i.e. the selected feature set may not be optimal for another classification algorithm. In addition, the computational burden of wrapper approaches is significantly higher when dealing with a large number of features [28,29].

Both filter and wrapper feature selection approaches are typically applied off-line (before the development of the final diagnostic system), using labelled patterns describing the system behaviour in a static environment. Furthermore, once selected, the feature set is never changed. However, the capability of a feature to provide useful diagnostic information may depend on the working and environmental conditions experienced by the component. For example, with reference to a problem of fault diagnostics in bearings, the amplitude of the vibration is sensible to the bearing degradation and allows distinguishing between normal and faulty conditions when the torque applied to the bearing is large, whereas the same feature is not useful in case of low torque. Contrarily, the frequency of the vibration is more effective than the vibration amplitude, for fault diagnostics in the case of low torque. For this

reason, the solution that we investigate in this work is to dynamically modify the feature set in order to consider the presence of an evolving environment. In this context, the information available for feature selection includes two different sources:

a) labelled data containing signal values and corresponding fault classes. They can be historical data or data collected in laboratory tests. They typically do not cover all the possible working conditions that can be experienced by the component during operation;

b) unlabelled data containing only the signal values. They are typically collected from an evolving environment and they possibly refer to working conditions different from those of the labelled data. The unavailability of the labels in fault diagnostic problems is due to the fact that the identification of the fault causing the malfunctioning (label) is typically very expensive and time consuming.

Given the available information, the main novelty of the proposed method is the development of a semi-supervised feature selection method. Its main idea is to evaluate the candidate feature set by considering three indicators: 1) the classification accuracy and precision on the available labelled data of a Support Vector Machine (SVM) classifier; 2) the confidence of a SVM classifier trained using the available labelled data and tested on unlabelled data collected in an evolving environment; 3) the silhouette index of the unlabelled data classes provided by the SVM. Finally, a sparse Borda Count method (a modified version of the original Borda Count method [30]) is used to perform a multi-objective ranking of all the feature sets and, thus, to identify the one with the expected most satisfactory trade-off among the three indicators.

With respect to the development of the initial diagnostic model, we consider the available labelled data to train a SVM classifier. SVM has been chosen since it is a mature empirical method for developing a classifier and its satisfactory classification performance in fault diagnostic applications has been verified [31–33].

The detection of the occurrence of a concept drift which causes a degradation of the diagnostic model accuracy is performed by using an α shape reconstruction technique [34] already introduced by the authors in [35]. Here the technique is modified in order to allow distinguishing gradual modifications of the working conditions only requiring an updating of the classifier from abrupt modifications, such as sudden large changes of operational or environmental conditions, requiring performing a new feature selection. In the former case, the classifier is updated using an algorithm inspired by the COMPacted Object Sample Extraction (COMPOSE) algorithm [36].

Two case studies are considered to verify the effectiveness of the proposed 4SFD method: 1) an artificial Gaussian dataset with simulated concept drifts and 2) a laboratory bearing dataset taken from the Case Western Reserve University characterized by nine different fault types and four different working loads.

Since the technical details of the α shape based drift detection and COMPOSE algorithm have been introduced in [37], this paper will focus on the overall scheme of the 4SFD method (Section 2), and the semi-supervised feature selection (Section 3). Section 4 shows the results of applying 4SFD to the two different case studies and Section 5 gives the conclusion of the whole paper.

2. 4SFD FRAMEWORK

The 4SFD method starts with an initial off-line feature selection and the development of a classifier, *f*. Both the feature selection and the development of the classification model use the available labelled data ($T = \{X_T, L_T\}$). The classification model considered in this work is a SVM, based on the pairwise coupling [38], which provides in output the probabilities, p_{jk}, that the *j*-th test pattern belongs to class *k*, *k*=1,…, N_{cl}. The initial feature selection is performed by a wrapper supervised approach for the maximization of the classification accuracy.

In operation, when the signal measurements arrive, the selected features are extracted and sent to the SVM for classification and to the concept drift detector. This latter module operates online is to detect a concept drift, and, interpret if it is a gradual or abrupt. In the case of no drift detection, the classes provided by the SVM are accepted; on the contrary, if a drift is detected, the classification model is updated. For the drift detected as gradual, the COMPOSE algorithm [36] is applied to identify a proper set of data to train the new SVM classifier; if the drift is abrupt, a new selection of the features must be performed. The flowchart of the 4SFD is shown in Figure 1~~Figure 1~~.

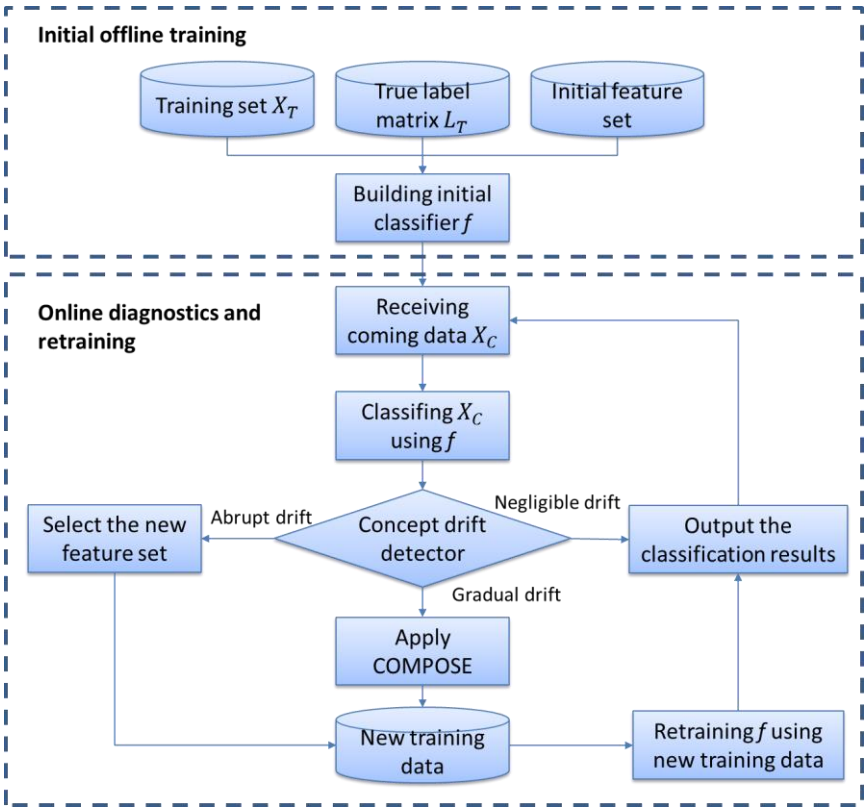

Figure 1 Flowchart of the 4SFD

Sections 3.1 and 3.2 will illustrate the concept drift detector and the algorithm used to update the classifier training set. which are modifications of already presented methods in [36,37]. In Section 4, the novel feature selection method is illustrated.

2.1. The concept drift detector

The detection of a concept drift is based on the use of the α shape surface reconstruction method [34], which allows computing the volume of a surface enveloping a set of data. The basic idea behind the concept drift detection is that test data collected after the occurrence of a concept drift are expected to be outside the α shape surface enveloping the training set [37]. Thus, the volume, $\acute{V}$, of the α shape surface enveloping the union of the training and test

patterns is larger than the volume, V, enveloping only the training patterns. The detection is based on the definition of two thresholds Th_n and Th_g, for the ratio R between the volumes V' and V:

$$R = \frac{V'}{V} \tag{1}$$

If R is lower than Th_n, no drift is detected since the test patterns are close to the training patterns; if R is between Th_n and Th_g, a gradual drift is detected, being the test patterns close to the train patterns, but outside the volume enveloping the training patterns, whereas, if R is greater than Th_g, the occurrence of an abrupt concept drift causing major modifications of the feature values is detected. The threshold parameters, Th_n and Th_g, are set considering the characteristics of the specific case study (in our case, the values are 1.05 and 1.2, respectively). More details on the α shape surface reconstruction method and its use for concept drift detection can be found in [36,37].

2.2. Updating of the classifier training set

According to the scheme of Figure 1~~Figure 1~~, once a gradual concept drift has been detected, the classification model is updated to take into account the effects of the evolving environment on the signal measurements. The classifier updating requires the construction of a new training set containing information extracted from the new working conditions. In this work, the construction of the new training set is performed with a modification of the original COMPOSE algorithm proposed in [36]. The basic assumption behind the algorithm is that the core region of the training data overlaps, at least partially, with a part of the drifted data. The key idea of the method is to aggregate the labelled training data with the unlabelled new data and to perform a shrinkage of the obtained dataset in order to identify a core region representing the trend of the concept drift. More details on the method can be found in Appendix 2 and [37].

3. SEMI-SUPERVISED FEATURE SELECTION METHOD

In this Section, we address the problem of selecting the feature set to be used for fault diagnostics in an evolving environment. The algorithm is applied each time an abrupt drift is detected. The set of all the N_{FS} features extracted from the raw signal measurements is indicated by $TFS = \{F_1, F_2, \dots, F_{N_{\mathrm{FS}}}\}$, and the candidate feature sets, FS_i, $FS_i \subset$ TFS, are the $2^{N_{\mathrm{FS}}} - 1$ possible combinations of the features. The reader interested in more advanced techniques for exploring all the possible feature sets, without considering exhaustive searches, may refer to [39,40].

In general, the inputs of diagnostic model are features values, X, extracted from the measured signals, and the output is the class label of the fault, L. The information available to develop the diagnostic model is:

- a set of labelled data, $T = \{X_T, L_T\}$, which contains both the feature values, X_T and the corresponding fault labels, L_T. X_T is a $N_T * N_{\mathrm{FS}}$ dimensional matrix, with N_T indicating the total number of patterns in X_T and N_{FS} the number of features, whereas the pattern labels are indicated by the $N_T * N_{\mathrm{cl}}$ dimensional matrix L_T:

$$L_T = \begin{pmatrix} h_{11} & \cdots & h_{1N_{cl}} \\ \vdots & \ddots & \vdots \\ h_{j1} & h_{jk} & h_{jN_{cl}} \\ \vdots & \ddots & \vdots \\ h_{N_T 1} & \cdots & h_{N_T N_{cl}} \end{pmatrix}, \quad h_{jk} = \begin{cases} 1, & \text{if pattern } j \text{ belong to class } k \\ 0, & \text{otherwise} \end{cases}, \; j = 1, 2, ..., N_T, k = 1, 2, ..., N_{cl}$$

  with N_{cl} indicating the total number of classes in the training set.

- a set of unlabelled data, $C = \{X_C\}$, which is a $N_C * N_{\mathrm{FS}}$ dimensional matrix containing only the feature values, X_C ; with N_C indicating the total number of patterns in X_C. These data are collected in batches and describe the component behaviour in evolving environments.

Usually, the labelled training data T is given in the known working condition, and the unlabelled data C are collected during the online phase, after the detection of the concept drift. The problem is addressed by developing a novel feature selection approach where a SVM classifier, f, is built based on T and the proposed feature set. Then, its expected performance in an evolving environment is evaluated. To this aim, since the classification accuracy in the new environment cannot be computed due to the unavailability of labelled data, we have considered three metrics (Section 4.1) and we have aggregated them into one performance indicator by using a Sparse Borda Count algorithm (Section 4.2).

3.1. Indicator of the expected performance of a feature set in a new environment

In order to assess the performance of a feature set in a new environment, a classifier whose inputs are the selected features is trained using a part of the available labelled data. Then, the following three metrics are computed:

- Indicator A: accuracy in the classification of labelled data collected previously to the concept drift occurrence and not used to build the classification model;

- Indicator B: confidence in the classification of the unlabelled data collected in the new environment (after the concept drift occurrence);
- Indicator C: compactness and separation of the classes assigned to the unlabelled data collected in the new environment (after the concept drift occurrence).

Indicator A is the performance indicator traditionally used in wrapper approaches to quantify the capability of the feature set of correctly classifying test data in stationary working conditions. It is considered since, within the COMPOSE scheme, the classifier is used to label data collected from the evolving environment, and, thus, errors in the classification of the labelled data would dramatically propagate and result in the retraining of classifiers with wrongly labelled data. Notice, however, that a satisfactory value of indicator A does not automatically guarantee a high accuracy of the diagnostic model in an evolving environment, i.e. in the classification of unlabelled data, X_C, collected after the occurrence of a concept drift. For this reason, indicators B and C are introduced to quantify the performance of the feature set in a new environment. Given the unavailability of the true labels of the patterns in X_C, indicators B and C focus on the classifications provided by the SVM classifier, being indicator B based on the idea that a good feature set should provide confident classifications of the data and indicator C on the idea that the classes identified in X_C should from compact and well separated clusters.

3.1.1. Indicator A: Accuracy on the labelled data

The idea is to quantify the accuracy of the classification model built using as input the feature set of interest and taking into account only the labelled data, $T = \{X_T, L_T\}$, collected before the occurrence of a concept drift. A classification model is accurate if it correctly assigns the true class to test patterns not used for model training. Being the outcome of the SVM classifier the probabilities, p_{jk} that the *j*-th test pattern belongs to the *k*-th class, a measure of the accuracy in the classification of the test set is provided by:

$$IA = 1 - \frac{\sum_{j=1}^{N_{te}} \sum_{k=1}^{N_{cl}} \left| h_{jk} - p_{jk} \right|}{N_{cl} \cdot N_{te}} \tag{2}$$

with N_{te} indicating the number of test patterns.

In order to obtain a robust evaluation of the accuracy, even in the case in which few labelled data are available, a Cross Validation (CV) procedure is applied. In practice, we repeat 10 times the random partition of the labelled dataset $T = \{X_T, L_T\}$ into two subsets formed by the same number of patterns, and we use the first one to train the SVM classifier and the second one to compute its accuracy. Eventually, the feature set accuracy is the average of the accuracy values obtained in the 10 runs:

$$IA = \frac{\sum_{i=1}^{10} IA_i}{10} \tag{3}$$

The value of IA is between 0 (all patterns misclassified) and 1 (all patterns correctly classified).

3.1.2. Indicator B: Confidence of unlabelled coming set

This metric measures how much the classifier built using the labelled data T collected before the concept drift occurrence is able to provide confident classifications of the unlabelled data C in the new environment. According to [41,42], the confidence of the classifier can be evaluated by considering the entropy:

$$E = \sum_{j=1}^{N_C} \sum_{k=1}^{N_{cl}} -p_{jk} \cdot \log p_{jk} \tag{4}$$

Entropy is a measure of the information content in the matrix p_{jk}: the smaller the entropy, the more confident the classification. However, since in fault diagnostic applications the major concern of the decision maker is to have a class clearly preferable from the others, the use of the entropy measure can have limitations. Let us consider, for example, a case of two classifiers which assign the same test patterns to classes 1,2 and 3 with the following probabilities: $O_1 = [0.6, 0.2, 0.2]$ and $O_2 = [0.6, 0.39, 0.01]$. According to the entropy measure, the classification O_2 would be evaluated as more confident than O_1, being entropy $E_2 = 0.72 < E_1 = 0.95$. However, from the point of view of the decision maker, even if in both cases the probability of class 1 is 0.6, he/she is more confident that the test pattern belongs to class 1 considering the classification O_1. This is due to fact that the second most probable class has a lower probability value in O_1 than that in O_2. In order to overtake this limitation of the entropy metric, in this work we propose a new confidence metric based on the evaluation of the difference between the probabilities of the class with the maximum probability and that with the second maximum probability. In particular, indicator B is defined by:

$$IB=\sum_{j=1}^{N_C}\left|\lambda_j-\mu_j\right| \tag{5}$$

where $\lambda_j=\max_{k=1:N_C}\left(p_{jk}\right)$, and μ_j is the second largest value among the p_{jk} values in row j. The larger IB, more confident is the classifier.

3.1.3. Indicator C: Silhouette index of the unlabelled data

Similarly to indicator B, indicator C considers the classification of the unlabelled data provided by the SVM classifier. Its objective is to evaluate whether the fault classes are easy to distinguish in the new environment. The conjecture is that if the classes assigned by the SVM classifier to the unlabelled test patterns are compact and well separated, then the classification accuracy is expected to be satisfactory. According to [43], compactness and separability of the obtained classification is evaluated considering the average silhouette index over all the test patterns:

$$IC=\frac{1}{N_C}\sum_{j=1}^{N_C}\frac{\left(b_j-a_j\right)}{\max\left(a_j,b_j\right)} \tag{6}$$

where a_j is the average Euclidean distance between the j-th pattern and the other patterns of the same class, and b_j is the distance between the j-th test pattern and the nearest pattern of another class, averaged over all the classes. The IC value ranges from -1 to 1: the larger the IC, the more separated and compact are the classes.

3.2. Aggregation of the three indicators

Once the three indicators have been computed for all the feature sets of interest, it is necessary to decide which is the feature set to be used for fault diagnostics in the new environment. This is a group decision-making process which involves aggregating the information from three multiple sources [44,45]. The problem is here addressed using a modified version of the Borda count method, which has been successfully applied in very different application fields [46,47]. Borda count is a single-winner vote method which ranks candidates according to the sum of ballots from all the voters. A drawback of the algorithm is that the final rank depends on irrelevant candidates, i.e. removing a candidate can potentially modify the ranking of the other candidates [30]. In order to overtake this limitation of the traditional Borda count method, we apply its modification proposed in [30], which focuses on the best and worst candidates. In practice, the modification consists in assigning ballots only to the feature sets in the first and last

quartiles of the rankings originated by the three indicators. The modified Borda count procedure is based on the following steps:

1) **Individual ranking:** rank all the candidate feature sets with respect to each indicator;

2) **Filtering:** filter the candidate feature sets by considering for each indicator only the feature sets in the upper and lower quartiles;

3) **Voting:** a score F_{upper}^{i} is assigned to each candidate feature set in the upper quartile of the distribution of the *i*-th indicator, *i*=1,2,3. Assuming that there are *x* candidate feature sets in the upper (lower)quartile, the mark 1 is assigned to the feature set in the upper quartile with the smallest indicator value, the mark 2 to the second-smallest, … the mark *x* to the feature set with the largest indicator value. Similarly, a score F_{lower}^{i} is assigned to all the feature sets in the lower quartile of the distribution of the *i*-th indicator: *x* to the feature set with the smallest indicator value, *x*-1 to the second smallest, 1 to the feature set with the largest indicator value in the lower quartile. The final F_{upper} (F_{lower}) value associated to a feature set is the sum of all the scores F_{upper}^{i} (F_{lower}^{i}) on all the indicators *IA*,*IB* and *IC*.

4) **Choosing:** Calculate the final score F_{final} of each candidate feature set based on equation (7), and select the candidate feature set with largest F_{final}:

$$F_{final} = F_{upper} - F_{lower} \tag{7}$$

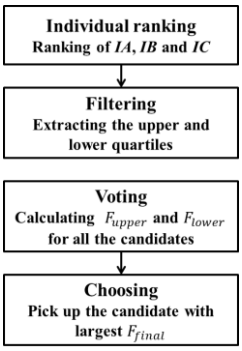

Figure 2 Sketch of the modified Borda count procedure

Once the best performing feature set is identified by the modified Borda count algorithm, the COMPOSE algorithm is applied to identify dataset for training the new SVM classifier based on the selected features; this will be used for fault diagnostics in the new environment, until a new concept drift is detected.

4. CASE STUDIES

In this Section, we test the performance of the 4SFD method considering one case study based on synthetic data and one diagnostic application concerning the classification of bearing defects in an evolving environment. The classification accuracy achieved by the 4SFD method is compared with that provided by:

Method 1): a SVM classifier built considering the labelled data and never changed during the tests; the input features are selected by applying a wrapper feature selection algorithm to the labelled data.

Method 2): the COMPOSE-based method described in [37]; the input features are selected by applying a wrapper feature selection algorithm to the labelled data and never changed during the tests.

Method 3): a SVM built using all the available features.

In all the three cases, the SVM classifiers are built using the "LIBSVM" source code [48].

4.1. Case study based on synthetic data

This case study mimics the occurrence of failures in an evolving environment characterized by periodic modifications of the operating conditions which are typically encountered by several components and industrial systems, such as the variation of the electricity production in an energy production plant [49], the variation of load experienced by automotive bearings [50], the variation of the external conditions experienced by a structure due to seasonal effects. The case study is built on that proposed in [36] taking into account the fact that the sensibility of the signals to the faults may change depending to the experienced operational conditions (e.g. vibration frequency is more useful for bearing fault diagnostics at low torque than at large torque).

In this case study, a 9-dimensional, 3-classes labelled dataset, $T = \{X_T, L_T\}$, is artificially generated. We assume to have this dataset available at time t=0. Features 1-6 values are sampled from a different 6-dimensional Gaussian distribution for each class (Table 1~~Table 1~~), whereas Features 7-9 values are sampled from a Gaussian distribution with zero mean and unitary standard deviation independently from the pattern class. These labelled data are used for the initial feature selection and the SVM classifier training. Then, the presence of an evolving environment is

simulated by assuming that batches of data become progressively available. In particular, every step of arbitrary time unit a batch formed by 30 patterns of each class is collected. Similar to the patterns in dataset *T*, Features 1-6 values are sampled from 6-dimensional Gaussian distributions, whose mean μ are changing with time according to the laws reported in Table 2~~Table 2~~, being the intensity of the concept drift controlled by parameters s_1, s_2, s_3 and z_1, z_2, z_3. Features 7-9 are always sampled from the same distributions used for *T* and are independent from the pattern classes. All the patterns provided to the diagnostic models are unlabelled except those of *T*. The overall dataset simulation is based on the repetition of 5 cycles, with each cycle formed by the sampling of 20 batches of data. Figure 3~~Figure 3~~ shows the distributions of the three-class data in the batches sampled at time 1, 5, 9, 15. Notice that features 7-9, being random noises, do not provide useful information for the data classification. At time *t*=1, the three classes are well separated and compact when observed in the subspace generated by features 1,2 and 3, whereas they are mixed and confused when observed in the subspace generated by features 4, 5 and 6. Then, the separation of the classes in the subspace of features 1, 2 and 3 gradually decreases until time *t*=9, whereas it increases in the subspace of features 4, 5 and 6. Contrarily, from time *t* = 11 to time *t* = 20 the Gaussian distributions gradually become more separated when observed in the subspace of features 1, 2 and 3 and less separated in the subspace of features 4, 5 and 6. Finally, at time *t* = 20 the classes are sampled from the same initial distribution and the sampling cycle is repeated.

Table 1 Mean and standard deviation values used for the generation of the initial dataset *T*.

Training dataset						
	F_1	F_2	F_3	F_4	F_5	F_6
Class 1	$N(\mu 1_0, 1)$	$N(\mu 1_0, 1)$	$N(\mu 1_0, 1)$	$N(\mu 2_0, 1)$	$N(\mu 2_0, 1)$	$N(\mu 2_0, 1)$
Class 2	$N(\mu 3_0, 1)$	$N(\mu 3_0, 1)$	$N(\mu 3_0, 1)$	$N(\mu 4_0, 1)$	$N(\mu 4_0, 1)$	$N(\mu 4_0, 1)$
Class 3	$N(\mu 5_0, 1)$	$N(\mu 5_0, 1)$	$N(\mu 5_0, 1)$	$N(\mu 6_0, 1)$	$N(\mu 6_0, 1)$	$N(\mu 6_0, 1)$
Control parameters						
parameter	$\mu 1_0$	$\mu 2_0$	$\mu 3_0$	$\mu 4_0$	$\mu 5_0$	$\mu 6_0$
value	1	5	5	5	9	5

Table 2 Parameters used for the simulation of the presence of an evolving environment.

Testing dataset						
	$t\epsilon[1,10]$					
Class	F_1	F_2	F_3	F_4	F_5	F_6
C1	$N(\mu 1_{t-1} + s_1 t, 1)$	$N(\mu 1_{t-1} + s_1 t, 1)$	$N(\mu 1_{t-1} + s_1 t, 1)$	$N(\mu 2_{t-1} + z_1 t, 1)$	$N(\mu 2_{t-1} + z_1 t, 1)$	$N(\mu 2_{t-1} + z_1 t, 1)$
C2	$N(\mu 3_{t-1} + s_2 t, 1)$	$N(\mu 3_{t-1} + s_2 t, 1)$	$N(\mu 3_{t-1} + s_2 t, 1)$	$N(\mu 4_{t-1} + z_2 t, 1)$	$N(\mu 4_{t-1} + z_2 t, 1)$	$N(\mu 4_{t-1} + z_2 t, 1)$
C3	$N(\mu 5_{t-1} + s_3 t, 1)$	$N(\mu 5_{t-1} + s_3 t, 1)$	$N(\mu 5_{t-1} + s_3 t, 1)$	$N(\mu 6_{t-1} + z_3 t, 1)$	$N((\mu 6_{t-1} + z_3 t, 1)$	$N((\mu 6_{t-1} + z_3 t, 1)$
	$t\epsilon[11,20]$					
C1	$N(\mu 1_{t-1} - s_1 t, 1)$	$N(\mu 1_{t-1} - s_1 t, 1)$	$N(\mu 1_{t-1} - s_1 t, 1)$	$N(\mu 2_{t-1} - z_1 t, 1)$	$N(\mu 2_{t-1} - z_1 t, 1)$	$N(\mu 2_{t-1} - z_1 t, 1)$
C2	$N(\mu 3_{t-1} - s_2 t, 1)$	$N(\mu 3_{t-1} - s_2 t, 1)$	$N(\mu 3_{t-1} - s_2 t, 1)$	$N(\mu 4_{t-1} - z_2 t, 1)$	$N(\mu 4_{t-1} - z_2 t, 1)$	$N(\mu 4_{t-1} - z_2 t, 1)$
C3	$N(\mu 5_{t-1} - s_3 t, 1)$	$N(\mu 5_{t-1} - s_3 t, 1)$	$N(\mu 5_{t-1} - s_3 t, 1)$	$N(\mu 6_{t-1} - z_3 t, 1)$	$N(\mu 6_{t-1} - z_3 t, 1)$	$N(\mu 6_{t-1} - z_3 t, 1)$
	$t\epsilon[21,30]$					
C1	$N(\mu 1_{t-1} + s_1 t, 1)$	$N(\mu 1_{t-1} + s_1 t, 1)$	$N(\mu 1_{t-1} + s_1 t, 1)$	$N(\mu 2_{t-1} + z_1 t, 1)$	$N(\mu 2_{t-1} + z_1 t, 1)$	$N(\mu 2_{t-1} + z_1 t, 1)$
C2	$N(\mu 3_{t-1} + s_2 t, 1)$	$N(\mu 3_{t-1} + s_2 t, 1)$	$N(\mu 3_{t-1} + s_2 t, 1)$	$N(\mu 4_{t-1} + z_2 t, 1)$	$N(\mu 4_{t-1} + z_2 t, 1)$	$N(\mu 4_{t-1} + z_2 t, 1)$

C3	$N(\mu 5_{t-1}+s_3 t,1)$	$N(\mu 5_{t-1}+s_3 t,1)$	$N(\mu 5_{t-1}+s_3 t,1)$	$N(\mu 6_{t-1}+z_3 t,1)$	$N((\mu 6_{t-1}+z_3 t,1)$	$N((\mu 6_{t-1}+z_3 t,1)$
	$t\epsilon[31,40]$					
C1	$N(\mu 1_{t-1}-s_1 t,1)$	$N(\mu 1_{t-1}-s_1 t,1)$	$N(\mu 2_{t-1}-z_1 t,1)$	$N(\mu 2_{t-1}-z_1 t,1)$	$N(\mu 2_{t-1}-z_1 t,1)$	$N(\mu 2_{t-1}-z_1 t,1)$
C2	$N(\mu 3_{t-1}-s_2 t,1)$	$N(\mu 3_{t-1}-s_2 t,1)$	$N(\mu 4_{t-1}-z_2 t,1)$	$N(\mu 4_{t-1}-z_2 t,1)$	$N(\mu 4_{t-1}-z_2 t,1)$	$N(\mu 4_{t-1}-z_2 t,1)$
C3	$N(\mu 5_{t-1}-s_3 t,1)$	$N(\mu 5_{t-1}-s_3 t,1)$	$N(\mu 6_{t-1}-z_3 t,1)$	$N(\mu 6_{t-1}-z_3 t,1)$	$N(\mu 6_{t-1}-z_3 t,1)$	$N(\mu 6_{t-1}-z_3 t,1)$
	... repeat until t=100					
parameter	s_1	s_2	s_3	z_1	z_2	z_3
value	0.26	0.25	0.28	0.31	0.33	0.32

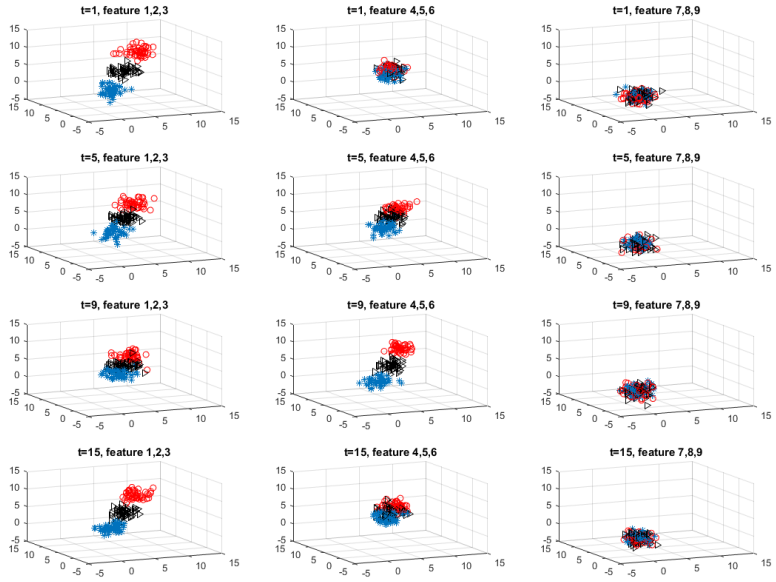

Figure 3 Three dimensional projections of the data batches in different feature subspaces

An initial supervised wrapper feature selection is performed using the labelled data available at time *t*=0. The objective is to identify the feature set which provides the best classification accuracy, and, in order to reduce the computational burden of the feature selection task, we have considered only feature sets formed by three features. In particular, as expected, an exhaustive search among all the 84 possible three-dimensional feature sets has selected the feature set formed by features 1, 2, and 3 as the one with the associated most satisfactory accuracy. This feature set has been used as initial feature set for the 4SFD method and as fixed feature set for the SVM of method 1) and the COMPOSE algorithm of method 2). In order to verify the effectiveness of the proposed feature selection approach,

we also compare the classification accuracy by using all the 9 features (without the feature selection) as the input of SVM. Figure 4~~Figure 4~~ shows the classification accuracy provided by the three methods. Notice that:

a) the SVM classifier of method 1) provides satisfactory classification performances only when the distributions used to sample the data are similar to those used to train the classifier at time t=0, i.e. around times 1, 20, 40, 60, 80, 100. On the contrary, as expected, the performance is quite unsatisfactory at time 10, 30, 50, 70, 90, when the classes are very confused in the subspace of features 1, 2 and 3.

b) the SVM classifier built using all the 9 features performs slightly better than that obtained using only features 1, 2 and 3, but it still underperforms at times t = 10, 30, 50, 70 and 90. The main reason is that the patterns of the different classes are partially overlapped with respect to features 4, 5 and 6 in the initial training set. Thus, the SVM cannot provide satisfactory performances when the patterns of the different classes become well separated with respect to features 4, 5 and 6.

c) the COMPOSE algorithm of method 2) is slightly better performing than the SVM classifier when data drift, given its ability of learning data modifications. On the other hand, the COMPOSE performance is not able to fully recover, as the SVM does, at time t = 20, 40, 60, 80 and 100 when data are sampled from the same distributions used at time t = 0. This is due to the error accumulation caused by the addition of training patterns collected from the evolving environment, whose true class is not known. Furthermore, given the inadequacy of features 1, 2 and 3 of distinguishing the classes at times around t = 10, 30, 50, 70 and 90, its overall performance is unsatisfactory.

d) the accuracy of the 4SFD method is more satisfactory than that provided by methods 1, 2 and 3. This is due the 4SFD capability of changing the feature set when data in the previously used feature set becomes less separated and distinguishable than in other possible new feature sets. This occurs for the first time at time t = 6 when the concept drift detector identifies an abrupt concept drift. Thus, a new feature set formed by features 4, 5 and 6 is selected using the three performance indicators and the Borda count procedure of Section 4.2 (Table 3~~Table 3~~). Notice that the selected feature set is characterized by the largest values of all the three performance indicators and, thus, is selected by the Borda count method. The column “ground truth accuracy” in Table 6 provides the percentage of patterns which would be correctly classified by a classifier trained with labelled patterns sampled from the same distribution originating the test patterns. The purpose here is to confirm that the selected feature set is able to provide the most satisfactory performance among all the possible three-dimensional feature sets.

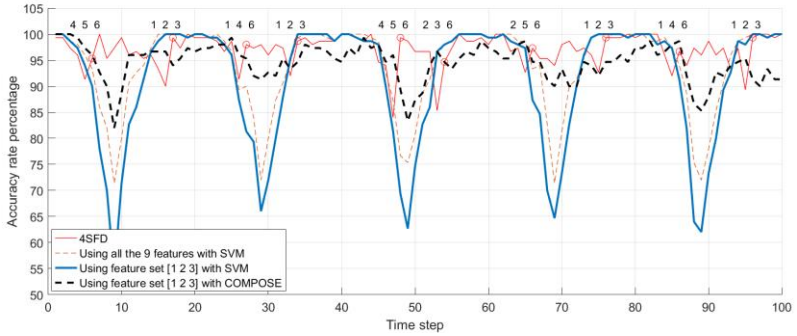

Figure 4 Percentage of patterns correctly classified by the three different methods. The circles indicate the time at which an abrupt concept drift is identified and the new feature set, formed by the features reported on the right side, is selected and used

Table 3 Borda count Table at time step 6.

feature set			*IA*	*IB*	*IC*	*IA* rank	*IB* rank	*IC* rank	F_{final}	ground truth accuracy	ground truth accuracy rank
4	**5**	**6**	**0.928**	**135.617**	**0.714**	**1**	**1**	**1**	252	**98.000**	**1**
1	4	5	0.908	131.432	0.671	15	12	10	184	98.000	1
3	5	6	0.910	130.986	0.682	12	13	7	194	98.000	1
1	4	6	0.917	131.438	0.694	8	11	4	212	97.333	4
2	5	6	0.913	134.342	0.702	11	3	3	224	97.333	4
1	3	6	0.920	132.769	0.652	4	9	13	206	96.667	6
2	3	6	0.900	126.485	0.641	21	26	18	128	96.667	6
2	4	5	0.917	130.218	0.669	9	15	11	188	96.667	6
2	4	6	0.919	133.972	0.691	6	5	5	226	96.667	6
3	4	5	0.926	134.256	0.682	2	4	8	230	96.667	6
.....											
7	8	9	0.552	5.961	0.174	84	84	84	-252	26.667	84

Figure 4~~Figure 4~~ shows that a concept drift is detected two times in each cycle: when features 4, 5, and 6 become more efficient than features 1, 2, 3 (times t=6, 27, 47, 67, 89) and when features 1, 2 and 3 return to be the most efficient (times t=15, 35, 56, 75, 96). The selected feature sets always contain at least two of the three most efficient features and are able to guarantee very high performances.

4.2. Bearing dataset: Case Western Reserve University Bearing dataset

This case study is designed based on the data reported in the Western Reserve University bearing dataset [51]. The experimental dataset contains 720 patterns referring to 9 different faults and 4 different working loads, as shown in Table 4~~Table 4~~.

Table 4 Attribute of bearing dataset 1

Fault label	Fault location	Fault intensity	working load
1	Inner race	7 mils	0,1,2,3 horsepower
2	Inner race	14 mils	0,1,2,3 horsepower
3	Inner race	21 mils	0,1,2,3 horsepower
4	Balls	7 mils	0,1,2,3 horsepower
5	Balls	14 mils	0,1,2,3 horsepower
6	Balls	21 mils	0,1,2,3 horsepower
7	Outer race	7 mils	0,1,2,3 horsepower
8	Outer race	14 mils	0,1,2,3 horsepower
9	Outer race	21 mils	0,1,2,3 horsepower

For each fault, 80 patterns are available, 20 for each working load. Each pattern is formed by the vibrational raw measurement collected by three accelerometers in a time window of 1.4 seconds at a frequency of 12,000 Hz. Among all the possible features that can be extracted from the raw measurements, we have preselected the 15 features reported in Appendix 1, by applying an unsupervised spectral feature selection method [52]. The obtained dataset, formed by 720 labelled 15-dimensional patterns, has been used to design 4 different experiments in order to test the performance of 4SFD and of the two reference methods. In all the experiments, the presence of an evolving environment is simulated by assuming that data become progressively available in batches and each batch contains patterns collected at a different load from the previous one. In all the experiments, a labelled dataset formed by 180 patterns at a given load is initially available (dataset T), whereas 6 batches formed by 90 unlabelled patterns become progressively available. The four experiments differ in the sequence with which the loads become available (Table 5~~Table 5~~).

Table 5 Sequence of the loads in the four experiments. Loads 1, 2, 3, 4 refer to horsepower 0, 1, 2, 3 respectively.

	Labelled Dataset (T)	Batch 1	Batch 2	Batch 3	Batch 4	Batch 5	Batch 6
Experiment 1	load 1 (180 patterns)	load 2 (90 patterns)	load 4 (90 patterns)	load 2 (90 patterns)	load 3 (90 patterns)	load 4 (90 patterns)	load 3 (90 patterns)
Experiment 2	load 2	load 3	load 1	load 4	load 3	load 4	load 1

	(180 patterns)	(90 patterns)	(90 patterns)	(90 patterns)	(90 patterns)	(90 patterns)	(90 patterns)
Experiment 3	load 3 (180 patterns)	load 2 (90 patterns)	load 4 (90 patterns)	load 1 (90 patterns)	load 2 (90 patterns)	load 1 (90 patterns)	load 4 (90 patterns)
Experiment 4	load 4 (180 patterns)	load 3 (90 patterns)	load 1 (90 patterns)	load 3 (90 patterns)	load 2 (90 patterns)	load 1 (90 patterns)	load 2 (90 patterns)

Similarly to the case study with the synthetic dataset, we consider only feature sets formed by 3 features in order to reduce the computational effort. The feature set formed by features {2, 26, 49} has been selected as initial feature set using the supervised feature selection method in [37,53]. Figure 5~~Figure 5~~ shows the classification accuracy obtained by the 3 methods in these 4 experiments.

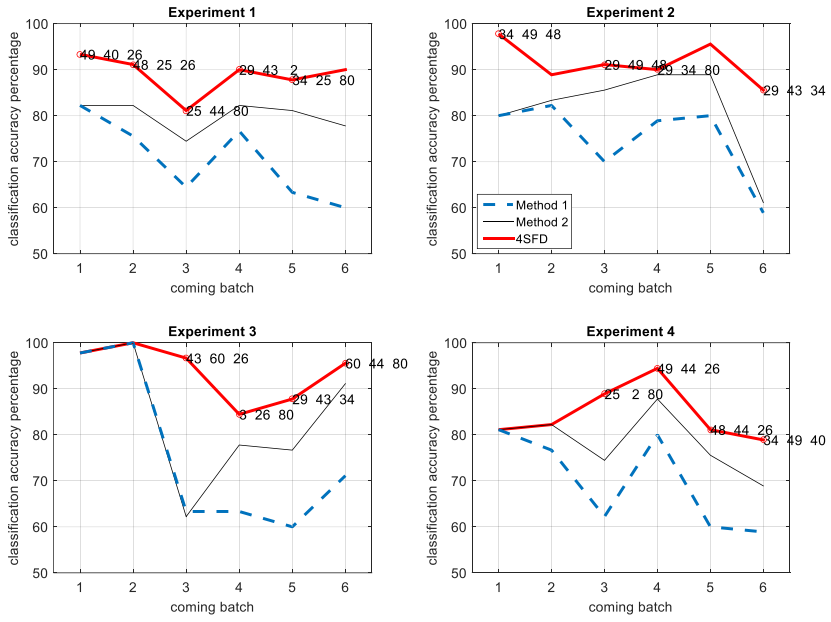

Figure 5 Classification accuracy percentage of the three different methods applied to the bearing case study

Similarly to the previous case study, the 4SFD method provides an overall more satisfactory performance than methods 1 and 2, thanks to its ability of changing the feature set. Method 2, based on COMPOSE, is able to learn the concept drift in case of gradual drift, e.g. in experiment 4 when the second batch becomes available, but is not able to

handle abrupt drifts, e.g. batch 3 in experiment 3. Figure 6~~Figure 6~~ shows that the feature set {43, 60, 26} selected by the 4SFD method in experiment 3, batch 3, allows obtaining more separated and compact classes than the initial feature set {2, 26, 49} used by the COMPOSE method.

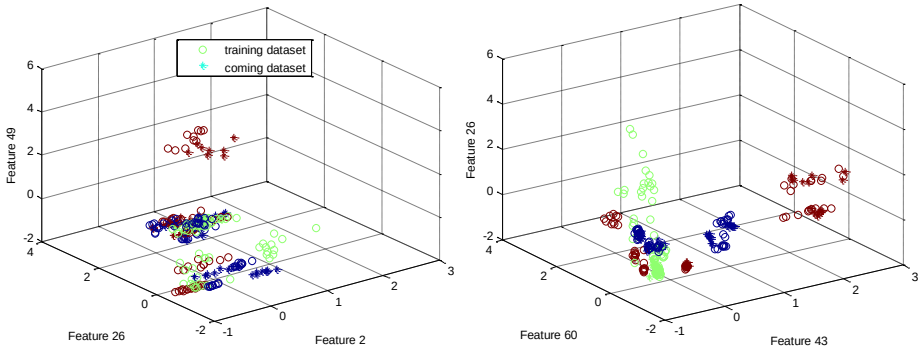

Figure 6 Projection of the initial dataset (circles) and of the data in batch 3 in experiment 3 on the initial feature set {2, 26, 49} used by the COMPOSE method (left) and on the feature set {43, 60, 26} used by the 4SFD method (right). Different colours indicate different classes of the data.

Table 6~~Table 6~~ reports the values of the three indicators used for the feature selection task by the 4SFD, when batch 3 in experiment 3 becomes available. Notice that, as expected, the selected feature set {43, 60, 26} is characterized by larger (more satisfactory) indicator values than the initial feature set {2, 26, 49}. Furthermore, the obtained accuracy is close to that of the feature set with the largest ground truth accuracy, {26, 49, 80}, which would be obtained using the true label information.

Table 6 Borda count Table of feature sets {2, 26, 49}, {43, 60, 26} and {26, 49, 80}

feature set			*IA*	*IB*	*IC*	*IA* rank	*IB* rank	*IC* rank	F_{final}	ground truth accuracy	ground truth accuracy rank
26	46	80	0.788	60.211	0.071	70	39	244	801	97.333	1
…											
43	60	26	0.784	64.543	0.059	83	13	267	814	96.677	10
…											
2	26	49	0.645	38.783	0.034	199	256	271	0	62.222	294
…											

5. CONCLUSION

In this work, we have developed a novel framework for performing fault diagnostics in evolving environments. The proposed Systematic Semi-Supervised Self-adaptable Fault Diagnostics approach (4SFD), allows detecting the need for updating the diagnostic model, dynamically selecting the features to be used for the diagnosis, and automatically updating the diagnostic model. Its main novelty is that it exploits the information provided by both labelled and unlabelled data and it can automatically adapt itself to the evolving environment by updating the features set used for the diagnosis.

The method is capable of handling the occurrence of concept drifts of different intensities and of automatically deciding whether it is necessary to update the classification model, in order to adapt it to the new environment, or it is required to select new features for the classification. Two case studies show its superiority with respect to traditional fault diagnostics methods in terms of diagnostic accuracy.

It is expected that the developed 4SFD will contribute to improve maintenance practice of engineering equipment which are subject to varying operating conditions and for which it is not possible to collect training data representative of all the possible working conditions experienced by the equipment during its lifetime. These conditions are typically encountered by electric components, bearings, gears, alternators, shafts and pumps in different industrial sectors such as aeronautics, automotive and electricity production plants.

The major limitation of the 4SFD method lies in its scalability to high-dimensional feature selection problems. Due to the effect of the curse of dimensionality, the computational efforts required to perform the time consuming exhaustive search of the best performing feature set among all the possible feature combinations tends to increase significantly with respect to the number of available features. The possibility of overtaking this limitation by adopting for the selection of the feature set a heuristic optimization algorithm, such as Genetic Algorithms, Differential Evolution, Ant Colony Algorithm and Particle Swarm Optimization algorithms will be object of future research work.

ACKNOWLEDGEMENT

Yang Hu gratefully acknowledges the financial support from China Scholarship Council and Politecnico di Milano (No. 201206110018). The participation of Enrico Zio to this research is partially supported by the China NSFC under grant number 71231001. The participation of Piero Baraldi and Francesco Di Maio is supported by the European

Union Project INNovation through Human Factors in risk analysis and management (INNHF, www.innhf.eu) funded by the 7th framework program FP7-PEOPLE-2011- Initial Training Network: Marie-Curie Action.

REFERENCES

[1] Jardine AKS, Lin D, Banjevic D. A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mech Syst Signal Process 2006;20:1483–510. doi:10.1016/j.ymssp.2005.09.012.

[2] Halme J, Andersson P. Rolling contact fatigue and wear fundamentals for rolling bearing diagnostics - state of the art. IME Part J J Eng Tribol 2010;224:377–93. doi:10.1243/13506501JET656.

[3] Baraldi P, Razavi-Far R, Zio E. Classifier-ensemble incremental-learning procedure for nuclear transient identification at different operational conditions. Reliab Eng Syst Saf 2011;96:480–8. doi:10.1016/j.ress.2010.11.005.

[4] Zhang X, Zhou J. Multi-fault diagnosis for rolling element bearings based on ensemble empirical mode decomposition and optimized support vector machines. Mech Syst Signal Process 2013;41:127–40. doi:10.1016/j.ymssp.2013.07.006.

[5] Xu Z, Li Y, Wang Z, Xuan J. A selective fuzzy ARTMAP ensemble and its application to the fault diagnosis of rolling element bearing. Neurocomputing 2016;182:25–35. doi:10.1016/j.neucom.2015.12.015.

[6] Zhang X, Wang B, Chen X. Intelligent fault diagnosis of roller bearings with multivariable ensemble-based incremental support vector machine. Knowledge-Based Syst 2015;89:56–85. doi:10.1016/j.knosys.2015.06.017.

[7] Gonçalves PM, de Carvalho Santos SGT, Barros RSM, Vieira DCL. A comparative study on concept drift detectors. Expert Syst Appl 2014;41:8144–56. doi:10.1016/j.eswa.2014.07.019.

[8] E IEŽ. Learning under Concept Drift: an Overview. Training 2010;abs/1010.4:1–36.

[9] Elwell R, Polikar R. Incremental learning of concept drift in nonstationary environments. IEEE Trans Neural Netw 2011;22:1517–31. doi:10.1109/TNN.2011.2160459.

[10] Feng Z, Liang M, Chu F. Recent advances in time–frequency analysis methods for machinery fault diagnosis: A review with application examples. Mech Syst Signal Process 2013;38:165–205. doi:10.1016/j.ymssp.2013.01.017.

[11] Di Maio F, Tsui KL, Zio E. Combining Relevance Vector Machines and exponential regression for bearing residual life estimation. Mech Syst Signal Process 2012;31:405–27. doi:10.1016/j.ymssp.2012.03.011.

[12] Xu Z, Xuan J, Shi T, Wu B, Hu Y. A novel fault diagnosis method of bearing based on improved fuzzy ARTMAP and modified distance discriminant technique. Expert Syst Appl 2009;36:11801–7. doi:10.1016/j.eswa.2009.04.021.

[13] Li B, Zhang PL, Wang ZJ, Mi SS, Liu DS. A weighted multi-scale morphological gradient filter for rolling element bearing fault detection. ISA Trans 2011;50:599–608. doi:10.1016/j.isatra.2011.06.003.

[14] Peng ZK, Chu FL. Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography. Mech Syst Signal Process 2004;18:199–221. doi:10.1016/S0888-3270(03)00075-X.

[15] He W, Zi Y, Chen B, Wu F, He Z. Automatic fault feature extraction of mechanical anomaly on induction motor bearing using ensemble super-wavelet transform. Mech Syst Signal Process 2015;54-55:457–80. doi:10.1016/j.ymssp.2014.09.007.

[16] Wu S-D, Wu P-H, Wu C-W, Ding J-J, Wang C-C. Bearing Fault Diagnosis Based on Multiscale Permutation Entropy and Support Vector Machine. Entropy 2012;14:1343–56. doi:10.3390/e14081343.

[17] Emmanouilidis C, Hunter A, MacIntyre J, Cox C. Selecting features in neurofuzzy modelling by

multiobjective genetic algorithms. Artif Neural Networks, 1999 ICANN 99 Ninth Int Conf (Conf Publ No 470) 1999;2:749–54 vol.2. doi:10.1049/cp:19991201.

[18] Benkedjouh T, Medjaher K, Zerhouni N, Rechak S. Remaining useful life estimation based on nonlinear feature reduction and support vector regression. Eng Appl Artif Intell 2013;26:1751–60. doi:10.1016/j.engappai.2013.02.006.

[19] Gowid S, Dixon R, Ghani S. A novel robust automated FFT-based segmentation and features selection algorithm for acoustic emission condition based monitoring systems. Appl Acoust 2015;88:66–74. doi:10.1016/j.apacoust.2014.08.007.

[20] Yang Y, Liao Y, Meng G, Lee J. A hybrid feature selection scheme for unsupervised learning and its application in bearing fault diagnosis. Expert Syst Appl 2011;38:11311–20. doi:http://dx.doi.org/10.1016/j.eswa.2011.02.181.

[21] Bolón-Canedo V, Sánchez-Maroño N, Alonso-Betanzos A. Recent advances and emerging challenges of feature selection in the context of big data. Knowledge-Based Syst 2015;86:33–45. doi:10.1016/j.knosys.2015.05.014.

[22] Na M, Sim YR, Park KH, Upadhyaya BR, Lu B, Zhao K. Failure detection using a fuzzy neural network with an automatic input selection algorithm. Intell Hybrid Syst Fuzzy Logic, Neural Network, Genet Algorithms 2002.

[23] Buckner M, Gribok A, Urmanov A, Hines JW. Application of generalized ridge regression for nuclear power plant sensor calibration monitoring. Proceeding Meet. 5th Int. Conf. Fuzzy Log. Intell. Technol. Nucl. Sci., 2002, p. 16–8.

[24] Verikas A, Bacauskiene M. Feature selection with neural networks. Pattern Recognit Lett 2002;23:1323–35. doi:10.1016/S0167-8655(02)00081-8.

[25] Seker S, Ayaz E. Feature extraction related to bearing damage in electric motors by wavelet analysis. J Franklin Inst 2003;340:125–34. doi:10.1016/S0016-0032(03)00015-2.

[26] Saeys Y, Inza I, Larrañaga P. A review of feature selection techniques in bioinformatics. Bioinformatics 2007;23:2507–17. doi:10.1093/bioinformatics/btm344.

[27] Guyon I, Guyon I, Elisseeff A, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res 2003;3:1157–82. doi:10.1162/153244303322753616.

[28] Dy JG, Brodley CE. Feature Selection for Unsupervised Learning. J Mach Learn Res 2004;5:845–89.

[29] Zhang Z, Chen H, Xu Y, Zhong J, Lv N, Chen S. Multisensor-based real-time quality monitoring by means of feature extraction, selection and modeling for Al alloy in arc welding. Mech Syst Signal Process 2015;60-61:151–65. doi:10.1016/j.ymssp.2014.12.021.

[30] Morais DC, De Almeida AT. Group decision making on water resources based on analysis of individual rankings. Omega 2012;40:42–52. doi:10.1016/j.omega.2011.03.005.

[31] Selak L, Butala P, Sluga A. Condition monitoring and fault diagnostics for hydropower plants. Comput Ind 2014;65:924–36. doi:10.1016/j.compind.2014.02.006.

[32] Liu X, Bo L, Luo H. Bearing faults diagnostics based on hybrid LS-SVM and EMD method. Measurement 2015;59:145–66. doi:10.1016/j.measurement.2014.09.037.

[33] Widodo A, Yang B-S. Support vector machine in machine condition monitoring and fault diagnosis. Mech Syst Signal Process 2007;21:2560–74. doi:10.1016/j.ymssp.2006.12.007.

[34] Guo B, Menon J, Willette B. Surface Reconstruction Using Alpha Shapes. Comput Graph Forum 1997;16:177–90. doi:10.1111/1467-8659.00178.

[35] Xu X, Harada K. Automatic surface reconstruction with alpha-shape method. Vis Comput 2003;19:431–43. doi:10.1007/s00371-003-0207-1.

[36] Dyer KB, Capo R, Polikar R. COMPOSE: A Semisupervised Learning Framework for Initially Labeled Nonstationary Streaming Data. IEEE Trans Neural Networks Learn Syst 2014;25:12–26. doi:10.1109/TNNLS.2013.2277712.

[37] Hu Y, Baraldi P, Di Maio F, Zio E. Fault Diagnostics in an Evolving Environment by COMPacted Object Sample Extraction (COMPOSE) algorithm. Neural Networks Learn Syst IEEE Trans 2015:under review.

[38] Wu T-F, Lin C-J, Weng RC. Probability Estimates for Multi-class Classification by Pairwise Coupling. J Mach Learn Res 2004;5:975–1005. doi:10.1016/j.visres.2004.04.006.

[39] Kohavi R, Kohavi R. Wrappers for feature subset selection. Artif Intell 1997;97:273–324. doi:10.1016/S0004-3702(97)00043-X.

[40] Zio E, Baraldi P, Pedroni N. Selecting features for nuclear transients classification by means of genetic algorithms. IEEE Trans Nucl Sci 2006;53:1479–93. doi:10.1109/TNS.2006.873868.

[41] Richard MD, Lippmann RP. Neural Network Classifiers Estimate Bayesian a posteriori Probabilities. Neural Comput 1991;3:461–83. doi:10.1162/neco.1991.3.4.461.

[42] Wan EA. Neural network classification: A Bayesian interpretation. IEEE Trans Neural Networks 1990;1:303–5. doi:10.1109/72.80269.

[43] Rousseeuw PJ. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. J Comput Appl Math 1987;20:53–65. doi:10.1016/0377-0427(87)90125-7.

[44] Forman E, Peniwati K. Aggregating individual judgments and priorities with the analytic hierarchy process. Eur J Oper Res 1998;108:165–9. doi:10.1016/S0377-2217(97)00244-0.

[45] Matsatsinis NF, Grigoroudis E, Samaras A. Aggregation and disaggregation of preferences for collective decision-making. Gr Decis Negot 2005;14:217–32. doi:10.1007/s10726-005-7443-x.

[46] Smith JH. Aggregation Preferences with Variable Electorate. Econometrica 1973;41:1027–41.

[47] Saari DG. Explaining All Three-Alternative Voting Outcomes. J Econ Theory 1999;87:313–55. doi:10.1006/jeth.1999.2541.

[48] Chang C, Lin C. LIBSVM: a library for support vector machines. Computer (Long Beach Calif) 2001:1–30.

[49] Ghadimi AA, Razavi F, Mohammadian B. Determining optimum location and capacity for micro hydropower plants in Lorestan province in Iran. Renew Sustain Energy Rev 2011;15:4125–31. doi:10.1016/j.rser.2011.07.003.

[50] Zimroz R, Bartelmus W, Barszcz T, Urbanek J. Diagnostics of bearings in presence of strong operating conditions non-stationarity - A procedure of load-dependent features processing with application to wind turbine bearings. Mech Syst Signal Process 2014;46:16–27. doi:10.1016/j.ymssp.2013.09.010.

[51] Bearing Data Center. Http://csegroups.case.edu/bearingdatacenter/pages/welcome-Case-Western-Reserve-University-Bearing-Data-Center-Website 2015. http://csegroups.case.edu/bearingdatacenter/pages/welcome-case-western-reserve-university-bearing-data-center-website (accessed May 6, 2015).

[52] Zhao Z, Liu H. Spectral feature selection for supervised and unsupervised learning. Proc 24th Int Conf Mach Learn - ICML '07 2007:1151–7. doi:10.1145/1273496.1273641.

[53] Baraldi; P, Cannarile; F, Di Maio; F, Zio; E. Hierarchical k-nearest neighbours classification and binary differential evolution for fault diagnostics of automotive bearings operating under variable. Reliab Eng Syst Saf 2015;-:-.

Appendix 1: List of features

Feature number	Feature name
2	Mean value
3	Kurtosis
25	Crest indicator
26	Clearance indicator
29	Peak value
34	Minimum Haar Wavelet coefficient
40	Maximum Haar Wavelet coefficient
43	Norm level D1 Daubechies Wavelet Transform
44	Norm Node 1 Symlet6 Wavelet
48	Norm Node 5 Symlet6 Wavelet
49	Norm Node 6 Symlet6 Wavelet
60	Norm Node 3 Symlet6 Wavelet
80	Norm Node 2 Symlet6 Wavelet
84	Norm Node 13 Symlet6 Wavelet
86	Norm Node 15 Symlet6 Wavelet

Appendix 2: The compose algorithm

The sketch of COMPOSE is shown in Figure 7~~Figure 7.~~.

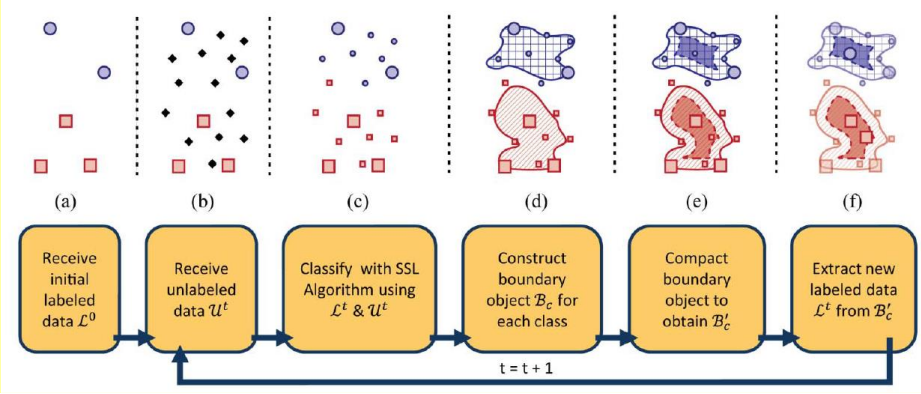

Figure 7 Sketch of COMPOSE algorithm [36]

~~The details of the procedure are as follows:~~

Assuming that a fault classifier has been already trained using the ~~labeled~~labelled data and that a concept drift has been identified in the test set, the COMPOSE allows ~~method that we propose for the construction~~ building a new training set,

which will be used for updating the diagnostic model. The COMPOSE procedure is based ~~of a new training set is based~~ on the following steps:

a) At t = 0, a classifier is trained ~~COMPOSE is provided~~by using ~~with~~ (possibly very few) labelled data, depicted in Figure 7 by ~~opposing classes of~~the (red) squares and (blue) circles;

b) A few unlabelled data, represented in Figure 7 by (black) diamonds are received;

c) ~~Label~~ Tthe unlabelled data (black diamonds) are~~by the fault~~ classifie~~d~~~~r~~ using the classifier built in ~~trained by labelled data collected from~~ step a);

d) the α shape surface reconstruction is applied to find the surface boundary of each class;

e) the core regions of each class are identified by applying a proper shrinkage to the obtained class surface boundaries; the shrinkage is achieved by removing the patterns which are on the surface of the α shape, the degree of shrinkage is controlled by a parameter (the details of the shrinkage procedures can be found in [17]);.

f) the new training set is formed by all the labelled patterns in the core regions identified in e).

Once the new training set has been obtained, it is used to train a new classifier which substitutes the old one. The procedure is entirely repeated each time a new concept drift is detected in a new batch of unlabelled patterns. If new patterns are coming, go back to step 2), otherwise stop the algorithm. More details on the algorithm can be found in [reference POLIKAR and REFERENCE our work to be pubblished]

Formattato: Evidenziato