A combination of support vector machine and k-nearest neighbors for machine fault detection

Short title: **SVM and kNN for machine fault detection**

Amaury B. Andre
amaury.andre@semeq.com.br
SEMEQ
Av Laranjeiras, 2392
13485-254 Limeira - SP
Brazil

Eduardo Beltrame
eduardo@semeq.com.br
SEMEQ
Av Laranjeiras, 2392
13485-254 Limeira - SP
Brazil

Jacques Wainer
wainer@ic.unicamp.br
University of Campinas
Computing Institute
Av. Albert Einstein 1251
13083-852 Campinas - SP
Brazil

Corresponding author:
Jacques Wainer
wainer@ic.unicamp.br

University of Campinas
Av. Albert Einstein 1251
13083-852 Campinas - SP - Brazil
Phone: 55(19)3521-5871
fax: 55(19)3521-5847

Statement: this paper is not been published elsewhere and has not been submitted simultaneously for publication elsewhere.

A combination of support vector machine and k-nearest neighbors for machine fault detection

Abstract

This article presents a combination of support vector machine (SVM) and k-nearest neighbor (kNN) to monitor rotational machines using vibrational data. The system is used as triage for human analysis and thus a very low false negative rate is more important than high accuracy. Data is classified using a standard SVM, but for data within the SVM margin, where misclassifications are more likelly, a kNN is used to reduce the false negative rate. Using data from a month of operations of a predictive maintenance company, the system achieved zero false negative rate and accuracy ranging from 75% to 84% for different machine types such as induction motors, gears, and rolling element bearings.

keywords: machine condition monitoring ; fault detection; machine learning; support vector machine; k-nearest neighbors

1 Introduction

The high costs in maintaining complex and sophisticated equipments represent a large percentage in industrial organizations' profitability. Depending on the specific industry, maintenance can represent from 15% to 60% of the costs of goods produced (Mobley, 2002). Therefore, it is important to enhance maintenance management systems, to reduce costs and avoid breakdowns. Predictive maintenance is the process of discovering when an equipment needs maintenance in order to avoid a catastrophic failure. Condition monitoring, one of the techniques used in predictive maintenance, is the process of monitoring some parameter or condition of the equipment whose change may indicate a failure in development. For rotating machines, the analysis of the vibrational data is a particularly useful condition monitoring technique.

This work describes a practical application of a combination of two techniques from statistical machine learning, support vector machine (SVM) and k-nearest neighbor (kNN), to fault detection in vibration data. The data was gathered from real world samples, from the predictive maintenance enterprise SEMEQ[1]. SEMEQ performs monthly diagnostics on around 40,000 equipments from plants in Latin America and in the U.S. The main diagnostic tool used by SEMEQ is vibration analysis, but lubricant analysis and thermal imaging are also used. This work only concerns with vibration data.

The aim of this research is to reduce the total amount of vibration data sent to the human analysts. Since for each month, only a very small percentage of machines are in defective condition, all the vibration data of machines in good condition can be considered as "noise" in the analysis process - they should not be sent to the analysts for interpretation and represent time and effort not spent in the careful analysis and diagnostic of machines in defective condition. An automated system that only presents to the analyst the data that contain evidence of some problem could increase the overall productivity of the analysis.

For such a system, a false negative (data that is falsely labbeled by the system as negative or normal) is much more costly than a false positive (data that is falsely labeled by the system as positive or idicative of a problem). When a false negative occurs, a machine's data will not be inspected by the human analyst, and the owner of the machine will not be warned of the need to fix that machine. If the machine breaks down and it can be shown that there were signs of its impending failure, SEMEQ will incurre in severe penalties. False positives are cases where data from "good machines" are sent to analysis, which is not a problem since the analyst will have the chance to inspect the data and

[1] http://www.semeq.net/

discard the indication of failure. So, in this work, a very low false negative rate is considerably more important than a high accuracy rate.

1.1 Related work

In the related works, machine learning techniques such as support vector machines and artificial neural networks (ANN) were also applied to detect faults in different equipments, such as induction motors (Casimira et al., 2006), rolling element bearings (Samanta et al., 2003; Samanta and Nataraj, 2009), gear couplings (Samanta, 2004), turbine blades (Kuo, 1995), compressors (Yang et al., 2005), among others.

Samanta et al. (2003) compared three ANNs, a multilayer perceptron, a radial basis function network, and a probabilistic neural network, combined with a genetic algorithm for feature selection to detect faults in bearings. The networks were capable of detecting bearing faults with 99,83%, 87,5%, and 96,31% accuracy respectively. But no differentiation was made between false positives and false negatives. The algorithms were tested for a small number of cases (72 samples for training and 72 samples for testing).

Samanta (2004) worked with gear couplings and presented a comparison of ANN and SVM. In the paper, 266 samples were used for training both classifiers (ANN and SVM), and 126 samples for testing the accuracy of classification. The accuracy was close to 100%, but again no differentiation was made between false positives and false negatives. Also, the main focus of the work were the results obtained by the application of genetic algorithms for feature selection.

Samanta and Nataraj (2009) also dealt with bearing condition using two classifiers, ANN and SVM, but particle swarm optimization algorithm was used for feature selection. The training and testing data set used were larger than for the previous works. The false negative rate was 4% for the SVM and 11%

for the ANN.

Rojas and Nandi (2006) used a SVM for bearing fault detection. The number of samples used were much higher, 960 training and testing samples, and the accuracy was around 98 to 99%.

1.2 Structure of this paper

This paper is organized as follows. Section 2 presents the fundamental concepts; section 3 presents the details of our technique for the classification task. Section 4 presents the results of the application of the classifier, and section 5 discusses the limitations and contributions of this work.

2 Fundamental concepts

2.1 Vibration Analysis

The concept of “vibration signature analysis” is simple: machines in good condition generally tend to have a fairly stable vibration pattern, which can be considered as a “signature” (Randall, 1975). Changes in the internal conditions are often reflected as changes in the vibration pattern.

Piezoelectric transducers (accelerometers) are used to measure vibration. The result of data acquisition is the *acceleration time wave.*

For any machine, many acquisitions are made: in general, each equipment is measured in its three dimensions: horizontal, vertical and axial directions, both on the coupling and non-coupling sides. For example, an induction motor is usually measured in the five positions detailed in figure 1.

Vibration analysis is commonly done by examining individual frequencies present in the signal. These frequencies correspond to certain mechanical components (for example, the various pieces that make up a rolling-element

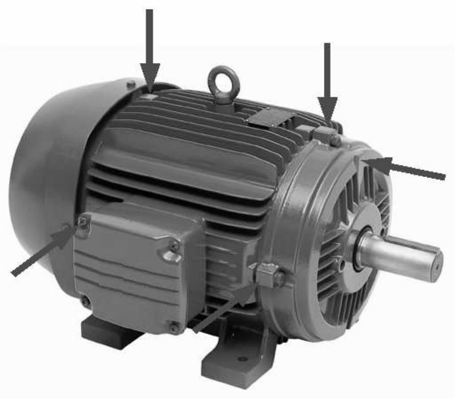

Figure 1: Induction motor example of where to make vibration measures

bearing) or certain malfunctions (such as shaft unbalance or misalignment). By examining these frequencies and their harmonics, the analyst can often identify the location and type of problem (Mobley, 2002).

The time domain wave data is transformed to its frequency domain representation. The acceleration time wave is transformed to its frequency domain, generating the *acceleration spectrum*; for higher precision in lower frequencies, the acceleration time wave is integrated and transformed, resulting in the *velocity spectrum*; for better precision in higher frequencies, the signal is demodulated and transformed, resulting in the *envelope spectrum.* Failures such as shaft misalignment or unbalance are better detected using the velocity spectra, while rolling element bearings faults are clearer in the envelope spectra.

2.2 Feature Extraction

As mentioned above, each measure point generates three spectra: the velocity, the acceleration, and the envelope spectrum. It is unfeasible to use the whole spectrum for classification, thus one usually computes single numbers that summarize characteristics of part or of the whole spectrum. These numbers are called *features.*

In this research we used the following features, computed from the time domain and from the frequency domain data of each spectrum:

- the root mean square (rms) (from time domain data) - *rms*
- peak to peak value (from time domain data) - *pkpk*
- the spectrum average (from frequency domain data) - *avg*
- the spectrum standard deviation (from frequency domain) - *std*
- the number of high peaks (peaks that are higher than the average plus three times the standard deviation from frequency domain) - *hp*
- the cumulative sum of the high peaks (from frequency domain) - *cshp*

In some cases, the detection of a fault require the comparisson of the current spectra with previously collected data. In ths research we used both the data collected in the previous month, and the *reference* for that spectrum. The reference is the data collected when the machine was known to be in good conditions, either because it was new or it had just undergone maintenance.

The following features are calculated as the difference from the current feature value, to the previous month and to the reference value:

- the difference from the previous month rms - *rmsa*
- the difference from the previous month pkpk - *pkpka*
- the difference from the previous month number of high peaks - *hpa*
- the difference from the previous month cumulative sum of the high peaks - *cshpa*
- the difference from the reference rms - *rmsr*
- the difference from the reference pkpk - *pkpkr*

- the difference from the reference number of high peaks - *hpr*
- the difference from the reference cumulative sum of the high peaks - *cshpr*

Thus, instead of using a particular spectrum as the input data to the classification process, we use the 14 features listed above.

2.3 SVM classifier

Support vector machine (SVM) is a statistical learning technique used in various classification problems (Cristianini and Shawe-Taylor, 2000). The standard SVM deals with binary classification problems, and its basic idea is to determine a linear boundary between the two different classes. The boudary (or *separating hyperplane*) is oriented in a way that its distance to the nearest data points in each class is maximized. The nearest data points are known as *support vectors.* Figure 2 illustrates the SVM: the boundary is represented by the solid line and the support vectors are emphasized. The *margin* of the SVM is the region from the hyperplanes that contain the suppor vectors in each side (in dashed line).

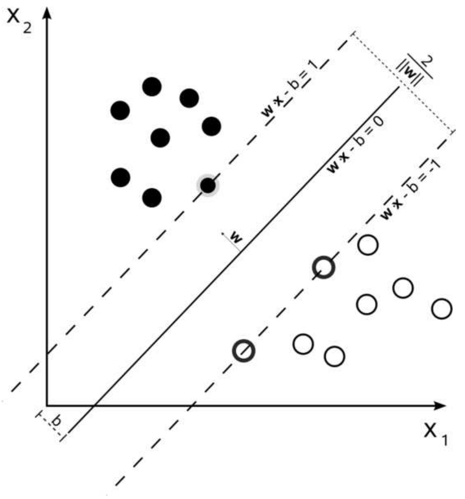

Figure 2: SVM hyperplane (figure taken form Wikipedia)

The linear boundary can be expressed as:

$$w \cdot x - b = 0, \qquad w \in \mathbf{R}^N, b \in \mathbf{R} \tag{1}$$

where w is the vector that defines the boundary, x is the input vector of dimension N, and b is a scalar threshold. The distance from the boundary to the margins is:

$$d = \frac{1}{\|w\|} \tag{2}$$

The optimal linear boundary that separates the two classes is calculated using the following optimization problem:

$$\begin{aligned} \text{minimize} &: \|w\| \\ \text{subject to} &: y_i(w \cdot x_i + b) \geq 1 \end{aligned}$$

where y_i is 1 or -1 depending on the class attributed to the input data x_i.

A further extension of SVM allows for soft margins, that is, one may allow points in the wrong side of the boundary, if that would increase the size of the margins. For a misclassified data x_i, ξ_i is its distance to the boundary. In this case, the optimization problem is:

$$\begin{aligned} \text{minimize} &: \tfrac{1}{2}\|w\|^2 + C\textstyle\sum_{i=1}^{M}\xi_i \\ \text{subject to} &: y_i(w \cdot x_i + b) \geq 1 - \xi_i \end{aligned} \tag{3}$$

where C is the error penalty for the misclassifications.

To solve equation 3, the problem is reduced to the equivalent Lagrangian dual problem. The solution itself will not be presented in this paper, but Widodo and Yang (2007) presents a very complete description of all steps and algorithms involved in SVM. In this research we used the Sequential Minimal Optimization algorithm from the package libsvm (Chang and Lin, 2001) to solve Eq. 3.

So far, we described SVM as an algorithm that attempts to calculate a linear boundary that separates two different classes. However, SVM can also be used in non-linear classification tasks, with the use of kernel functions. Kernel functions work by transforming the input state space to a higher dimensional space, where the data can be linearly separated as described before.

In this work, we use the Gaussian radial basis function (RBF) kernel, defined as:

$$RBF(x,y) = e^{-\|x-y\|^2/2\gamma^2} \tag{4}$$

where γ is the parameter of dispersion of the RBF.

3 Classification with Unbalanced Costs

For this research, the cost of a false negative is far superior than the cost of a false positive. We call this situation the *unbalanced cost* of errors.

The standard SVM technique assumes balanced cost. Equation 3 makes no distinction between the two forms of misclassifications: a positive example being classified as negative (false negative), and the other way around, both are treated equaly (the C and ξ_i terms)

We extended the standard SVM in two different ways to account for the unbalanced cost. The first way is to alter equation 3 to add different costs for false negatives and false positives. Thus, we make the cost of false positives C, and of false negatives $\alpha \times C$, where $\alpha > 1$. The new equation is:

$$\begin{array}{rcl} \text{minimize} & : & \frac{1}{2}\|w\|^2 + C\sum_{i \text{is a false positive}} \xi_i + \alpha \times C \sum_{j \text{is a false negative}} \xi_j \\ \text{subject to} & : & y_i(w \cdot x_i + b) \geq 1 - \xi_i \end{array} \tag{5}$$

This change would make it likely that the false negatives, if they exists, will be closer to the separating hyperplane than the false positives, since their cost is

α times that of the false positives.

3.1 SVM+kNN

Our second extension to a standard SVM is to more carefully consider the points that fall close to the separating boundary, since it is where the false negatives are likely to appear.

The k-nearest-neighbors (kNN) algorithm can be described as follows: a new sample is classified in accordance to a function of its k-most closest neighbors' classification. This function can be, for example, a majority polling, or a consensus criteria. The main parameter of the algorithm is the number of neighbors k that should be checked to decide a classification.

For the data points inside a region close to the hyperplane, defined as the region whose distance to the hyperplane is $t \times d$, where d is the margin of the SVM algorithm (equation 2), we apply a kNN consensus criteria, in which a new data is classified as normal if and only if *all* of its k neighbors in the training set are also normal. If at least one of the k-nearest neighbors is classified as defective, the new data point also received the defective classification.

Figure 3 illustrates the algorithm. The star is a new data point. Since it lies "close to the hyperplane", that is, within the dotted lines, it will not follow the standard SVM decision procedure and will not receive the classification of "normal" (white circles). We will instead use the kNN consensus criteria. Let assume that that k=5, that is, we will pool all 5 nearest neighbors of the new data point (the points within the circle) and if any of them is classified as defective (filled circles), the new data point will receive a defective classification.

This strategy combines the two main advantages of each algorithm: the generalization efficiency of SVM, specially for data points far away from the hyperplane, and the complex boundary that the kNN algorithm creates close to

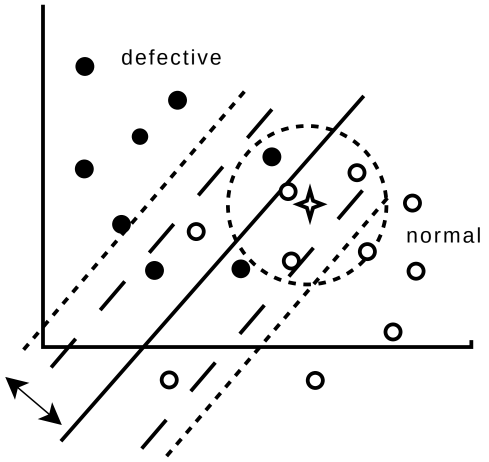

Figure 3: SVM + KNN

the separating hyperplane where the misclassifications, specially false negatives, are more likelly.

4 Results

4.1 Feature and parameter selection

Even though we do not work with the whole spectrum, but with 14 features described above, not all features may be useful for the classification task. In fact, features that are not useful for the classification task may hinder it (Guyon and Elisseeff, 2003). Thus selecting which of the features are relevant is an important step in any automatic classification task. Many techniques have been developed for feature selection (for example Guyon and Elisseeff (2003) and other papers in that JMLR Special issue on the topic). In this research we used a rather traditional method of forward selection by cross-validation.

The method starts by selecting a subset of the data and a quality metric for compassion between the different classifiers. Then, using only one of the features at a time, we compute the quality metric for each single feature classifier using cross-validation. The feature that maximizes the metric is selected. In the next interaction, all the features that were not included in the previous interactions, are added, one at a time, and the feature that again maximizes the metric is kept. The process repeats until the addition of any of the other features decreases the metric. The final set of featrues is then set of features of the previous interaction.

Besides selecting the features, our SVM+KNN classifier has the following parameters whose values need to be determined:

- C from equation 3
- γ from the RBF kernel function 4
- α the multiplying parameter for the cost of false negatives
- k from the kNN algorithm
- t the proportion of the margin in which the KNN algorithm will be used

The determination of the parameters of the classifier was done by brute force exploration of a few grid points. For each combination of features, using cross validation, we determined the quality metric of the classifier for all combinations of the 5 different values for each parameter.

For the feature selection algorithm we used a set of 1000 spectra from induction motors, equally divided into normal and defective. For the quality metric we used a variation of accuracy: if there was any case of false negatives we would say that the quality of the classifier was 0, if there was no false negatives, we measured the accuracy, that is the number of correct classifications divided by the total number of examples.

The results of the rounds of feature selection are displayed in Table 1. Each row in the table represents the results for one round in the feature selection algorithm, and the number in bold face is the best result for each round. Some classifiers have 0% accuracy because they generated false negatives. In the first column, there is the indication of the features already selected. The iterative procedure stopped at the sixth interaction because the accuracy dropped with the addition of any new feature.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Feature	rms	pkpk	hp	cshp	avg	dev	rmsa	pkpka	hpa	cshpa	msr	pkpkr	hpr	cshpr
	78.3	62.7	49	0	43	0	0	71	68.7	0	0	0	0	0
1		**77.7**	75.7	76.7	0	62.3	77.7	77.3	76	68	0	62.7	66.3	0
1,2			78	**78.8**	0	68.3	76	78.7	71.7	75	78.7	68	71	74.3
1,2,4			71.7		73.7	74.3	**79.7**	76.7	70	72.3	78	74.7	70.7	76.3
1,2,4,7			78.7		73	69.7		77.7	78.3	75	**81.3**	75.7	66	75.3
1,2,4,7,11			73.3		77.7	70.3		**80.7**	75.7	70.3		72.7	76.7	78

Table 1: Feature selection

The five selected features are: the rms and peak to peak current values, the difference from the current rms to the previous month, the difference from the current rms to the reference month, and the cumulative sum of the high peaks.

For these features, the parameters that achieved best accuracy (keeping the false negative rate to zero) are:

- $C = 1$
- $\alpha = 20$
- $\gamma = 10$
- $k = 23$
- $t = 0.75$

4.2 SVM + KNN Classifier

In this section we present the results of applying the SVM+KNN classifier with the features and parameters defined above. These results are a sample of the

SVM+KNN classifier in real life use. We used data sampled from 11 months of operations from SEMEQ to train the classifier, and used the classifier on all data from the 12th month to evaluate the false negative rate and accuracy. The data is separated by type of machine and by the type of spectrum. The training set was composed of the same number of normal and defective spectra (for each machine and spectrum types). The testing set was the whole month data.

Table 2 shows the results for induction motors. The first column indicates the type of spectra, the second the number of spectra used in the training set, sampled from the eleven months of operations and equally divided into normal and defective, the third column shows the number of spectra used in the test set, the fourth column the number of false positives, that is the number of spectra incorrectly labeled as defective by the classifier, the fifth column the number of false negatives, and the sixth column the accuracy of the classifier (number of correct labeled examples divided by the number of examples used in the test set).

Spectrum	Training Set	Testing Set	False Positives	False Negatives	Accuracy (%)
velocity	1000	1223	329	0	73.1
acceleration	1000	1058	211	0	80.1
envelope	1000	1051	288	0	72.6
				Average	75.3

Table 2: Results for induction motors

Table 3 shows the results for rolling element bearings, Table 4 for gear coupling, and Table 5 for pumps.

5 Discussion and Conclusion

There has been some tradition in the use of statistical learning techniques to automatically detect indications of a developing failure from spectra data

Spectrum	Training Set	Testing Set	False Positives	False Negatives	Accuracy (%)
velocity	791	790	188	0	76.2
acceleration	629	876	171	0	80.5
envelope	320	626	89	0	85.8
				Average	80.8

Table 3: Results for rolling element bearings

Spectrum	Training Set	Testing Set	False Positives	False Negatives	Accuracy (%)
velocity	40	1003	84	0	91.6
acceleration	774	1023	144	0	85.9
envelope	232	1009	260	0	74.2
				Average	83.9

Table 4: Results for gear coupling

(Samanta et al., 2003; Samanta, 2004; Rojas and Nandi, 2006; Samanta and Nataraj, 2009; Casimira et al., 2006; Yang et al., 2005). Some of these researches have achieved very high accuracy in dealing with particular machine types. But unless one is ready to accept a fully automatic vibration analysis system, the goal of high accuracy is somewhat misguided. If the automatic system is to be used in tandem with human analysis, then the goal should be very low false negative rates. This was the direction of this research.

All the previous work in machine learning techniques applied to machine fault diagnostic dealt with a single type of machine (motors, bearings, gear). Most of these works also used small data sets to train and test the classifiers, and the data was obtained in laboratory settings. Our research deals simultaneously with many different machine types, and we use a much larger data set. Finally, our research uses field data, both as training and as testing.

We achieved the main goal of zero false negative rate, but the accuracy of the system ranges from 75% to 83%, which still falls short of the accuracy achieved by other research. There are different directions to explore in order to improve the accuracy of our solution, provided it still maintain the false negative

Spectrum	Training Set	Testing Set	False Positives	False Negatives	Accuracy (%)
velocity	749	1028	184	0	82.1
acceleration	910	1037	318	0	69.3
envelope	710	1025	207	0	79.8
				Average	77.1

Table 5: Results for pumps

rate very low (or better, zero).

One direction of future exploration is to improve the feature selection and parameter setting phase. We opted to use single set of features (and its corresponding classifier parameters) for all the machine types. That was a project decision based on time constraints - feature selection and parameter setting is a time consuming step, and thus we selected the features for the most common machine type in our data set, and used those features and parameters for the other machine types. The fact that these features and parameters work well for the other machine types is in itself an interesting result, but it is likely that one may improve the accuracy by determining a particular feature and parameter set for each machine type.

The second line of future investigation is more costly but will probably result in a significant increase in accuracy. As we mentioned, our data represent real results from operations of SEMEQ, and this data associates to each *machine* the presence or not of a fault. But for training of the system, we associated the presence or not of a fault to each *spectrum* from the machine! Thus, there may be spectra that were learned as defective when in fact they were not, and some other spectrum for that machine were the ones in which the fault was clearer. Our system has a very cautious notion of “normality” - if something similar was classified as defective, the new data must also be classified as so. Therefore, this automatic association of a fault in the machine to a fault in any spectra of that machine is probably responsible for the high rate of false positives. But, in order to more precisely associate the fault in the machine to one or few of its spectra would require the help of an specialist. We have not yet determined if the cost of such enterprise is worth for the project.

By the time of the writing of this paper, the system described is in daily operation at SEMEQ for one year.

References

Casimira, R., Boutleuxa, E., Clercb, G., and Yahouib, A. (2006). The use of features selection and nearest neighbors rule for faults diagnostic in induction motors. *Engineering Applications of Artificial Intelligence*, 19(2):169–177.

Chang, C.-C. and Lin, C.-J. (2001). Libsvm: a library for support vector machines. *http://www.csie.ntu.edu.tw/ cjlin/libsvm*.

Cristianini, N. and Shawe-Taylor, N. (2000). *An Introduction to Support Vector Machines*. Cambridge University Press, Cambridge.

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

Kuo, R. J. (1995). Intelligent diagnosis for turbine blade faults using artificial neural networks and fuzzy logic. *Engineering Applications of Artificial Intelligence*, 8(1):25–34.

Mobley, R. K. (2002). *An introduction to predictive maintenance*. Butterworth Heinemann.

Randall, R. B. (1975). Vibration signature analysis - techniques and instrument systems. *Noise Control and Vibration Reduction*, 6:81–89.

Rojas, A. and Nandi, A. K. (2006). Practical scheme for fast detection and classification of rolling-element bearing faults using support vector machines. *Mechanical Systems and Signal Processing*, 20(7):1523–1536.

Samanta, B. (2004). Gear fault detection using artificial neural networks and support vector machines with genetic algorithms. *Mechanical Systems and Signal Processing*, 18(3):625–644.

Samanta, B., Al-Balushi, K. R., and Al-Araimi, S. A. (2003). Artificial neural networks and support vector machines with genetic algorithm for bearing fault detection. *Engineering Applications of Artificial Intelligence*, 16(7-8):657–665.

Samanta, B. and Nataraj, C. (2009). Use of particle swarm optimization for machinery fault detection. *Engineering Applications of Artificial Intelligence*, 18(2):308–316.

Widodo, A. and Yang, B.-S. (2007). Application of nonlinear feature extraction and support vector machines for fault diagnosis of induction motors. *Expert Systems with Applications*, 33(1):241–250.

Yang, B.-S., Hwang, W.-W., Kim, D.-J., and Tan, A. C. (2005). Condition classification of small reciprocating compressor for refrigerators using artificial neural networks and support vector machines. *Mechanical Systems and Signal Processing*, 19(2):371–390.