DEEP AUTOENCODING GMM-BASED UNSUPERVISED ANOMALY DETECTION IN ACOUSTIC SIGNALS AND ITS HYPER-PARAMETER OPTIMIZATION

Harsh Purohit, Ryo Tanabe, Takashi Endo, Kaori Suefusa, Yuki Nikaido, Yohei Kawaguchi

Research and Development Group, Hitachi, Ltd.
1-280, Higashi-koigakubo Kokubunji-shi, Tokyo 185-8601, Japan
harsh_pramodbhai.purohit.yf@hitachi.com

ABSTRACT

Failures or breakdowns in factory machinery can be costly to companies, so there is an increasing demand for automatic machine inspection. Existing approaches to acoustic signal-based unsupervised anomaly detection, such as those using a deep autoencoder (DA) or Gaussian mixture model (GMM), have poor anomaly-detection performance. In this work, we propose a new method based on a deep autoencoding Gaussian mixture model with hyper-parameter optimization (DAGMM-HO). In our method, the DAGMM-HO applies the conventional DAGMM to the audio domain for the first time, with the idea that its total optimization on reduction of dimensions and statistical modelling will improve the anomaly-detection performance. In addition, the DAGMM-HO solves the hyper-parameter sensitivity problem of the conventional DAGMM by performing hyper-parameter optimization based on the gap statistic and the cumulative eigenvalues. Our evaluation of the proposed method with experimental data of the industrial fans showed that it significantly outperforms previous approaches and achieves up to a 20% improvement based on the standard AUC score.

Index Terms— Acoustic anomaly detection, Unsupervised learning, Autoencoder, Gaussian mixture model

1. INTRODUCTION

Anomaly events can decrease the quality of manufactured products and deteriorate the reliability of the industrial processes. To avoid this issue, many anomaly detection techniques are applied for the smart factory maintenance. These techniques are mainly based on sensor data parameters, environment variables, quality metrics of the industrial outcomes, and image-based methods [1, 2, 3]. Alternatively, recent acoustic scene classification and event detection technologies [4, 5] have shown promise for detecting anomalies from sound signals. Our objective in this study, is to develop an acoustic signal-based unsupervised anomaly detection method.

Existing unsupervised anomaly detection methods can be grouped into three categories. (i) Reconstruction-based methods, which assume that anomalies cannot be projected similarly on a low-dimensional space. For example, principle component analysis (PCA) and algorithms based on deep autoencoders (DA) [6, 7, 8, 9, 10] have been widely used an achieved acceptable results. However, the performance of these methods is limited because they are based on just one aspect of the reconstruction error. Moreover, low-dimensional representations sometimes loose the essential information of the input data. (ii) Cluster analysis methods like Gaussian mixture models (GMM) [11] and k-means [12] which, cluster data samples and find anomalies by means of a predefined distance score. However, the performance of these methods is limited by an over-simplified density estimation model that has insufficient capacity. (iii) One-class classification approaches, in which algorithms learn a discriminatory boundary surrounding the normal instances [13]. These discriminatory boundary and cluster-based methods can not be directly applied to very high-dimensional data.

Recently, a deep autoencoding Gaussian mixture model (DAGMM) for unsupervised anomaly detection was proposed [14]. The DAGMM model utilizes a deep autoencoder to generate a low-dimensional representation and reconstruction error for each input samples, which is then further fed into a Gaussian mixture model. The parameters of the DA and GMM networks are optimized simultaneously in an end-to-end fashion. However, when applying this method to acoustic anomaly detection, we found that its performance is significantly dependent on two hyper-parameters: (i) the number of GMM components and (ii) the dimension of the compressed features in the autoencoder.

To address these issues, we propose a deep autoencoding Gaussian mixture model with hyper-parameter optimization (DAGMM-HO) and evaluate it in the context of acoustic anomaly detection. This method can determine the anomaly patterns in the time and frequency domains when applied to spectrograms. We propose the following techniques to be used for constructing the DAGMM-HO model:

- Gap statistic [15]-based clustering to determine the number of GMM components
- Cumulative eigenvalues in principle component analysis (PCA) [16] to determine the size of the reduced dimension in DA.

We evaluated the proposed DAGMM-HO method to see if it could distinguish between normal and abnormal functioning of industrial fans with different sizes and designs. Standard area under curve (AUC) scores and F1 scores were used to evaluate and compare the proposed approach with the conventional methods.

2. DAGMM-HO-BASED ACOUSTIC ANOMALY DETECTION MODEL

The proposed acoustic anomaly detection model with the deep autoencoding Gaussian mixture model with hyper-parameter optimization (DAGMM-HO) is composed of two parts, as shown in Fig1. In the first part, hyper-parameter tuning has been performed to determine the correct number of GMM components and the optimal dimension for the autoencoder. The second part is the standard
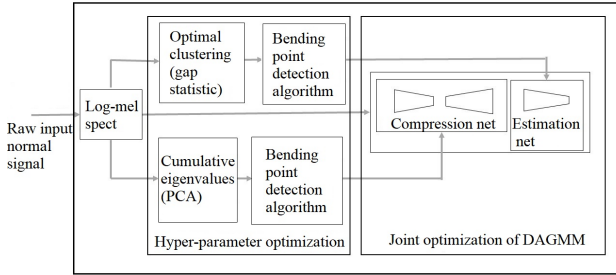
Figure 1: Overview of DAGMM-HO-based acoustic anomaly detection.

DAGMM model with the hyper-parameters selected from the first part. The DAGMM model consists of a *compression* sub-network for dimension reduction using a DA and an *estimation* sub-network that performs a mixture membership prediction in terms of the log-likelihood for the compressed representation of each data sample. Using these predicted membership values, we can estimate the parameters of the GMM through the simultaneous minimization of the reconstruction loss from the compression network and the sample log-likelihood from the estimation network. Before introducing the hyper-parameter optimization methods, we outline the DAGMM method and related issues.

2.1. DAGMM model

The compression network gives a compressed representation $\mathbf{z_c}$ of a given sample $\mathbf{x}$. The feature $\mathbf{z}$ can be defined from two sources: (i) the low-dimensional representation learned by the deep autoencoder and (ii) the feature derived from the reconstruction error. Formally, this is illustrated as

$$\mathbf{z_c} = f(\mathbf{x}; \theta_{enc}), \qquad (1)$$

$$\mathbf{x}' = g(\mathbf{z_c}; \theta_{dec}), \qquad (2)$$

$$\mathbf{z_r} = d(\mathbf{x}, \mathbf{x}'), \qquad (3)$$

$$\mathbf{z} = [\mathbf{z_c}, \mathbf{z_r}], \qquad (4)$$

where $\mathbf{z_r}$ is a potential multi-dimensional feature derived from the reconstruction error, θ_{enc} and θ_{dec} are the parameters of the deep autoencoder, $f(\cdot)$ denotes the encoding function, $g(\cdot)$ denotes the decoding function, and $d(\cdot)$ denotes the function for calculating the reconstruction error features, and $\mathbf{x}'$ is the reconstructed counter part of $\mathbf{x}$.

The estimation network performs a density estimation on the output $\mathbf{z}$ from the compression network. We use a multi-layer neural network to predict the mixture membership for each sample, i.e., we estimate GMM parameters in place of using conventional methods like expectation-maximization (EM) [11]. For the compressed representation $\mathbf{z}$, the estimation network performs the membership prediction as follows:

$$\boldsymbol{\gamma} = softmax(\mathbf{p}), \qquad (5)$$

$$\mathbf{p} = MLN(\mathbf{z}; \theta_{est}), \qquad (6)$$

where $\boldsymbol{\gamma}$ is a K-dimensional vector for the soft mixture-component membership prediction and $\mathbf{p}$ is the output of a multi-layer network

(MLN) parameterized by θ_{est}. Given a batch of N samples and their membership prediction, $\forall 1 \leq k \leq K$, where K is equivalent to the number of GMM components, we can further estimate the parameters in the GMM as

$$\phi_k = \sum_{i=1}^{N} \frac{\boldsymbol{\gamma}_{ik}}{N}, \qquad (7)$$

$$\boldsymbol{\mu}_k = \frac{\sum_{i=1}^{N} \boldsymbol{\gamma}_{ik} \mathbf{z}_i}{\sum_{i=1}^{N} \boldsymbol{\gamma}_{ik}}, \qquad (8)$$

$$\boldsymbol{\Sigma}_k = \frac{\sum_{i=1}^{N} \boldsymbol{\gamma}_{ik} (\mathbf{z}_i - \boldsymbol{\mu}_k)(\mathbf{z}_i - \boldsymbol{\mu}_k)^T}{\sum_{i=1}^{N} \boldsymbol{\gamma}_{ik}}, \qquad (9)$$

where $\boldsymbol{\gamma}_i$ is the membership prediction for the low-dimensional representation $\mathbf{z_i}$, and ϕ_k, $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$ are the mixture probability, mean, and covariance for component k in the GMM, respectively. Accordingly, the sample energy/likelihood of an input data sample can be inferred as

$$E(\mathbf{z}) = -\log\Big(\sum_{k=1}^{K} \phi_k \frac{\exp\big(-\frac{1}{2}(\mathbf{z} - \boldsymbol{\mu_k})^T \boldsymbol{\Sigma_k^{-1}}(\mathbf{z} - \boldsymbol{\mu_k}) \big)}{\sqrt{|2\pi\boldsymbol{\Sigma}_k|}} \Big). \qquad (10)$$

Given a dataset of N samples, the objective function for joint training of the compression and estimation networks is as follows:

$$J(\theta_{enc}, \theta_{dec}, \theta_{est}) = \frac{1}{N} \sum_{i=1}^{N} L(\mathbf{x}_i, \mathbf{x}'_i) + \frac{\lambda_1}{N} \sum_{i=1}^{N} E(\mathbf{z}_i) \\ + \lambda_2 P(\boldsymbol{\Sigma}). \qquad (11)$$

The three main components of the objective function are as follows. (i) $L(\mathbf{x}_i, \mathbf{x}'_i)$ represents the reconstruction error of the deep autoencoder for the given sample $\mathbf{x}$, i.e., $L(\mathbf{x}_i, \mathbf{x}'_i) = \|\mathbf{x}_i - \mathbf{x}'_i\|_2^2$. (ii) The second term shows the energy of the input sample $\mathbf{z_i}$. Minimization of this term leads us to model probabilities from the observed input samples. λ_1 is a hyper-parameter that decides the regularization weight from the estimation network. (iii) The third term is utilised to prevent the singularity problem. Here, $P(\boldsymbol{\Sigma}) = \sum_{k=1}^{K} \sum_{j=1}^{d} \frac{1}{\Sigma_{kj}}$, d is the number of dimensions in the low-dimensional representations provided by the compression network. Once the model is trained, we can obtained the output as the estimation of sample energy $E(\mathbf{z}_i)$ and predict the samples of high energy as anomalies in accordance with pre-chosen threshold η. However, hyper-parameters like K (number of components in GMM) and c (reduced dimension) have to be carefully optimized to ensure high accuracy in the subsequent anomaly detection task.

3. HYPER-PARAMETER OPTIMIZATION

In order to develop an acoustic anomaly detection method based on DAGMM, we found that two important hyper-parameters should be tuned accurately. Fig 2 depicts the change in accuracy of anomaly detection with respect to the number of GMM components and reduced dimension in the DA. The black point represents the optimal value for both hyper-parameters at which the anomaly detection system provides the highest accuracy. We propose methods to determine the correct number of clusters in the GMM and optimal dimensions in the DA with the use of normal data as follows.
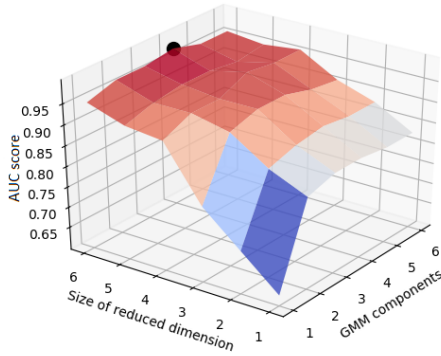
Figure 2: Change in accuracy with respect to number of GMM components and reduced dimensions.

Figure 3: Illustration of curve bending point detection.

Algorithm 1: Curve bending point detection

Input: Gap statistic values with respective number of clusters $\mathcal{G}_k$

Output: Curve bending point x^*

begin

$\quad (x, y) = (k, \mathcal{G}_k)$

$\quad x_{n_i} = \frac{x_i - min\{x_i\}}{max\{x_i\} - min\{x_i\}}; \quad y_{n_i} = \frac{y_i - min\{y_i\}}{max\{y_i\} - min\{y_i\}}$

$\quad x_{d_i} = x_{n_i}; \quad y_{d_i} = y_{n_i} - x_{n_i}$

$\quad x_{lmx} = x_{d_i}; \quad y_{lmx} = y_{d_i}$ such that

$\quad\quad y_{d_{i-1}} < y_{d_i}, \; y_{d_{i+1}} < y_{d_i}$

$\quad T_{lmx} = y_{lmx} - \frac{\sum_{i=1}^{m-1}(x_{n_{i+1}} - x_{n_i})}{m-1}$

$\quad$ If any difference value (x_{d_j}, y_{d_j}), where $j > i$, drops

$\quad\quad$ below the threshold, declare $x_{lmx} = x^*$

end

3.1. Components in GMM

We propose a method to estimate the optimal number of GMM components using gap statistic [15]. A curve is obtained by calculating the gap statistic for different numbers of GMM components. A bending point representing the maximum change on this curve is used to obtain the optimal component count. The idea behind gap statistic is to find a way to standardize the comparison of variance quantity $\mathcal{D}_o(k)$ with a null reference distribution of the data, i.e., a distribution with no obvious clustering. The gap statistic value $\mathcal{G}_k$ for different cluster counts k is calculated using uniform distribution as a reference with the following equation:

$$\mathcal{G}_k = \mathbb{E}_n\big(log(\mathcal{D}_r(k)) - log(\mathcal{D}_o(k)),\tag{12}$$

where $\mathbb{E}_n$ denotes expectation under a sample of size n from the reference distribution and $\mathcal{D}_o(k)$ and $\mathcal{D}_r(k)$ represent within-cluster dispersion in the original and reference data, respectively. Within-cluster dispersion $\mathcal{D}(k)$ between points in the given cluster C_k containing points n_k can be calculated as

$$\mathcal{D}(k) = \frac{1}{2n_k}\sum_{k=1}^{K}\sum_{x_i, x_j \in C_k} \|x_i - x_j\|_2^2.\tag{13}$$

The curve in Fig 3 represents gap statistic values $\mathcal{G}_k$ for different cluster counts k. We are interested in the point of maximum change (represented as the red dot) on the curve because it indicates the best cluster count. For example, in Fig 4, six different clusters are present but the best minimum number of components to represent the overall data is four. In order to find this, we need to find the bending point that gives the maximum change in the gap value with respect to the number of clusters. To find this point, we apply Algorithm 1 to the gap curve.
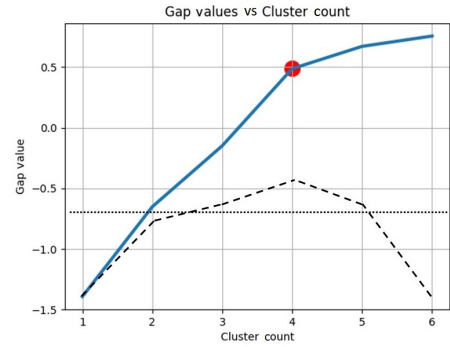
The input of Algorithm 1 consists of gap statistic values and the respective number of cluster counts. After fitting the curve to these values, we find the point that gives the maximum change in the gap value with respect to the particular cluster count. In the first step, a smoothing spline is fitted to preserve the shape of the curve (x, y) obtained using $\mathcal{G}_k$. Next, the difference values (x_{d_i}, y_{d_i}) are calculated over the normalized (x_{n_i}, y_{n_i}) values of the curve (dashed line curve in Fig 3). Local maxima (x_{lmx}, y_{lmx}) are obtained for this difference curve. A threshold T_{lmx} is defined using these local maxima values (doted line in Fig 3). Local maxima corresponding to the set of difference values (x_{d_j}, y_{d_j}) greater than this threshold is considered as the maximum change point. The value of $x_{lmx} = x^*$ corresponding to this maximum point is the crude yet best estimate of the number of distinct clusters that incorporate similar acoustic characteristics.

3.2. Reduced dimension in DA

To determine the optimal dimension size at the output of the encoder in the DA, we find the bending point on the curve obtained by taking the cumulative sum of the variance ratio for every component obtained by PCA. We can determine the variance contribution of every component and compress the data w.r.t the number of components required to maintain maximum variance in data using PCA. As described in Section 3.1, we use the bending point on the cumulative sum of the variance ratio curve to obtain best estimate of reduced dimension in the DA.

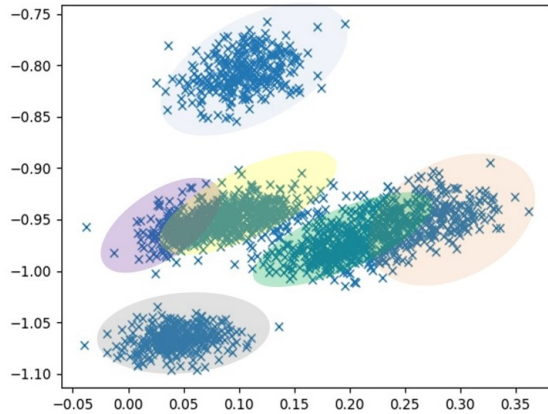We obtain a variance ratio ρ by dividing the variance value of

Figure 4: 2-D representation of audio signals of six industrial fans.

Table 1: Precision, recall, F1 score and AUC of compared methods

Method	Precision	Recall	F1 score	AUC
DAE	0.68	0.64	0.65	0.66
GMM	0.78	0.69	0.75	0.77
PCA+GMM	0.65	0.66	0.65	0.69
DAE+GMM	0.88	0.69	0.79	0.79
OC-SVM	0.62	0.85	0.66	0.65
PCA+OC-SVM	0.61	0.83	0.65	0.63
DAE+OC-SVM	0.64	0.86	0.67	0.67
Proposed	**0.94**	**0.93**	**0.94**	**0.96**

by a function f. $Drop(p)$ denotes a dropout layer with a keep probability p during training. Euclidean distance was used to compute the reconstruction error for training.

In the experiments, all the anomalous segments and the same number of normal segments were used as the test dataset and the rest of the normal segments were utilized as the training dataset in our experiment. Energy scores (Equation 10) for test-data samples were calculated while predicting from the trained model. These scores were then used to determine the state/health of the machine using a pre-specified threshold.

We compared the proposed approach with the following unsupervised techniques. (i) Deep autoencoder (DA). The deep autoencoder is trained on the normal data and the sample reconstruction error is used as the criterion for anomaly detection. (ii) One-class support vector machine (OC-SVM). OC-SVM is a popular kernel-based method of anomaly detection. In the experiment, we used the radial basis function (RBF) kernel. Note that we provided OC-SVM with an unfair advantage by optimizing its hyper-parameters ν and γ. (iii) Gaussian mixture model (GMM) with *diagonal* covariance matrix method. Variants of these methods can be used in a two-step approach. For instance, in step one, we can perform the dimensionality reduction by PCA or by learning the deep autoencoder. Then in step two, we can use OC-SVM or GMM for the classification and density estimation, respectively.

The area under the curve (AUC), precision, recall, and F1 score were calculated to evaluate the anomaly detection performance. As shown in Table 1, the proposed method had a significant improvement in terms of F1 score. Moreover, the two-step method with DA and density estimation performed better compared to the single-step methods and provided up to a 20 % improvement in AUC.

each component v_i by the total variance v_t, i.e., the sum of variances of all the components obtained by PCA. Then, a variance ratio curve $\mathcal{V}$ is obtained by taking the cumulative sum of this ρ of every component arranged in descending order. Finally, as described in Algorithm 1, we find the bending point on the variance ratio curve and determine the optimal size of the encoder output c.

We utilize PCA to estimate the dimension because, in terms of construction, PCA is similar to encoder in that it reduces dimension through the projection of data on principal components. For both, the DA and PCA, the parameters are estimated such that the reconstruction error is minimized, so, PCA can give a good estimate of the reduced dimension size to be used in a neural network-based autoencoder. The DA is flexible in design and non-linearities can be introduced in the model by using different activation functions. Hence, with an increasing amount of features, PCA might be unable to achieve as good a compression as the DA can, but it can be used to obtain an estimate of the reduced dimension.

4. EXPERIMENTATION

To evaluate the effectiveness of the proposed approach, we applied it using real experimental data. We collected sound data ($\sim$ 10 min each) of six industrial fans with different sizes and manufacturers. A microphone was placed at 50 cm away from each fan to collect the audio signal. Saturation of the recorded signal was avoided during data collection. We observed that similar fans generated homogeneous acoustic profiles.

For feature extraction, a log-melspectrogram was calculated from raw audio signals. To compute the mel spectrogram, we considered a frame size of 1024, a hop size of 512, and 64 mel filter banks. We predict that for these input features, our trained model will have a low reconstruction error. The DAGMM-HO network structure for the experiment is summarized as follows.

After determining the optimal number of components in the GMM, i.e., k, and optimal number of dimensions, c, the compression and estimation networks were defined as follows. Compression network: $FC(60, 30, tanh)$ - $FC(30, 10, tanh)$ - $FC(10, c, none)$ - $FC(c, 10, tanh)$ - $FC(10, 30, tanh)$ - $FC(30, 60, tanh)$. Estimation network: $FC(c + 1, 10, tanh)$ - $Drop(0.5)$ - $FC(10, k, softmax)$, where $FC(a, b, f)$ means a fully connected layer with a input neurons and b output neurons activated

5. CONCLUSIONS

We developed an acoustic anomaly detection method using a deep autoencoding Gaussian mixture model with hyper-parameter optimization. We proposed an automated procedure to determine the optimal number of GMM components and reduce the dimension in the DA. Significant improvement in accuracy over conventional methods was achieved during an experiment with real-time data of industrial fans. The proposed method achieved an AUC score of 96% and F1 score of 94%, while the best conventional technique had AUC and F1 scores of only 79%. In the future, domain adaptation methods can be used to extend our solution to a wider variety of sensors.

6. REFERENCES

[1] K. A. Kaiser and N. Z. Gebraeel, "Predictive maintenance management using sensor-based degradation models," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 39, no. 4, pp. 840–849, 2009.

[2] J. A. Carino, M. Delgado-Prieto, D. Zurita, M. Millan, J. A. O. Redondo, and R. Romero-Troncoso, "Enhanced industrial machinery condition monitoring methodology based on novelty detection and multi-modal analysis," *IEEE Access*, vol. 4, pp. 7594–7604, 2016.

[3] O. Janssens, R. Schulz, V. Slavkovikj, K. Stockman, M. Loccufier, R. Van de Walle, and S. Van Hoecke, "Thermal image based fault diagnosis for rotating machinery," *Infrared Physics & Technology*, vol. 73, pp. 78–87, 2015.

[4] H. Lim, J. Park, and Y. Han, "Rare sound event detection using 1d convolutional recurrent neural networks," in *Proceedings of the Detection and Classification of Acoustic Scenes and Events (DCASE) Workshop*, 2017, pp. 80–84.

[5] Y. Koizumi, S. Saito, H. Uematsu, Y. Kawachi, and N. Harada, "Unsupervised detection of anomalous sound based on deep learning and the Neyman–Pearson lemma," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 1, pp. 212–224, 2018.

[6] T. Tagawa, Y. Tadokoro, and T. Yairi, "Structured denoising autoencoder for fault detection and analysis," in *Proceedings of the Asian Conference on Machine Learning (ACML)*, 2015, pp. 96–111.

[7] E. Marchi, F. Vesperini, F. Eyben, S. Squartini, and B. Schuller, "A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional LSTM neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 1996–2000.

[8] Y. Kawaguchi and T. Endo, "How can we detect anomalies from subsampled audio signals?" in *Proceedings of the IEEE 27th International Workshop on Machine Learning for Signal Processing (MLSP)*, 2017, pp. 1–6.

[9] D. Oh and I. Yun, "Residual error based anomaly detection using auto-encoder in smd machine sound," *Sensors*, vol. 18, no. 5, p. 1308, 2018.

[10] R. Salakhutdinov and G. Hinton, "Deep boltzmann machines," in *Artificial intelligence and statistics*, 2009, pp. 448–455.

[11] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006.

[12] M. Goldstein and S. Uchida, "A comparative evaluation of unsupervised anomaly detection algorithms for multivariate data," *PloS one*, vol. 11, no. 4, p. 173, 2016.

[13] F. Aurino, M. Folla, F. Gargiulo, V. Moscato, A. Picariello, and C. Sansone, "One-class svm based approach for detecting anomalous audio events," in *2014 International Conference on Intelligent Networking and Collaborative Systems*. IEEE, 2014, pp. 145–151.

[14] B. Zong, Q. Song, M. R. Min, W. Cheng, C. Lumezanu, D. ki Cho, and H. Chen, "Deep Autoencoding Gaussian Mixture Model for Unsupervised Anomaly Detection," in *International Conference on Learning Representations (ICLR)*, 2018.

[15] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 2, pp. 411–423, 2001.

[16] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.