Fault Detection in Hard Disk Drives Based on Mixture of Gaussians

Lucas P. Queiroz, Francisco Caio M. Rodrigues, Joao Paulo P. Gomes, Felipe T. Brito, Iago C. Brito, Javam C. Machado
LSBD - Department of Science Computer
Federal University of Ceará, UFC
Fortaleza, Brazil
Email: {lucas.queiroz, caio.rodrigues, joao.pordeus, felipe.timbo, iago.chaves, javam.machado}@lsbd.ufc.br

***Abstract*—Being able to detect faults in Hard Disk Drives (HDD) can lead to significant benefits to computer manufacturers, users and storage system providers. As a consequence, several works have focused on the development of fault detection algorithms for HDDs. Recently, promising results were achieved by methods using SMART (Self-Monitoring Analysis and Reporting Technology) features and anomaly detection algorithms based on Mahalanobis distance. Nevertheless, the performance of such methods can be seriously degraded when the normality assumption of the data does not hold. As a way to overcome this issue, we propose a new method for fault detection in HDD based on a Gaussian Mixture Model (GMM). The proposed method is tested in a real world dataset and its performance is compared to three other HDD fault detection methods.**

I. Introduction

Fault detection methods have attracted much attention due to its possible benefits in various domains. Examples can be found in [1], [2], [3], [4] and [5]. In general, a fault detection system analyzes a set of sensor measurements and identifies the occurrence of anomalies that may indicate incipient failures (faults) [2]. In the presence of an incipient failure, the equipment may still be working but will fail in a near future.

In recent years, several works have focused on the development of fault detection algorithms with application on Hard Disk Drives (HDD). This fact may be explained by the increasing amount of data generated not only by people but also by machines (Internet of Things) [6]. As a consequence, the interest in storage services providers has significantly increased.

Currently, most HDD manufacturers implement the Self-Monitoring, Analysis and Reporting Technology (SMART). SMART is a monitoring system that tests several performance parameters to detect incipient failures [7]. In SMART anomalies are detected when any of the SMART parameters exceeds its threshold. Since the number of false alarms shall be minimized, the choice of the threshold results in a method capable of detecting only 3% to 10% of the fault occurrences [8].

Improving Fault Detection Rate (FDR) with reduced impact on the False Alarm Rate (FAR) have been the the objective of several works such as [9], [8], [10] and [11]. In [9] the authors used a non-parametric hypothesis test to monitor the SMART parameters and observed some improvements when compared to the standard SMART algorithm. Better results were also observed in [8], where the authors used a Support Vector Machine (SVM) classifier on SMART data. SVM achieved a FDR of 50.6% with zero FAR. The works of Wang *et. al.* [10], [11] modeled the problem as an anomaly detection task, where a statistical model is built using only fault-free HDDs. A HDD is detected as faulty if its SMART parameters have low probability of belonging to the healthy HDDs distribution. Currently, these works present the best results with FDRs of 67% and 68.4% respectively.

It is important to point that the works that achieved the best results assume that the healthy HDDs are normally distributed. Such approach may provide poor results when this assumption does not hold. A possible solution for this limitation consists in the use of non-parametric distribution models. Among them, one can cite the Gaussian Mixture Model (GMM) as one of the most commonly used. In the GMM, the data distribution is modeled by a linear combination of a given number of Gaussians. The combination factors and the parameters of the Gaussians are usually estimated using the Expectation-Maximization (EM) algorithm [12].

The following paper presents a fault detection method with application on Hard Disk Drives. The fault detection algorithm is based on the use of a GMM to model the distribution of SMART data from healthy HDDs. For a HDD at a given time instant, a window of time delayed SMART data are compared to the GMM model and a set of estimators are calculated to verify the state of health of the equipment. The performance of the proposed method is assessed on a real world HDD failure dataset and showed promising.

The remainder of this paper is as follows: In Section II presents the theoretical background. Section III describes our proposed method for fault detection. In Section IV, we discuss the experimental results comparing with related works. Some directions of the future work and conclusion are presented in Section V.

II. Theoretical Background

For a successful understanding of our proposal this section describes two methods (RFE and GMM) that are part of the proposed approach. The Recursive Feature Elimination (RFE) is a feature selection algorithm and the Gaussian Mixture

Model (GMM) is a non-parametric density estimation method. In addition to that, a fault detection performance metric, the Receiver Operating Characteristic (ROC) curve, is presented.

A. Recursive Feature Elimination

Recursive Feature Elimination (RFE) [13] is a widely-used feature selection algorithm which eliminates low weight features until a predefined number of features are left. RFE was chosen because it selects a feature subset which provides the best fault detection accuracy. RFE also alleviates the problem of overfitting, improving the performance of the model. An external estimator is used to assign weights to features and it's trained in every step of the process. First, the estimator is trained on the initial set of features and weights are assigned to each one of them. Then, features whose absolute weights are the smallest are eliminated from the current set features. The process is repeated until the desired number of features is reached.

RFE has been successfully employed in a number of applications, such as genetics [13], agroindustrial problems [14], among others. A common approach is to use RFE with a linear Support Vector Machine (SVM) classifier to select the features to be eliminated, where the feature ranking consist of weight values which are given by the correlation coefficients of the support vectors. Even though the method was originally conceived to work with SVM, it can be easily extended to use other classifiers, such as Random Forest (RF) [14].

B. Gaussian Mixture Model (GMM)

A Gaussian mixture model (GMM) is a statistical model that has the form of a weighted sum of Gaussian distributions. More formally, a GMM is given by the equation,

$$p(\mathbf{x}|\lambda) = \sum_{i=1}^{M} w_i \mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i), \tag{1}$$

where $\mathbf{x}$ is a m-dimensional vector, w_i is the weight of the i-th Gaussian, λ is its respective vector of parameters, M is the number of Gaussian and $\mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i)$ is a probability density function of the Gaussian distribution, defined by:

$$\mathcal{N}(\mathbf{x}|\mu_i, \Sigma_i) = \frac{\exp\left(-\frac{1}{2}(\mathbf{x}-\mu_i)^{\mathrm{T}}{\Sigma_i}^{-1}(\mathbf{x}-\mu_i)\right)}{\sqrt{(2\pi)^D|\Sigma_i|}}, \tag{2}$$

with mean vector μ_i and covariance matrix Σ_i. The mixture weights satisfy the following constraint: $\sum_{i=1}^{M} w_i = 1$. The parameters of the density model are collectively represented as $\lambda = \{w_i, \mu_i, \Sigma_i\}$, where $i = 1, ..., M$.

Given the training data, the maximum likelihood model parameters are estimated using the iterative Expectation-Maximization algorithm (EM). The EM algorithm [12] is the most popular technique used to estimate parameters of a mixture given a fixed number of mixture components, and it can be used to compute the parameters of any parametric mixture distribution.

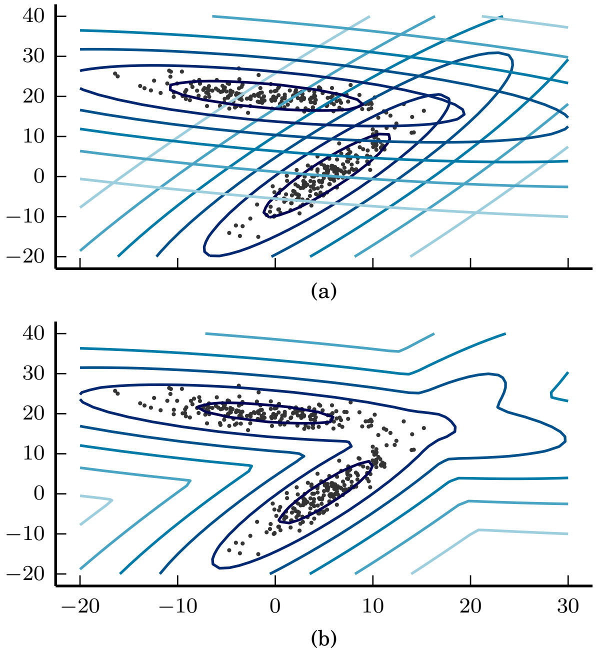

Fig. 1. Example of applying a Gaussian Mixture Model to fit the distribution of a set of points. Image 1 (a) shows two separate Gaussians fitting a subset of the original points. Image 1 (b) shows a mixture model composed of two Gaussians.

Figure 1 shows an example of fitting a mixture model on a set of points. Two subsets of points are clearly distinguishable in the original set. Figure 1 (a) shows how each subset would be fitted by an individual Gaussian. Figure 1 (b) shows a GMM approximation composed of two Gaussians, which was able to correctly capture the behavior of the whole set.

The number of Gaussian distributions is a free parameter when fitting a mixture model. To find a optimal number of Gaussians to fit the model, the Bayesian Information Criterion (BIC) [15] can be used. BIC is an information criteria that tries to balance the log-likelihood function and model complexity, tending to favor simpler and fitted models [16]. BIC is defined as:

$$\mathrm{BIC} = -2 \cdot \ln \hat{L} + r \cdot \ln(n), \tag{3}$$

where $\hat{L}$ is the maximized value of the likelihood function of the model, r is the number of free parameters in the model and n is the sample size. Models with the lowest BIC values are preferable, since they are less prone to overfitting.

C. Receiver Operating Characteristic (ROC)

A commonly used metric to evaluate the performance of a binary classifier is the Receiver Operating Characteristic (ROC) curve [17]. The ROC curve is a two-dimensional plot which illustrates the relationship between False Alarm Rate (FAR) and Fault Detection Rate (FDR). A false alarm (also called as a false positive) is a statistical error that incorrectly classifies the HDD as unhealthy when it is in fact healthy. On the other hand, a fault detection (also known as a true positive) is determined when a HDD is classified as faulty when it is

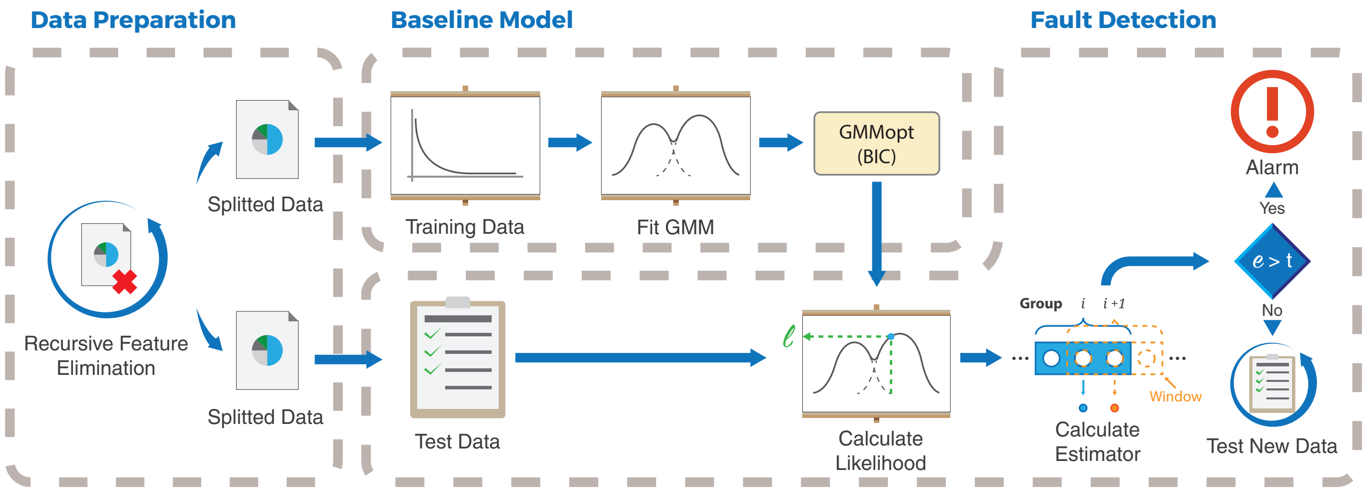

Fig. 2. Workflow of our method for fault detection. It involves the initial step of Data Preparation followed of Baseline Model and Fault Detection steps.

indeed. Generally, FAR is plotted on the horizontal axis, while FDR is plotted on the vertical axis. FAR is the ratio of the false alarmed HDDs to the total healthy HDDs. It is, for instance:

$$FAR = \frac{n_{fa}}{N_H}, \tag{4}$$

where n_{fa} is the number of false alarmed HDDs and N_H is the total number of healthy HDDs.

FDR can be defined as the ratio of the detected unhealthy HDDs to the total unhealthy HDDs. It is defined as:

$$FDR = \frac{n_{du}}{N_U}, \tag{5}$$

where n_{du} is the number of detected unhealthy HDDs and N_U is the total of unhealthy HDDs.

The main idea of ROC curve is to depict the trade-off between FAR and FDR. That is, a loose classification criteria could lead to a high FDR but could have the downside of a high FAR as well. On the other hand, a tight classification criteria would enforce a low FAR which could imply a low FDR.

III. Proposed Method

In this section, we describe the proposed method for fault detection in hard disk drives. The method is suited to operate on SMART data. The method is divided into three steps: data preparation, baseline model construction and fault detection. Figure 2 shows the scheme of our approach.

The data preparation step begins by applying the RFE method to select the features and then splitting the dataset in training data and testing data. The training data is formed by the early samples (beginning of life) of 60% of the disks labeled as healthy, randomly selected among all healthy disks. The test data consists of all the HDDs labeled as faulty and the healthy disks that are not in the training set.

In the baseline model step, the model is built using GMM from the training data. Intuitively, this model describes the behavior of healthy disks.

The fault detection step consists of observing the disks belonging to the testing set. An estimator is computed from the comparison between the disk data and the baseline model. The values returned by the estimator are monitored to verify if they exceeded a failure threshold.

A. Feature Selection

We run the RFE algorithm to get a rank list according to the feature importance. The RFE selection method is basically a recursive process that ranks the features according to its impact on the performance of a classifier. In this work, we used the Random Forest (RF) [18] classifier.

RF is an ensemble method that combines several decision trees. Each tree is fitted to a random sample with replacement (bootstrap). From out-of-bag samples, the forest chooses the final class through a voting or an averaging process. This method proved to achieve good results on many different datasets [19].

The feature importance might be estimated by Gini criterion [20]. It is based on the principle of impurity reduction that is followed in most traditional classification tree algorithms.

B. Baseline Model

In the training procedure, the training data is used to create a model of healthy HDDs. To represent this model, we fit a GMM according to the training data. The likelihood function given by a GMM is used to get a dissimilarity measure from a data sample, as described in Equation 1. This measure represents the proximity level between a generic HDD and a healthy HDD.

To fit the baseline model using a GMM, we need to pick the number of Gaussians to adequately represent the training data. In that case, BIC criteria was used to estimate the optimal number of Gaussians. As explained in Section II-B, BIC prefers well fitted model and penalizes the complexity of the GMM.

C. Fault Detection

In this step, we seek to distinguish anomalous behaviors from healthy behaviors in each disk belonging to the testing set using a dissimilarity measure based in the likelihood values. This measure is calculated as the negative log-likelihood of a disk sample with respect to the GMM adopted in the baseline model. Our intuition is that healthy disks within the testing set have small dissimilarity values, that is, they are near to the baseline model. On the other hand, the failed disks have high dissimilarity values, meaning that they are far from the baseline model.

However, if we look only at these dissimilarity values, there is the possibility of obtaining false alarms due to measurement errors or some unknown behavior. This problem was also reported in previous works such as [11], [10]. To minimize it, we used a sliding window on the dissimilarity values. Instead of observing only one point at a time, we observe a set of data within a sliding window of fixed size. The window is used to group some dissimilarity values in each disk. In this approach, an anomaly is detected only when several increased dissimilarity values are present. Figure 3 shows the process to calculate the estimators for each window.

To detect anomalies, we can compute an estimator over the dissimilarity values belonging to a window. A similar approach was also used in [10]. In this work, the author suggests the use of location changes or scale changes based on estimators. In our work, we evaluated the mean and variance of the dissimilarity measures as indicators of anomalies. In this case, a possible fault is detected when the estimator is greater than a certain threshold.

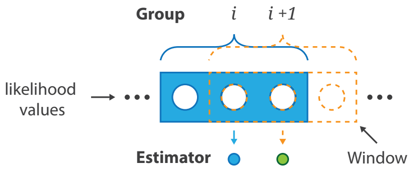

Fig. 3. Sliding window operation on the likelihood values for a given disk belonging to the testing set. At instant i, the estimator is calculated from first three likelihood values. In a next instant, the window is moved to the right (instant $i+1$) and another estimator is calculated.

IV. Results

In this section, we present the experimental results and implementation details of the proposed method. Moreover, a comparison is done with other approaches. It is worthy mentioning that the results are obtained by averaging the outputs of 10 experiments. The proposed method was implemented in Python using scikit-learn package [21] version 0.17.

A. Smart Data Set and Feature Selection

The dataset employed in this paper consists of time series of SMART features and it was provided by the Center for Magnetic Recording Research, University of California, San Diego [8]. This dataset was collected from real HDDs and it is the same employed by many related works [9], [10], [11]. It includes 369 drives, a total of 68411 samples, from one model, where 178 drives are labeled as healthy and 191 drives are labeled as failure. Hard drives labeled as healthy have passed through a reliability demonstration test executed by the manufacturer in a controlled environment. On the other hand, hard drives labeled as failed were returned to the manufacturer by the users after a failure. The 300 most recent samples (observations) were saved on disk and collected every two hours on the operating drives. Only the last 600 hours data could be recorded (i.e., if the time exceeded 600 hours, the data were overwritten). Some failed drives have less than 300 samples because they were not able to operate 600 hours. Each sample also contains features like the drives serial number, total power-on-hours, and 60 other performance-monitoring features. Not all features were monitored in every drive, and the unmonitored features were set to constants.

The application of the RFE method with the Random Forest estimator returned a set of 8 features[1]. The estimator was trained using 3-fold cross validation and forests consisting of 10 trees.

B. GMM-Based Method

The GMM used to fit the baseline model has the optimal number of Gaussians estimated by the BIC criteria, as explained in II-B. We configured the GMM with 10 Gaussians, that was where the BIC value seemed stable and close to a minimum.

To fit the training set using GMM, 20 iterations of the EM algorithm were enough to reach the stopping criteria for the procedure.

C. Performance Evaluation

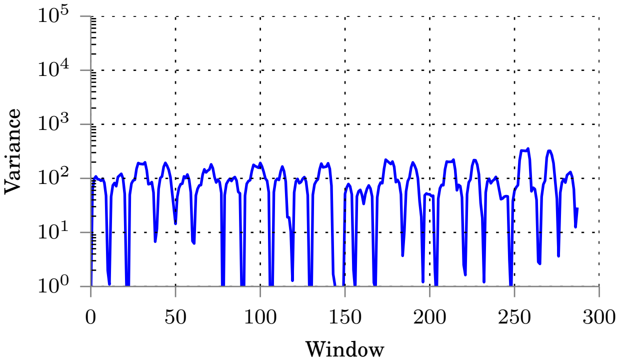

Fig. 4. Example of the behavior using variance estimator on the dissimilarity values of a healthy HDD. During observations, the estimator begins close to the value 10^2 and at the end of the experiments reaches peaks of two to three times the initial value.

The experiments were performed on the testing set, which consists of all 191 faulty HDDs plus the remaining 72 healthy

[1] The 8 selected features: Servo5, CSS, FlyHeight2 ,FlyHeight11, Servo10, FlyHeight3, Writes, Temp4.

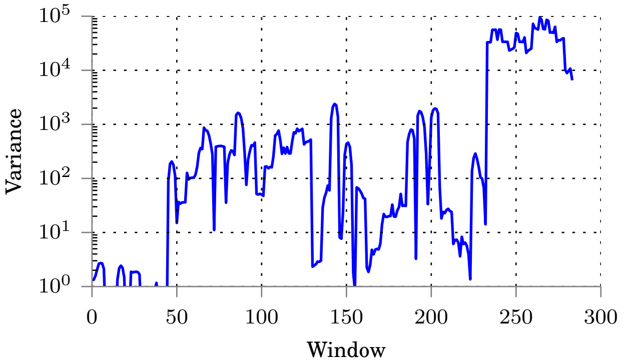

Fig. 5. During observations, the estimator presents large fluctuation during the lifetime and when it is near failure reaches values of order much upper than the healthy disk values shown in the Figure 4.

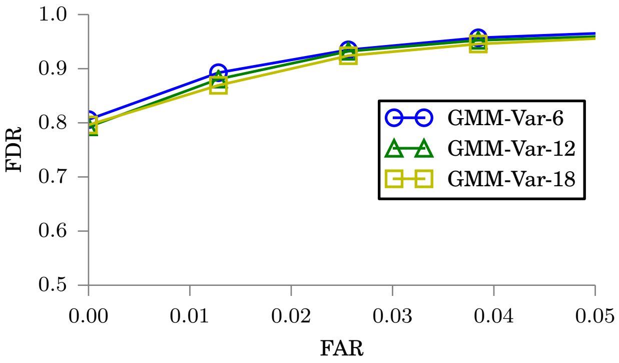

Fig. 6. ROC curves for the our method using variance estimator with different window sizes (6, 12, 18) in the interval between 0% to 5% of FAR.

disks. Figures 4 and 5 show typical behaviors of the variance estimator calculated for a healthy and a faulty HDD.

For the healthy disk, the variance values are at a similar level with a slight increase in the end of the data series. On the other hand, the variance values for a faulty disk present a significant increase which may indicate an incipient failure. Note that the values are on a logarithmic scale.

The mean and variance estimators are evaluated to compose our method using the performance of the ROC curve at 0% FAR, which every healthy HDD is correctly labeled. For each estimator, we compare the window sizes of 6, 12 and 18, which correspond respectively to the windows of 12, 24 and 36 hours, since the samples were collected every 2 hours. Figure 6 shows the performance of the variance estimator for each window size and Figure 7 shows for the mean estimator.

Both estimators presented values close to each other but for small window sizes the variance seemed slightly better than the mean. It is preferable use small windows because with less points is possible to detect anomalies earlier. For this reason, the posterior analyzes will be conducted with window size of 6.

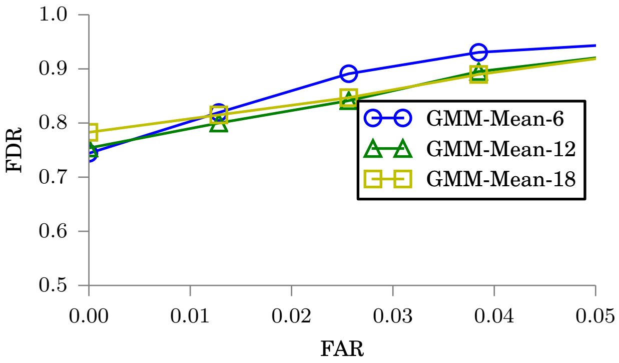

Fig. 7. ROC curves for the our method using mean estimator with different window sizes (6, 12, 18) in the interval between 0% to 5% of FAR.

As mentioned in Section I, we compare our results to a set of methods applied to the same dataset. Figure 8 shows the ROC curves obtained by TSP [11], FSMD [10], SVM [8] and our method using variance and mean estimators.

Our method achieved a FDR of 80.59% with 0% FAR using the variance estimator and 74.41% using the mean estimator. TSP, FSMD and SVM returned 68.42%, 67.02% and 50.6%, respectively. Considering these, we can state that our method had an improvement of 12% when compared to the best available method.

A possible explanation relies on the fact that the best results were obtained by methods which assume that the data are normally distributed [10] [11]. Since this assumption may not hold on in most cases, a GMM can be a better option to model the behavior of the healthy disks.

Concerning the computational complexity, TSP and FSMD methods are essentially bounded by the Mahalanobis distance measure. In order to compute this distance, it is necessary to calculate the covariance matrix. This process has a complexity of $O(nm^2)$ when $n > m$, where n is the number of observations and m is the number of features. In our method, the complexity is essentially given by the GMM, where the EM algorithm with M Gaussians requires $O(nm^2M)$ operations per iteration [22].

Considering the 10 Gaussians used to train the GMM and the 20 iterations of the EM algorithm, our method executes a number of computations in the order of 200 times greater than the TSP and FSMD methods. It should be noticed that this time is spent only in the offline training step.

V. Conclusions and Future Work

An approach based on a non-parametric model is presented for fault detection in hard disk drives. To begin, a feature selection is done using RFE with RF. A baseline model is built upon a subset of healthy HDDs (training set) using a GMM. For a given HDD, a dissimilarity measure is computed from the baseline model. This measure is grouped in a sliding window and an estimator is calculated on window values.

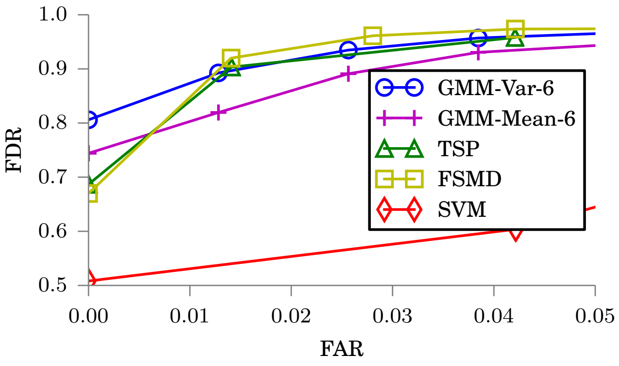

Fig. 8. ROC curves for all methods in the interval between 0% to 5% of FAR.

A failure is detected when an estimator returns a value that exceeds a certain threshold.

On the basis of our experiments we can state that our method outperformed previous HDD fault detection works (TSP, FSMD and SVM). We achieved 80.59% FDR at 0% FAR against 68.42% of the TSP method, the best previous result. Although the proposed method has a high computational complexity, this is only observable in the training phase, which is offline.

Finally, future works could take into account the growth of the estimator measure and get strategies for failure predictions in HDDs.

Acknowledgment

This Research was partially supported by Funcap, Capes/Brazil, LENOVO and LSBD/UFC.

References

[1] J. P. P. Gomes, R. K. H. Galvao, T. Yoneyama, and B. P. Leao, “A new degradation indicator based on a statistical anomaly approach,” *IEEE Transactions on Reliability*, vol. 65, no. 1, pp. 326–335, March 2016.

[2] L. R. Rodrigues, J. P. P. Gomes, F. A. S. Ferri, I. P. Medeiros, R. K. H. Galvo, and C. L. N. Jnior, “Use of phm information and system architecture for optimized aircraft maintenance planning,” *IEEE Systems Journal*, vol. 9, no. 4, pp. 1197–1207, Dec 2015.

[3] F. C. M. Rodrigues, L. P. Queiroz, J. P. P. Gomes, and J. C. Machado, “Predicting overtemperature events in graphics cards using regression models,” in *2015 Brazilian Conference on Intelligent Systems (BRACIS)*, Nov 2015, pp. 328–332.

[4] T. W. Rauber, L. H. S. Mello, V. F. Rocha, and F. M. Varejo, “Multi-label fault classification experiments in a chemical process,” in *Intelligent Systems (BRACIS), 2014 Brazilian Conference on*, Oct 2014, pp. 265–270.

[5] F. D. Maio, P. Baraldi, E. Zio, and R. Seraoui, “Fault detection in nuclear power plants components by a combination of statistical methods,” *IEEE Transactions on Reliability*, vol. 62, no. 4, pp. 833–845, Dec 2013.

[6] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of things: A survey on enabling technologies, protocols, and applications,” *IEEE Communications Surveys Tutorials*, vol. 17, no. 4, pp. 2347–2376, Fourthquarter 2015.

[7] E. Pinheiro, W.-D. Weber, and L. A. Barroso, “Failure trends in a large disk drive population.” in *FAST*, vol. 7, 2007, pp. 17–23.

[8] J. F. Murray, G. F. Hughes, and K. Kreutz-Delgado, “Machine learning methods for predicting failures in hard drives: A multiple-instance application,” *J. Mach. Learn. Res.*, vol. 6, pp. 783–816, 2005.

[9] G. Hughes, J. Murray, K. Kreutz-Delgado, and C. Elkan, “Improved disk-drive failure warnings,” *Reliability, IEEE Transactions on*, vol. 51, no. 3, pp. 350–357, 2002.

[10] Y. Wang, Q. Miao, E. Ma, K.-L. Tsui, and M. Pecht, “Online anomaly detection for hard disk drives based on mahalanobis distance,” *Reliability, IEEE Transactions on*, vol. 62, no. 1, pp. 136–145, 2013.

[11] Y. Wang, E. Ma, T. Chow, and K.-L. Tsui, “A two-step parametric method for failure prediction in hard disk drives,” *Industrial Informatics, IEEE Transactions on*, vol. 10, no. 1, pp. 419–430, 2014.

[12] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum likelihood from incomplete data via the em algorithm,” *Journal of the royal statistical society. Series B (methodological)*, pp. 1–38, 1977.

[13] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, “Gene selection for cancer classification using support vector machines,” *Machine learning*, vol. 46, no. 1-3, pp. 389–422, 2002.

[14] P. M. Granitto, C. Furlanello, F. Biasioli, and F. Gasperi, “Recursive feature elimination with random forest for ptr-ms analysis of agroindustrial products,” *Chemometrics and Intelligent Laboratory Systems*, vol. 83, no. 2, pp. 83–90, 2006.

[15] G. Schwarz *et al.*, “Estimating the dimension of a model,” *The annals of statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[16] R. E. Kass and A. E. Raftery, “Bayes factors,” *Journal of the american statistical association*, vol. 90, no. 430, pp. 773–795, 1995.

[17] T. Fawcett, “An introduction to roc analysis,” *Pattern Recogn. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.

[18] L. Breiman, “Random forests,” *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[19] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, “Do we need hundreds of classifiers to solve real world classification problems?” *Journal of Machine Learning Research*, vol. 15, pp. 3133–3181, 2014.

[20] L. Breiman, J. Friedman, C. J. Stone, and R. A. Olshen, *Classification and regression trees*. CRC press, 1984.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[22] R. A. Redner and H. F. Walker, “Mixture densities, maximum likelihood and the em algorithm,” *SIAM review*, vol. 26, no. 2, pp. 195–239, 1984.