Condition monitoring with Mean Field Independent Components Analysis

Niels Henrik Pontoppidan, Sigurdur Sigurdsson and Jan Larsen

Informatics and Mathematical Modelling - IMM
Technical University of Denmark
Richard Petersens Plads - Building 321
DK-2800 Kongens Lyngby - Denmark
www.imm.dtu.dk

Abstract

We discuss condition monitoring based on mean field independent components analysis of acoustic emission energy signals. Within this framework, it is possible to formulate a generative model that explains the sources, their mixing and the noise statistics of the observed signals. Using a novelty detection approach based on normal condition examples only, we detect faulty examples with high precision. The detection is done by evaluating the likelihood that the model, trained with normal examples, generated the signals, compared to a threshold obtained with normal examples. Acoustic emission energy signals from a large diesel engine are used to demonstrate this approach.

The experiment show that mean field independent components analysis detects the induced fault with higher accuracy than principal components analysis, while at the same time selecting a more compact model.

Key words: Mean Field Independent Components Analysis, Condition Monitoring, Unsupervised learning

1 Introduction

In this paper, we apply *mean field independent component analysis* (MFICA) to condition monitoring of a large two-stroke diesel engine. The setup is as follows: from a collection of examples gathered under normal running conditions

Email addresses: nhp@imm.dtu.dk (Niels Henrik Pontoppidan), siggi@imm.dtu.dk (Sigurdur Sigurdsson), jl@imm.dtu.dk (Jan Larsen).

we learn the hidden signals in signal and how they are mixed together. Given a new collection of observed signals we test if the obtained model explains the data equally well as it did with the known normal ones; if not the condition is faulty.

The interesting result is that condition monitoring can be achieved with a single feature by a simple threshold comparison, and further where the model and threshold is derived from normal condition data only. Still we obtain a system capable of detecting several types of faults; however with such hypothesis all faults are just labeled faulty, and not directly identified by the system. This setup is highly relevant as we are facing a problem where specific modeling of specific faults is not economically feasible.

We will introduce MFICA, apply it to a collection of labeled normal and faulty examples, and show how the two classes separate by looking at the underlying hidden signals and the independent components. We compare this to a setup where only normal examples are available for training. It shows that we can still separate the two classes with a model build from normal examples only, while obtaining good detection of faulty examples. With this in mind, we outline our unsupervised method and its results.

1.1 Data setup

Acoustic emission signals were acquired from the two-stroke test bed engine at MAN B&W Diesel in Copenhagen. The signals were processed with Root mean square in time-domain, sampled at 20 kHz, before resampled in the crank angle domain. Finally, they were partitioned such that a single example represents the AEE during a single engine cycle. The engine cycle for a two-stroke engine is one revolution of the crankshaft, so each sample corresponds to the AEE at a certain angular position of the crankshaft and piston. Figure 1 show the engine with piston and crankshaft (left), an AEE "radar-plot" where the AEE amplitude is shown as distance from the center (middle), and finally the AEE signal in the linear crank angle domain (right). Each observation consist of 2048 AEE samples in the crank angle domain from -180^o to 180^o, i.e. one revolution of the crankshaft. The peaks in the AEE signals are the results of engine related events, e.g., the peak around 0° is the combustion and fuel injection operation.

In the crank angle domain all signals containing an engine cycle, have same length regardless of running speed [1]. However, many engine designs, including the MAN B&W Diesel test bed engine in Copenhagen, optimize performance by moving the angular position of certain events as a function of load and speed settings, e.g., by advancing the fuel injection start to inject more

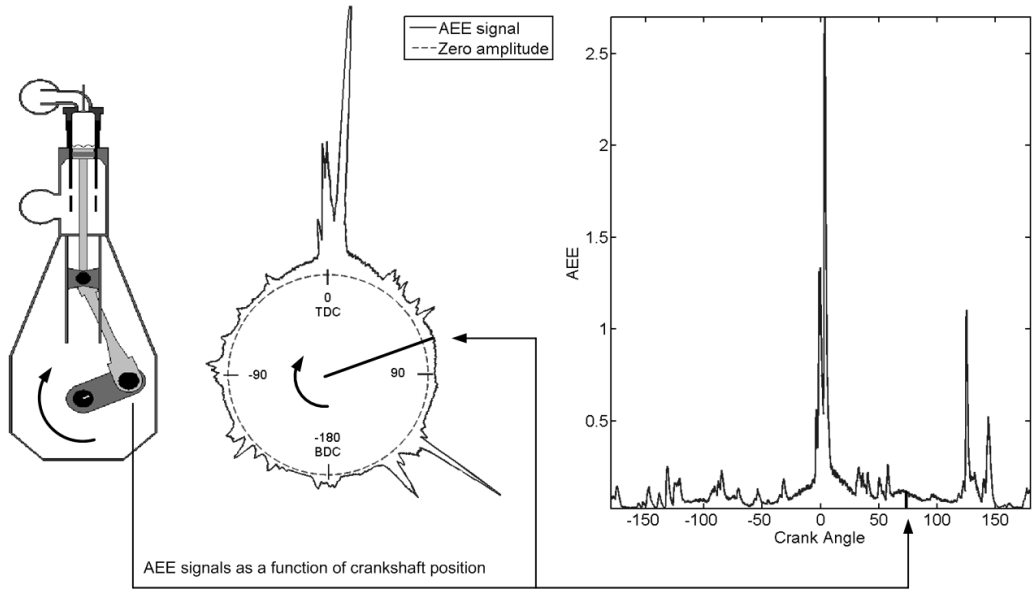

Fig. 1. Crank Angle domain sampling of acoustic emission energy signals. The two arrows point at the data points as a position of the crankshaft wrt. 0° aka. Top Dead Center (TDC) where the combustion takes place.

fuel. In this paper we will focus on a single running condition, and refer to our ongoing research on event alignment [2] for handling the non-stationary case under multiple running conditions.

We denote an observation of the AEE signal in an engine cycle by $\mathbf{x}$ being a $d \times 1$ vector of non-negative elements. From a set of from N such vector cycles we build the training matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$ as seen in Figure 2 to the left. In the blind source separation (BSS) framework, we solve the inverse problem assuming that the training matrix is generated by a linear mixing of K underlying non-negative AEE signals plus white Gaussian noise given by

$$\mathbf{X} = \mathbf{AS} + \mathbf{\Gamma}, \tag{1}$$

where $\mathbf{A}$ is a $d \times K$ mixing matrix containing the hidden AEE signals as columns, $\mathbf{S}$ is $K \times N$ source matrix containing the independent components as rows, and $\mathbf{\Gamma}$ is $d \times N$ noise matrix. With this setup the independent components in the source matrix are not prototype AEE signals, but gain factors holding the amplitude of the corresponding columns of the mixing matrix that contain the hidden AEE signals. As shown in Figure 2 an observation is generated by multiplying the K hidden AEE signals with the corresponding independent components and adding noise, e.g. $\mathbf{x}_{1:d,1} = \mathbf{A}_{1:d,1} \cdot \mathbf{S}_{1,1} + \mathbf{A}_{1:d,2} \cdot \mathbf{S}_{2,1} + \mathbf{\Gamma}_{1:d,1}$

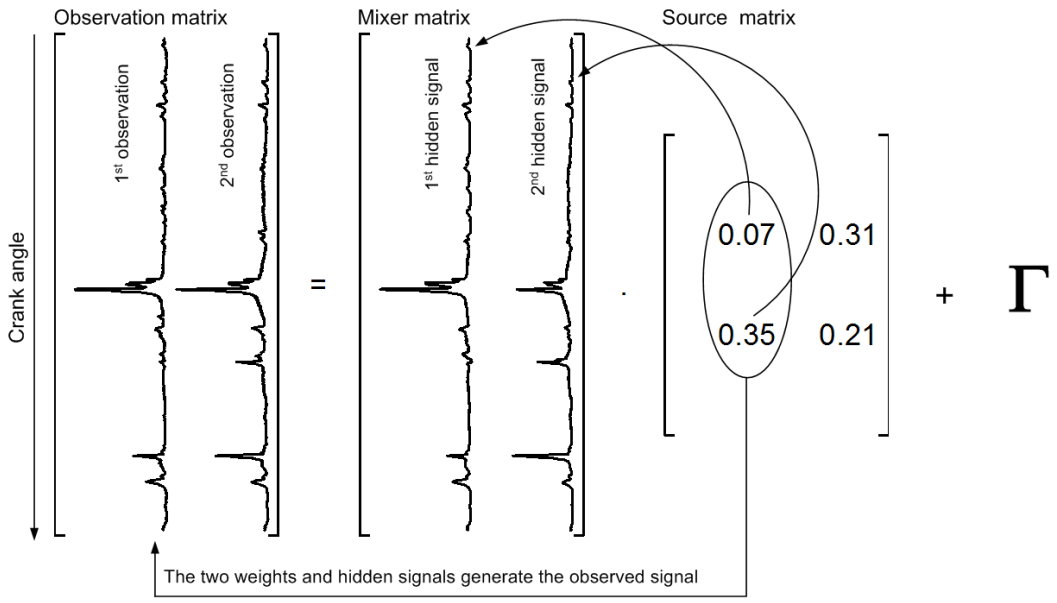

Fig. 2. $\mathbf{X} = \mathbf{AS} + \mathbf{\Gamma}$, Matrix setup for blind source separation with two observations and components. $\mathbf{X}$, $\mathbf{A}$, $\mathbf{S}$ and $\mathbf{\Gamma}$ are observation, mixing, source and noise matrix respectively. Each observation signal $\mathbf{x}$ is generated by mixing the columns in the mixing matrix $\mathbf{A}$ weighted by the corresponding set of gain factors (as a column) in the source matrix $\mathbf{S}$ by the corresponding column in the Gaussian noise matrix $\mathbf{\Gamma}$.

2 Mean field independent components analysis

In a condition monitoring framework using the MFICA, the columns of the mixing matrix $\mathbf{A}$ may be interpreted as underlying AEE signals, generated by specific events. For instance, these sources could be the results of specific engine impacts and scratching. As the AEE signals are inherently nonnegative, it is appropriate to assume that the various energy sources are additive and that interference is also non-negative. This implies that both the elements of the mixing matrix and source matrix are nonnegative.

2.1 Training

Recently, the ICA was extended with a Bayesian framework using an advanced mean field training [3], making it possible to incorporate constraints on the source and mixing matrix. The MFICA accomplishes this by defining an appropriate prior distribution over the sources. Given the noise model in Equation 1, the likelihood for the parameters and sources of the MFICA may be written as

$$p(\mathbf{X}|\mathbf{A}, \mathbf{\Sigma}, \mathbf{S}) = (\det(2\pi\mathbf{\Sigma}))^{-N/2} \exp\left(-\tfrac{1}{2}\mathrm{Tr}\{(\mathbf{X} - \mathbf{AS})^\top \mathbf{\Sigma}^{-1}(\mathbf{X} - \mathbf{AS})\}\right) \quad (2)$$

where noise has zero mean and $\mathbf{\Sigma}$ is the noise covariance matrix. The noise is inevitable positive after RMS, but since the signal to noise ratio is so high with AEE signals, the mismatch due to the zero mean assumption is of negligible size. The aim of MFICA is to estimate the unknown quantities, the sources $\mathbf{S}$, the mixing matrix $\mathbf{A}$ and the noise covariance $\mathbf{\Sigma}$ from the observed data. For the condition-monitoring problem, we need to characterize the unknown parameters. The noise is simplified to an isotropic Gaussian distribution where $\mathbf{\Sigma} = \sigma^2\mathbf{I}$, the sources are assumed to be exponential distributed with the prior distribution $p(\mathbf{S}) = \eta \exp(-\eta\mathbf{S})$ where $\eta > 0$, and the elements of mixing matrix is assured nonnegative by combining Lagrange multipliers to the mean field training. The parameter estimation is done in a Bayesian manner, by integrating out the hidden variable $\mathbf{S}$, i.e.,

$$p(\mathbf{X}|\mathbf{A}, \mathbf{\Sigma}) = \int p(\mathbf{X}|\mathbf{A}, \mathbf{\Sigma}, \mathbf{S}) p(\mathbf{S}) \, d\mathbf{S} \tag{3}$$

and using this new likelihood to optimize the parameters. Unfortunately, this integral is intractable to solve analytically. Instead, equation (3) is approximated using the so-called adaptive Thouless-Anderson-Palmer mean field approach [4]. For details on the MFICA we refer to [3] and also to available Matlab toolbox [5].

While the parameters may be estimated with MFICA, we still need to determine the number of components K. This corresponds to a model selection problem where we are interested in finding a model that fits the data well in the face of limited data, i.e., has good generalization capabilities on unseen data. If the K is selected too small compared to the optimal K, we get a too simple model that does not capture the underlying function generating data, i.e., the sources and mixing matrix. On the other hand, selecting a K that is too large gives a too complex model that fits to the additive noise. Various methods for model selection have been proposed, e.g., empirically with cross-validation resampling schemes [6] that we use here and algebraically, e.g. using Bayesian information criterion [7].

2.2 Using the trained model on a new example

Given a new example $\mathbf{x}$ and a model defined by $\mathbf{A}$ and $\mathbf{\Sigma}$ we redo the part of the inverse problem in Equation 1 where $\mathbf{S}$ is estimated whilst keeping the mixing matrix $\mathbf{A}$ and noise covariance matrix $\mathbf{\Sigma}$ fixed. Due the constraints on $\mathbf{s}$ the solution is obtained using the same mean field optimizer that was used for the training. The output of this optimization is the components $\mathbf{s}$ and the corresponding negative log likelihood (NLL) $-\log p(\mathbf{x}|\mathbf{A}, \mathbf{s}, \mathbf{\Sigma})$. We have previously showed[8] how the log likelihood (NB not negative) dropped significantly just after a condition change, and further how it regained its usual

level after a temporary fault in the water brake disappeared. Effectively for classification purposes the dimensionality is reduced from 2048 (original data dimension) to one (the NLL) with MFICA.

2.3 Classification with the negative log likelihood

The MFICA does a good job on separating the normal and faulty examples in the NLL-domain. A simple but effective classification method is to set an NLL threshold to separate the faulty examples from the normal. This is related to novelty detection [9]. With another set of known normal examples the cumulated density of the NLL feature is estimated, i.e., the NLL given $\mathbf{A}$ and $\mathbf{\Sigma}$ is obtained. In this cumulated density model an inherent rejection rate of say 5% is selected, thus we already know that we will face 5% false alarms. Even with an inherent rejection rate of 0% we could still face false alarms, since the threshold is based on a sample set. The false alarms can be lowered using sliding windows and binomial hypothesis test on the classification outputs in the window against the selected rejection rate (see further [10]).

3 Comparing two and one class training

In this section, we will compare the results obtained when solving two different BSS problems. We take 140 known normal and 140 known faulty examples and split them into 2 sets of 40 and 100 examples for training and testing respectively. We will call the first problem *two class* as the observation matrix is build from the 80 labeled training examples. In the second problem, the *one class*, we only use the 40 normal examples for training. We will solve the *two class* problem assuming two independent components and compare this result with the result that we obtain when solving the *one class* problem assuming only a single component. It is expected that one of the columns in the *two class* mixing matrix should resemble the column in the *one class* mixing matrix, and that the two classes separate in component domain. In **??** the left panel show the 200 labeled test examples that are separated in two classes, while the right panels show the two columns of the mixing matrix (1st and 2nd hidden signal). As also indicated in Figure 2 the 1st hidden signal contains the additional AEE signal encountered under the faulty condition, while the 2nd hidden signal models the normal condition. The two hidden signals from the *two class* problem are repeated in Figure 4 to the right, where we see that the second column resembles the column of the *one class* mixing matrix shown above the two. Furthermore the left panels of Figure 4, that display the estimated density and cumulated density of the 200 labeled test points, show that the normal and faulty examples also separate in the single component

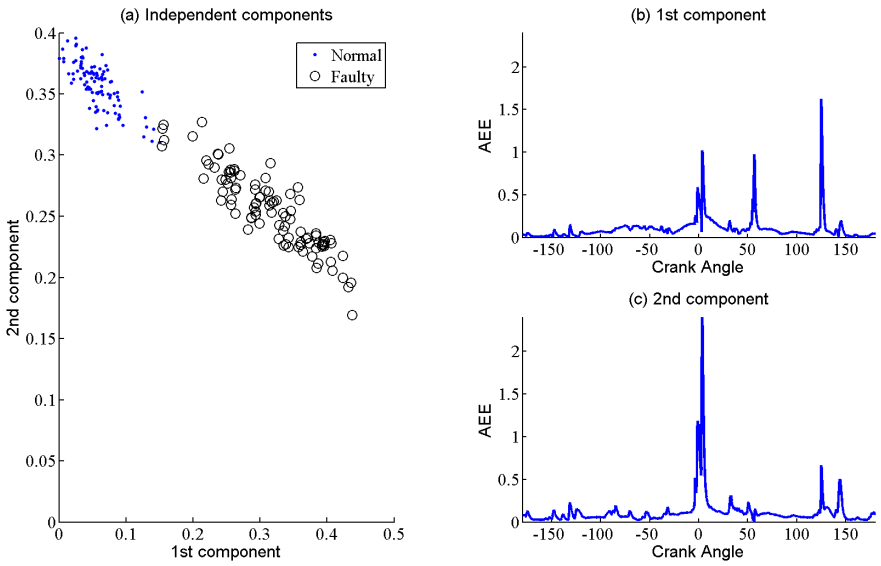

Fig. 3. Separation with *two class* model. The normal and faulty examples separate in source domain; where the faulty examples tend to contain more of the 1st hidden signal. Further we observe that the 2nd hidden signal resembles the single component obtained from normal conditions only in Figure 4.

domain from the *one class* problem – although not as good as in the *two class* problem.

4 Unsupervised condition monitoring

Due to economic figures, the supervised setup is not an option for our application. It is simply too expensive to conduct repeated experiments for a wide range of faults and engines. We therefore aim for an unsupervised setup where the system models the normal condition and detects deviations from that. The setup can later be turned into a semi-supervised setup, with possibility of identifying specific faults by allowing new models trained on data with those faults to compete with the normal condition model.

In the *one class* example we assumed one independent component, but as the following results shows, more than one component yields better classification performance, due to the ability to model the modes of variation in the normal condition. We have recently followed that idea, and trained an unsupervised model on a collection of examples acquired under multiple normal load settings. However it turned out that the performance was inferior to our event alignment method[10].

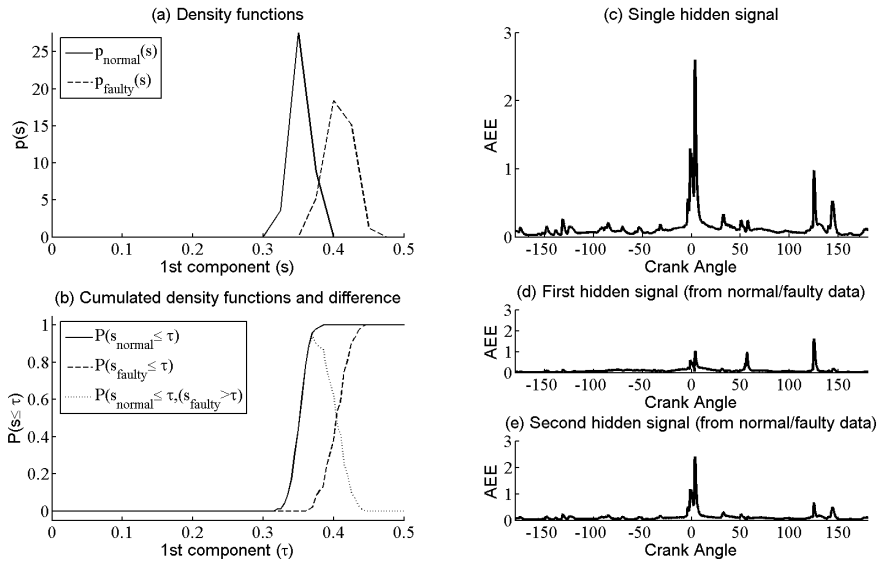

Fig. 4. Separation with *one class* model. The observation matrix consist of normal examples only, this is one class, single component MFICA. The two panels to the left show the empirical densities and cumulated densities for the sources of normal and faulty data. The lower panel show that $s \leq 0.36$ for 95% of the normal examples whilst $s > 0.36$ for virtually all faulty examples. To three panels to the right show the single hidden signal from the "unsupervised" mixing matrix, that can be compared to the two hidden signals obtained from the *two class* setup in Figure 3. Clearly the second hidden signal resembles the single hidden signal better than the first thereby being the "normal" hidden signal.

4.1 Experimental setup

The experiment was conducted by acquiring the AEE signals before and after a fault-condition was induced by cutting off the supply of lubricating oil to the monitored cylinder. This resulted in increased friction and wear that possibly could lead to a severe fault called scuffing. In this paper we only used the first two hours of data, so what we detect is an early warning. Visual inspection after 6 hours of running without lubricating oil revealed contact marks on the upper rings inside the cylinder.

- 70 repetitions with resampling of both training and evaluation examples
- 20 examples in training matrix
- Testing for 1-12 components
- Set rejection rate 5%
- NLL threshold learning set with 70 normal examples
- Evaluation set containing 70 examples with known labels, whereas 30-40 are normal

MFICA performs better	Equal performance	PCA performs better
47 times	18 times	5 times

Table 1
Performance statistics on the 70 runs

4.2 Results

We will compare the performance of MFICA to a very similar Principal Components Analysis method, described in [11], that does not obey the non-negativity constraints on **A** and **S**. For each of the 67 experiments we compare the best MFICA and PCA model. The model with lowest false alarm rate and highest detection rate (as a squared distance from the optimal Receiver-Operator-Characteristics point) is the best model. Of the 70 experiments MFICA is better than PCA in 47 ($\sim 67\%$), in 18 experiments ($\sim 26\%$) the two methods have equal performance. PCA is only better than MFICA in 5 of the experiments ($\sim 7\%$). If we compare the number of components in the "best" models (for MFICA the 65 experiments and PCA the 23 experiments), we see in Figure 5 that the *PCA best* histogram is peaked around 6 components. The *MFICA best* histogram is flatter and with a peak at only one component.

In Figure 6 we show the improvement in the ROC domain for the 47 experiments where MFICA is better than PCA. In 38 of those, the improved false alarm rate is achieved without decreasing the detection rate. The mean improvement is -0.045 (from 0.0752 to 0.03) on the false alarms and -0.0057 (from 1 to 0.9943) on the detection rate. So although PCA already gives good classification performance, MFICA is capable of improving on that.

5 Conclusion

We have described how the advanced blind source separation technique, Mean field independent components analysis, can be applied to a realistic condition-monitoring problem. The experiments show how this method performs better than a similar method based using Principal Components Analysis. We have planed improved experiments where the lubricating oil level is reduced over time, to induce even more subtle faults.

Acknowledgements

The European Commission through the AE-WATT project funds the authors, grant GRD2-2001-50014. Project partner MAN B&W Diesel A/S kindly pro-

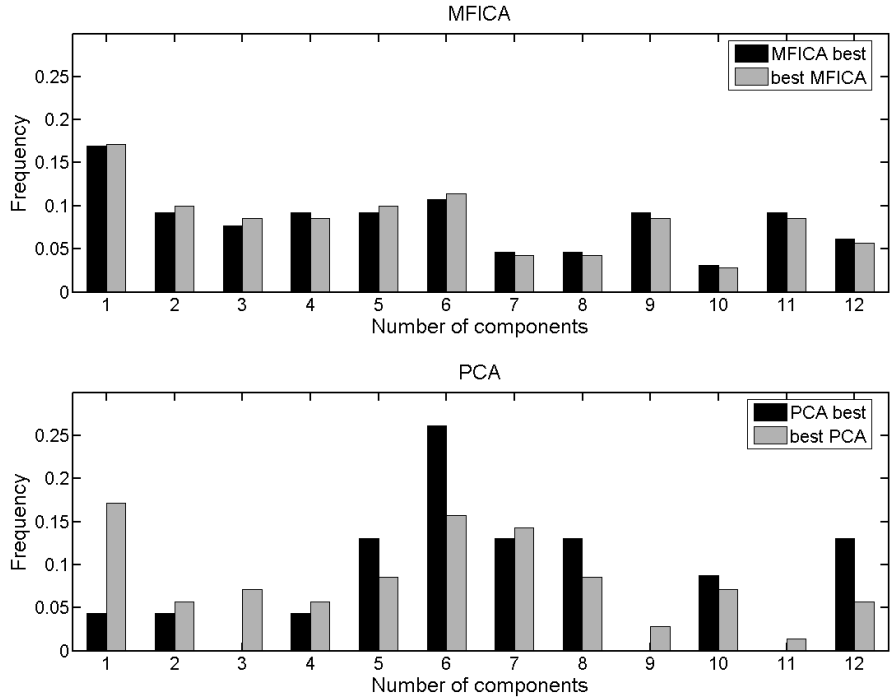

Fig. 5. Normalized histograms for best number of components. The histograms called *best MFICA/PCA* are from all experiments, whereas the *MFICA/PCA best* are build from the experiments where MFICA or PCA is at least as good as the other model, i.e. *MFICA best* is build from 65 examples and *PCA best* from 23.

vided experimental data. The authors recognize the fruitful collaboration with the department of Mechanical Engineering at Heriot-Watt University. The engine sketch to the left in Figure 1 is due to Ryan Douglas, Heriot-Watt University.

References

[1] G. Chandroth, A. Sharkey, Utilising the rotational motion of machinery in a high resolution data acquisition system, in: Proc of Computers and ships- from ship design and build, through automation and management and on to ship support, 1999.

[2] N. H. Pontoppidan, J. Larsen, Non-stationary condition monitoring through event alignment, in: IEEE Workshop on Machine Learning for Signal Processing, IEEE Press, Piscataway, New Jersey, 2004, pp. 499–508.
URL `http://www.imm.dtu.dk/pubdb/p.php?3131`

[3] P. Højen-Sørensen, O. Winther, L. Hansen, Mean field approaches to independent component analysis, Neural Computation 14 (2002) 889–918.
URL `http://www.imm.dtu.dk/pubdb/p.php?611`

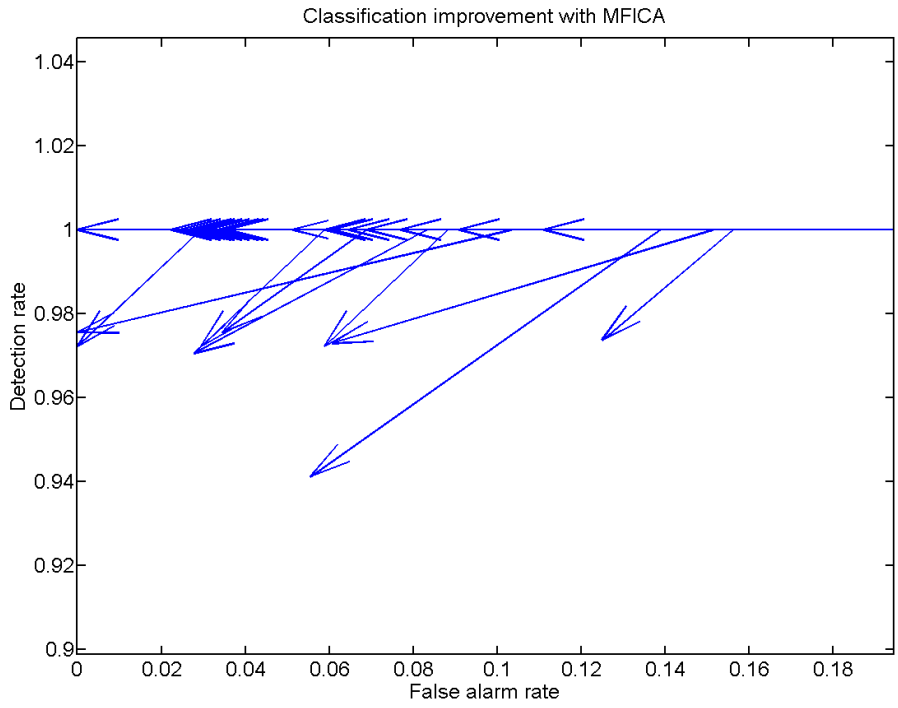

Fig. 6. Improvement with MFICA. The arrows point from the PCA to the MFICA ROC point for the experiments where MFICA is better. Only 9 of the 47 arrows point downwards, the rest show reduced false alarm rates without decreased detection rate

[4] M. Opper, O. Winther, Tractable approximations for probabilistic models: The adaptive thouless-anderson-palmer mean field approach, Phys. Rev. Lett. 86 (2001) 3695–3699.
URL `http://www.imm.dtu.dk/pubdb/p.php?614`

[5] T. Kolenda, S. Sigurdsson, O. Winther, L. Hansen, J. Larsen, Dtu:toolbox, Internet (2002).
URL `http://mole.imm.dtu.dk/toolbox/`

[6] B. Ripley, Pattern Recognition and Neural Networks, Cambridge University Press, Cambridge, 1996.

[7] G. Schwarz, Estimating the Dimension of a Model, The Annals of Statistics 6 (1978) 461–464.

[8] N. H. Pontoppidan, J. Larsen, T. Fog, Independent component analysis for detection of condition changes in large diesels, in: O. P. Shrivastav, B. Al-Najjar, R. B. Rao (Eds.), COMADEM 2003, COMADEM International, 2003, pp. 493–502.
URL `http://www.imm.dtu.dk/pubdb/p.php?2400`

[9] C. Bishop, Novelty detection and neural network validation, in: IEE Proceedings - Vision Image and Signal Processing, Vol. 141, 1994, pp. 217–222.

[10] N. H. Pontoppidan, J. Larsen, S. Sigurdsson, Non-stationary condition monitoring of large diesel engines with the AEWATT toolbox, in: H. C. Pusey (Ed.), Essential Technologies for succesful prognostics, 59th meeting of the Society for Machinery Failure Prevention Technology, Society for Machinery Failure Prevention Technology, Winchester, VA 22601, 2005, pp. 515–524.
URL http://www.imm.dtu.dk/pubdb/p.php?3351

[11] N. H. Pontoppidan, J. Larsen, Unsupervised condition change detection in large diesel engines, in: C. Molina, T. Adali, J. Larsen, M. Van Hulle, S. Douglas, J. Rouat (Eds.), 2003 IEEE Workshop on Neural Networks for Signal Processing, IEEE Press, Piscataway, New Jersey, 2003, pp. 565–574.
URL http://www.imm.dtu.dk/pubdb/p.php?2465