A Semi-Supervised Feature Selection Approach for Fault Diagnostics in Evolving Environments

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

[1] *Department of Energy, Politecnico di Milano, via la Masa 34, 20156, Milan, Italy*

yang.hu@polimi.it
piero.baraldi@polimi.it
francesco.dimaio@polimi.it

[2]*Chair System Science and the Energy Challenge, Fondation Electricité de France (EDF), CentraleSupélec, Université Paris Saclay, Grande Voie des Vignes, 92290 Chatenay-Malabry, France*

enrico.zio@polimi.it

ABSTRACT

This paper introduces a Semi-Supervised Feature Selection (SSFS) approach for selecting the most suitable features for fault diagnostics in evolving environments. The effectiveness of the proposed SSFS approach is verified with respect to an application concerning the classification of the defect type of bearings in Fully Electric Vehicles operating at different loads. The results show that SSFS allows adapting the diagnostic model to the varying load by updating the set of features used for the classification and achieves more satisfactory diagnostic accuracy than the traditional diagnostic models. The proposed diagnostic approach can contribute significantly to the maintenance practice of components such as gearboxes, alternators, shafts and pumps, whose working conditions are usually characterized by evolving environment.

1. INTRODUCTION

Most industrial components are worked in Evolving Environments (EE) characterized by continuous or periodic changes in the operating conditions. Performing fault diagnostics in EE is very challenging, since the datasets used for training the diagnostic models do not fully cover all the possible environmental conditions that the components experience during their whole life (Nandi, Toliyat, & Li, 2005; Peng & Chu, 2004; Zio, 2016)

In this work, we consider the problem of selecting and updating the set of features most suitable for building diagnostic models in evolving environments. The purpose of feature selection is to reduce the number of features used in input to a diagnostic model to reduce its complexity and improve its performance (Emmanouilidis, Hunter, MacIntyre, & Cox, 1999). Feature selection algorithms are typically based on a procedure for searching a feature set in the space of all possible combinations of features and on the evaluation of its expected diagnostic performance. Filter feature selection approaches evaluate the feature set considering statistical properties of the data, whereas wrapper approaches are based on the construction of a classifier trained by the selected features and on the evaluation of its performance (Guyon, Guyon, Elisseeff, & Elisseeff, 2003; Saeys, Inza, & Larrañaga, 2007). Filter approaches are computationally simpler, faster, and easier to implement, but, since they neglect the dependencies between the feature sets and the classification models, they typically have less satisfactory performances than wrapper approaches. However, wrapper approaches strongly depend on the classification algorithm, i.e. the selected feature set may not be optimal when a different classification algorithm is used. In addition, the computational burden of wrapper approaches is significantly higher (Dy & Brodley, 2004; Zhang et al., 2015).

Typically, both filter and wrapper approaches are applied off-line, using data characterizing the component behavior in a static environment, and the selected features are never changed during the on-line application of the diagnostic model. However, the diagnostic performance of the priori selected feature set can deteriorate, since the capability of a feature to provide useful diagnostic information may depend on the working and environmental conditions experienced by the component.

To overcome this problem, we propose a novel Semi-Supervised Feature Selection (SSFS) approach. The main idea is to assess the performance of a feature set in EEs by considering two indicators: (A) the accuracy and precision of a Support Vector Machine (SVM) classifier trained using labelled data collected from a static environment, and (B) the confidence of the same classifier when used to classify unlabeled data collected from EEs. Indicator A is a traditional performance indicator quantifying the classification ability of the feature set, whereas indicator B assesses the performance of the feature set by estimating how confident is the classifier when used outside its training domain on unlabeled data collected from an EE. A Borda Count method (Morais & De Almeida, 2012) is used to perform a multi-objective ranking of all the feature sets and, thus, to identify the one with the most satisfactory trade-off among indicators A and B.

The proposed SSFS approach is verified with respect to a dataset containing the results of laboratory tests on defective bearings performed within the FP7 European Project HEMIS (Electrical powertrain Health Monitoring for Increased Safety of FEVs). The data refer to six different types of defects and six different working loads.

The paper is organized as follows: Section 2 presents the problem addressed in this work; in Section 3, the SSFS approach is described; in Section 4, the experimental test setup and the application of the developed method to the experimental data is discussed; finally, some conclusions and remarks are drawn in Section 5.

2. FEATURE SELECTION IN EVOLVING ENVIRONMENTS

The overall target of feature selection in fault diagnostic applications is to select the feature subset that allows building the diagnostic model with the most satisfactory performance. Assuming to have available an overall feature set $TFS = \{F_1, F_2, \dots, F_{N_{\mathrm{FS}}}\}$ of N_{FS} features, there are totally $2^{N_{\mathrm{FS}}} - 1$ possible candidate feature subsets, FS_i, $FS_i \subset \mathrm{TFS}$.

In the case of EE, the following information are typically available to perform feature selection:

- a set of N_T labelled data, $T = \{X_T, L_T\}$. X_T is the signal value matrix with $N_T * N_{\mathrm{FS}}$ dimension, N_T is the total number of patterns in X_T. L_T is a binary matrix of dimension $N_T * N_{cl}$, with N_{cl} representing the total number of fault classes, whose generic element j,k is equal to 1 if the *j*-th pattern is of class *k* and 0, otherwise.

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

  These labelled data are assumed to be collected in stationary working conditions, before the occurrence of a concept drift.

- a set of N_C unlabelled data, $C = \{X_C\}$, where X_C is a matrix of dimension $N_C * N_{\mathrm{FS}}$ containing the signal values. These data describe the component behaviour in evolving environments.

3. THE SEMI-SUPERVISED (SSFS) FEATURE SELECTION APPROACH

The main idea of SSFS is to evaluate the performance of each candidate feature subset, FS_i, $FS_i \subset \mathrm{TFS}$, by calculating the following two indicators:

- Indicator A: the classification accuracy of a SVM classifier on the labelled data T.
- Indicator B: the confidence of the SVM classifier in the assignment of the unlabelled data C collected in the new environment.

Indicator A quantifies the capability of the feature set of correctly classifying test data in stationary working conditions. Notice, however, that a satisfactory value of indicator A does not automatically guarantee a high accuracy of the diagnostic model in an EE. Thus, in order to quantify the performance of the feature set in a new environment, we introduce indicator B. Given the unavailability of the true labels of the patterns in X_C, indicators B focuses on the evaluation of the confidence of the classifications provided by the SVM.

Sections 3.1 and 3.2 below will further discuss the computation of indicators A and B, whereas Section 3.3 shows the Borda Count-based procedure for aggregating the information provided by these two indicators.

3.1. Indicator A

For a generic feature subset FS_i, Indicator A is a measure of the accuracy of a SVM classifier built considering as input signals the features in FS_i and trained using 50% of the labelled data of T. The classification accuracy is evaluated using the remaining 50% of the data of T. The classifier used in this paper is a SVM with pairwise coupling (Wu, Lin, & Weng, 2004), which provides in output the probabilities p_{jk} that the *j*-th test pattern belongs to class *k*, *k*=1,…, N_{cl}. Assuming to have available N_{te} labelled test patterns, indicator A is defined by:

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

IA_{FS_i} is the value of indicator A of candidate feature set *i*, h_{jk} is the corresponding element in L_T; in order to obtain a robust evaluation of the accuracy, 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 equally sized subsets: the first one is used to train the

SVM classifier and the second one to compute its accuracy. *IA* is then computed as the average of the 10 runs:

$$IA_{FS_i} = \frac{1}{10}\sum_{m=1}^{10} IA_{FS_i}(m) \tag{2}$$

The value of this indicator is between 0 (all patterns misclassified) and 1 (all patterns correctly classified). The larger IA, the more accurate is the classifier built with the feature subset FS_i.

3.2. Indicator B

Indicator B measures to what extent the SVM classifier built using the labelled data T is able to provide confident classifications of the unlabelled data C in the new environment. According to (Richard & Lippmann, 1991; Wan, 1990), the confidence of a classifier can be evaluated by considering the entropy:

$$E = \sum_{j=1}^{N_C}\sum_{k=1}^{N_{cl}} -p_{jk} \cdot \log p_{jk} \tag{3}$$

E 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, rather than the entropy itself, the use of the entropy measure can have limitations. For example, let us consider a case of two classifiers, O_1 and O_2, which assign the same test pattern 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 equation (3), the classification O_2 is considered more confident than O_1 , ($E_2 = 0.72 < E_1 = 0.95$). However, from the point of view of the decision maker in a fault diagnostic problem, 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 output of classifier O_1. This is due to fact that the second most probable class is assigned by classifier $O_1$with a lower probability value than that by classifier O_2. In order to overtake this limitation of the entropy metric, in this work we propose a new confidence metric, from the decision making point of view, based on the evaluation of the difference between the probabilities of the class of the maximum probability and that with the second maximum probability. Thus, indicator B is defined by:

$$IB_{FS_i} = \sum_{j=1}^{N_C} \left|\lambda_j - \mu_j\right| \tag{4}$$

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

3.3. Borda count Method

Once indicators A and B have been computed for all the feature sets of interest, we need to select the feature set to be used for the fault diagnostics in the new environment. This is a group decision-making process which involves aggregating the information from multiple sources (Forman & Peniwati, 1998; Matsatsinis, Grigoroudis, & Samaras, 2005). The aggregation problem is here addressed using the Borda count method, which has been successfully applied in very different application fields (Saari, 1999; Smith, 1973). Borda count is a single-winner vote method, which ranks candidates according to the sum of ballots from all the voters. The detailed procedure is based on the following steps:

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

2) **Voting:** for any indicator i, two scores F^i_{upper} and F^i_{lower} are assigned to each candidate feature set. With respect to F^i_{upper}, assuming that there are *x* candidate feature sets, the mark 1 is assigned to the feature set 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^i_{lower} is assigned to the feature sets: the mark *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. The final F_{upper} (F_{lower}) value associated to a feature set is the sum of all the scores F^i_{upper} (F^i_{lower}) on both of the indicators *IA* and *IB*

3) **Choosing:** The final score F_{final} of each candidate feature set is:

$$F_{final} = F_{upper} - F_{lower} \tag{5}$$

The selected feature set is the one with largest F_{final} Once the best performing feature set is selected by the Borda count method, the corresponding SVM classifier will be retrained using all the data in T and used for fault diagnostics in the new environment.

3.4. The SSFS algorithm

Figure 1 shows the sketch of the SSFS. The first step is performed when only one batch of labelled data ($T = \{X_T, L_T\}$), referring to a stationary working condition is available. The data are used for an initial wrapper feature selection whose objective is the maximization of the classification accuracy (Indicator *IA*).

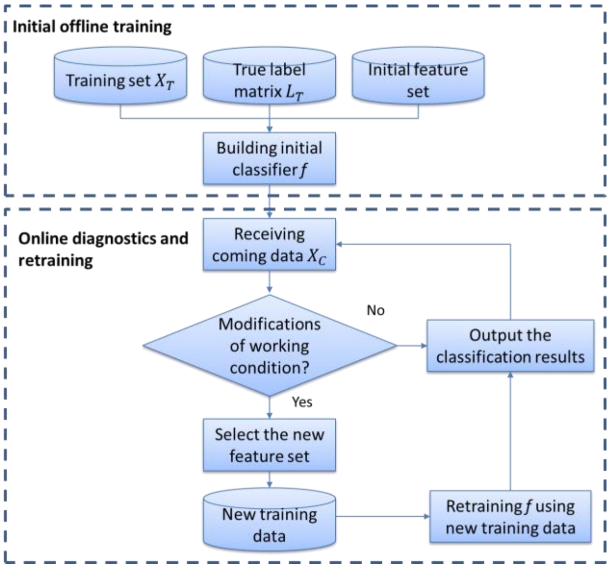

Figure 1. Sketch of the SSFS approach

Once the feature set with the best classification performance on the labelled data is identified, the corresponding SVM classifier f is used until a modification of the working conditions occurs. In this work, we assume that the time at which there is a modification of the working conditions is known. Notice that in the case in which the information on the modification of the working conditions is not explicitly available, one can resort to automatic methods for concept drift detection (Brzezinski & Stefanowski, 2014; Dries & Rückert, 2009).

Each time a new batch of unlabelled data collected from new operational conditions become available, the feature selection algorithm described in Sections 3.2 and 3.3 is applied and a new feature subset is selected. Finally, the corresponding SVM classifier f is trained and can be used for fault diagnostics in new operative conditions.

4. NUMERICAL APPLICATION

In this Section, we apply the SSFS approach to a problem of diagnosing bearing defects in evolving environments. We consider data collected in laboratory tests performed on defective bearings within the FP7 European Project HEMIS (Electrical powertrain Health Monitoring for Increased Safety of FEVs). The tests have been performed on bearings of an electric engine, considering six different types of defects (Table 1). For each defect type, six different working loads have been applied to the bearing during the laboratory tests (Table 2). A raw vibrational signal has been measured at a frequency of 20kHz using an accelerometer. Then, 87 features have been extracted from non-overlapping time windows of the raw signal. The considered features are statistical indicators, such as mean value, kurtosis, peak value etc., and wavelet transform coefficients, such as minimum and maximum Haar wavelet coefficient, symlet wavelet coefficient, etc. Among the 87 extracted features, 15 have been preselected to reduce the computational burden of the analysis using an unsupervised spectral feature selection method (Zhao & Liu, 2007). The list of the considered features is reported in Appendix 1. For each defect type, 84 patterns at 6 different loads have been obtained (Table 1 and 2).

The classification accuracy achieved by the SSFS method is compared with that of a SVM classifier built considering the labelled training set T and never updated (here after named 'pure SVM'). The source code of the SVM classifiers used in this paper is taken from "LIBSVM" (Chang & Lin, 2001).

Six different experiments are designed in order to compare the performance of the proposed SSFS approach with that of the pure SVM. 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 fixed load, different from the load of the previous batch. In all the experiments, a labelled dataset formed by 98 patterns at a prefixed load is initially available, whereas 5 batches formed by unlabelled patterns become progressively available. The six experiments differ in the sequence with which the batches of data become available, as shown in Table 3.

Due to limitations in computation and storage resources, we do not perform an exhaustive search among all the $2^{15}-1$=32767 possible feature subsets of 15 features, but we limit the search to consider only the $\binom{15}{3} = 455$ feature subsets formed by 3 features.

Table 1. Defect types

Defect label (class)	Fault location	Fault intensity	Number of available patterns
1	Inner race	1 mm	84
2	Inner race	1.5 mm	84
3	Inner race	2 mm	84
4	Outer race	1 mm	84
5	Outer race	1.5 mm	84
6	Outer race	2 mm	84
7	Healthy	0 mm	84

Table 2. Loads applied during the laboratory test

Label	load	Rotational speed	Number of available patterns
1	100 Nm	250 Rpm	98
2	100 Nm	300 Rpm	98
3	100 Nm	320 Rpm	98
4	150 Nm	250 Rpm	98
5	150 Nm	300 Rpm	98
6	150 Nm	320 Rpm	98

Table 3. Sequence with which the batches of data become available

	Training dataset	Batch 1	Batch 2	Batch 3	Batch 4	Batch 5
Experiment 1	all patterns in load 1	load 2	load 5	load 6	load 3	load 4
Experiment 2	all patterns in load 2	load 1	load 5	load 6	load 3	load 4
Experiment 3	all patterns in load 3	load 2	load 5	load 6	load 1	load 4
Experiment 4	all patterns in load 4	load 5	load 1	load 2	load 3	load 6
Experiment 5	all patterns in load 5	load 6	load 1	load 2	load 3	load 4
Experiment 6	all patterns in load 6	load 5	load 1	load 2	load 3	load 4

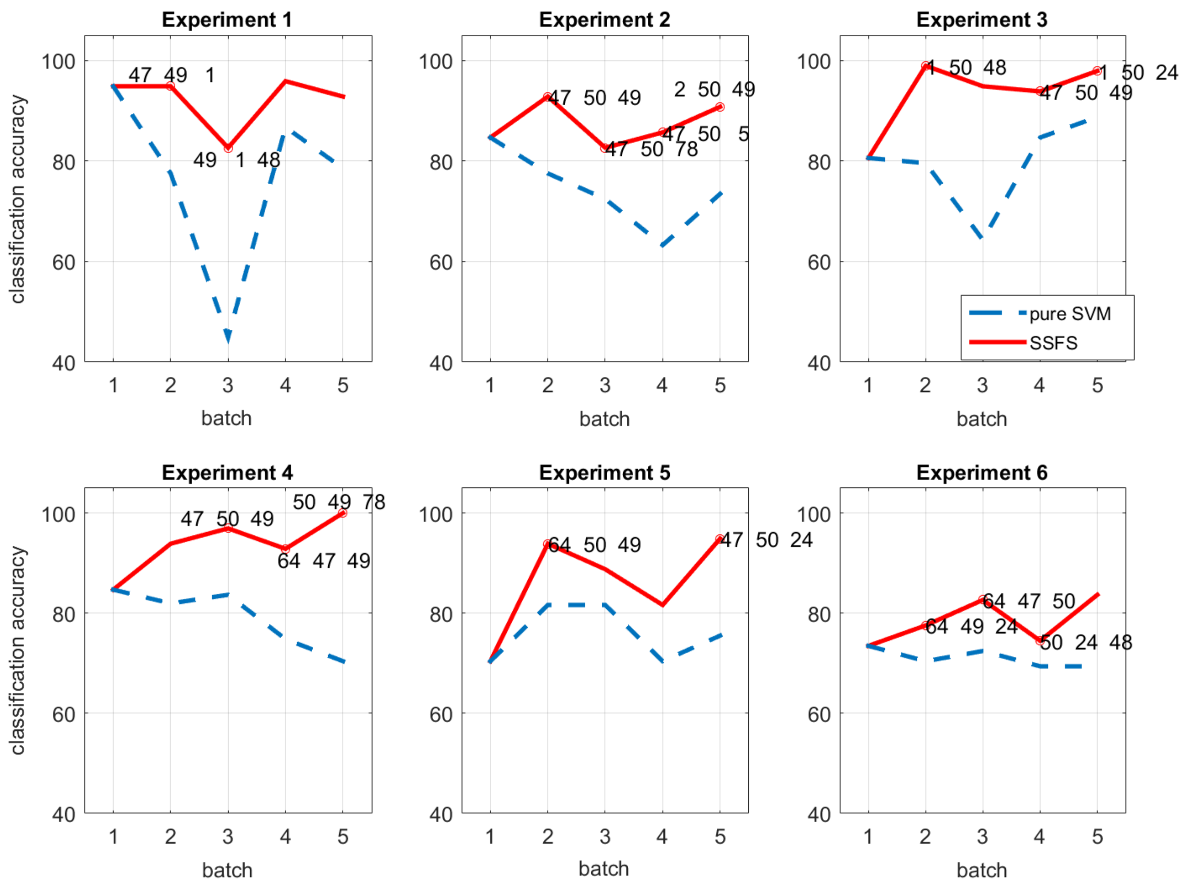

Figure 2. Classification accuracy provided by the pure SVM and the SSFS approaches in the 6 experiments. The numbers in the Figures refer to the Table in the Appendix and indicate the features selected by the SSFS algorithm.

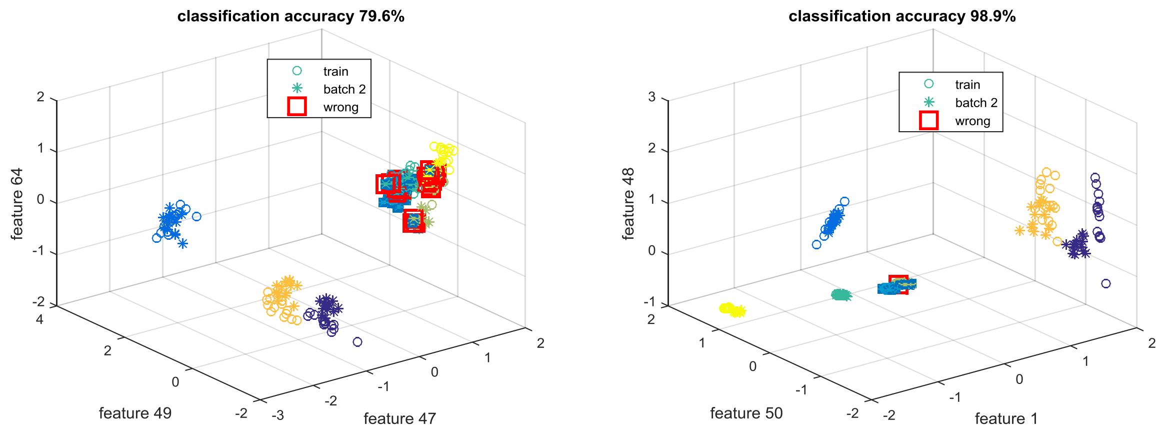

Figure 3. Projection of the training dataset (circles) and of the data in batch 2 (stars) of experiment 3 on the initial feature set {47, 49, 64} used by the pure SVM (left) and on the feature set {1, 48, 50} used by the SSFS method (right). Different colors indicate different classes of the data, red squares represent wrong classifications.

Figure 2 shows the classification accuracy provided by the pure SVM and the SSFS approaches in the 6 experiments. When in correspondence of a new batch of data, the feature set used by the SSFS approach is changed, the new selected feature set is indicated in the Figure. Notice that, thanks to the dynamic updating of the feature set, the SSFS approach is able to provide more accurate predictions than the pure SVM.

Figure 3 shows the projection of the patterns of the second batch of experiment 3 in the feature sets used by the pure SVM, formed by features {47, 49, 64} and by the SSFS approach, formed by features {1, 48, 50}. We can clearly see that patterns of different classes tend to overlap when they are projected on the feature set {47, 49, 64}, whereas they are well separated on the feature set {1, 48, 50}.

5. CONCLUSION

In this work, we propose a Semi-Supervised Feature Selection (SSFS) approach for performing fault diagnostics in evolving environments. SSFS allows adapting the diagnostic model to the evolving environment by automatically changing the feature set used for the classification. The approach has been successfully verified with respect to a problem of classification of bearing defects in different load conditions.

A limitation of the SSFS method lies in the computational efforts required for the feature selection task, since it needs to perform exhaustive search among all the possible feature sets. Search algorithms such as Genetic Algorithms, Differential Evolutions and Ant Colony can be used to select the best feature sets, without the necessity of exploring all the possible solutions.

Future work will also be devoted to improve the efficiency of SSFS by designing a drift detector able to decide when it is necessary to change the feature set.

ACKNOWLEDGEMENT

This research has received funding from the7th European Union Framework Programme (FP7/2007-2013) under grant agreement no.314609. Further information can be found on the project website (www.hemis-eu.org). The authors are grateful for the contributions of Prashanth Dhurjati and Toni Escamilla (IDIADA, Spain), and Daniel Astigarraga and Ainhoa Galarza (CEIT, Spain) in the design and execution of the laboratory tests and their suggestions during the development of this research. The participation of Yang Hu is supported from China Scholarship Council (No. 201206110018). The participation of Enrico Zio to this research is partially supported by the China NSFC under grant number 71231001.

REFERENCES

Brzezinski, D., & Stefanowski, J. (2014). Reacting to Different Types of Concept Drift: The Accuracy Updated Ensemble Algorithm. *Neural Networks and Learning Systems, IEEE Transactions on*. doi:10.1109/TNNLS.2013.2251352

Chang, C., & Lin, C. (2001). LIBSVM: a library for support vector machines. *Computer*, 1–30. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.114.4532&rep=rep1&type=pdf

Dries, A., & Rückert, U. (2009). Adaptive concept drift detection. *Statistical Analysis and Data Mining*, *2*(5-6), 311–327. doi:10.1002/sam.10054

Dy, J. G., & Brodley, C. E. (2004). Feature Selection for Unsupervised Learning. *Journal of Machine Learning Research*, *5*, 845–889. Retrieved from http://portal.acm.org/citation.cfm?id=1016787

Emmanouilidis, C., Hunter, A., MacIntyre, J., & Cox, C.

(1999). Selecting features in neurofuzzy modelling by multiobjective genetic algorithms. *Artificial Neural Networks, 1999. ICANN 99. Ninth International Conference on (Conf. Publ. No. 470)*. doi:10.1049/cp:19991201

Forman, E., & Peniwati, K. (1998). Aggregating individual judgments and priorities with the analytic hierarchy process. *European Journal of Operational Research*. doi:10.1016/S0377-2217(97)00244-0

Guyon, I., Guyon, I., Elisseeff, A., & Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*, 1157–1182. doi:10.1162/153244303322753616

Matsatsinis, N. F., Grigoroudis, E., & Samaras, A. (2005). Aggregation and disaggregation of preferences for collective decision-making. *Group Decision and Negotiation*, *14*(3), 217–232. doi:10.1007/s10726-005-7443-x

Morais, D. C., & De Almeida, A. T. (2012). Group decision making on water resources based on analysis of individual rankings. *Omega*, *40*(1), 42–52. doi:10.1016/j.omega.2011.03.005

Nandi, S., Toliyat, H. A., & Li, X. (2005). Condition monitoring and fault diagnosis of electrical motors - A review. *IEEE Transactions on Energy Conversion*. doi:10.1109/TEC.2005.847955

Peng, Z. K., & Chu, F. L. (2004). Application of the wavelet transform in machine condition monitoring and fault diagnostics: a review with bibliography. *Mechanical Systems and Signal Processing*. doi:10.1016/S0888-3270(03)00075-X

Richard, M. D., & Lippmann, R. P. (1991). Neural Network Classifiers Estimate Bayesian a posteriori Probabilities. *Neural Computation*, *3*(4), 461–483. doi:10.1162/neco.1991.3.4.461

Saari, D. G. (1999). Explaining All Three-Alternative Voting Outcomes. *Journal of Economic Theory*, *87*, 313–355. doi:10.1006/jeth.1999.2541

Saeys, Y., Inza, I., & Larrañaga, P. (2007). A review of feature selection techniques in bioinformatics. *Bioinformatics*. doi:10.1093/bioinformatics/btm344

Smith, J. H. (1973). Aggregation Preferences with Variable Electorate. *Econometrica*, *41*(6), 1027–1041.

Wan, E. A. (1990). Neural network classification: A Bayesian interpretation. *IEEE Transactions on Neural Networks*, *1*(4), 303–305. doi:10.1109/72.80269

Wu, T.-F., Lin, C.-J., & Weng, R. C. (2004). Probability Estimates for Multi-class Classification by Pairwise Coupling. *J. Mach. Learn. Res.*, *5*, 975–1005. doi:10.1016/j.visres.2004.04.006

Zhang, Z., Chen, H., Xu, Y., Zhong, J., Lv, N., & Chen, S. (2015). Multisensor-based real-time quality monitoring by means of feature extraction, selection and modeling for Al alloy in arc welding. *Mechanical Systems and Signal Processing*, *60-61*, 151–165. doi:10.1016/j.ymssp.2014.12.021

Zhao, Z., & Liu, H. (2007). Spectral feature selection for supervised and unsupervised learning. *Proceedings of the 24th International Conference on Machine Learning - ICML '07*, 1151–1157. doi:10.1145/1273496.1273641

Zio, E. (2016). Some Challenges and Opportunities in Reliability Engineering. *IEEE Transactions on Reliability*, -(-), -.

Appendix 1: List of features

Feature number	Feature name
1	Mean value
2	Kurtosis
5	Crest indicator
24	Clearance indicator
47	Peak value
48	Minimum Haar Wavelet coefficient
49	Maximum Haar Wavelet coefficient
50	Norm level D1 Daubechies Wavelet Transform

64	Norm Node 1 Symlet6 Wavelet
48	Norm Node 5 Symlet6 Wavelet
78	Norm Node 6 Symlet6 Wavelet
80	Norm Node 3 Symlet6 Wavelet
81	Norm Node 2 Symlet6 Wavelet
84	Norm Node 13 Symlet6 Wavelet
86	Norm Node 15 Symlet6 Wavelet