Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier

A Methodology for Prognostics Under the Conditions of Limited Failure Data Availability

GISHAN D. RANASINGHE[1], TONY LINDGREN[2], MARK GIROLAMI[3], AJITH K. PARLIKAD[1]
[1]University of Cambridge, Department of Engineering, Institute for Manufacturing, 17 Charles Babbage Road, Cambridge, CB3 0FS, UK.
[2]Stockholm University, Department of Computer and System Sciences, Forum 100, 164 40 Kista, Sweden and Scania Commercial Vehicles, Service Support Solutions, Södertälje, Sweden.
[3]University of Cambridge, Department of Engineering, Civil Engineering Building, 7A JJ Thomson Avenue, Cambridge, CB3 0FA, UK and The Alan Turing Institute, British Library, 96 Euston Road, London, NW1 2DB, UK.

Corresponding author: Gishan D. Ranasinghe (e-mail: gd416@cam.ac.uk).

The first author's PhD research is funded by the EPSRC through a Doctoral Training Partnership grant (EP/M508007/1). This research was partly supported by the Next Generation Digital Infrastructure project (EP/R004935/1) funded by the EPSRC and BT.

ABSTRACT When failure data are limited, data-driven prognostics solutions underperform since the number of failure data samples is insufficient for training prognostics models effectively. In order to address this problem, we present a novel methodology for generating failure data which allows training datasets to be augmented so that the number of failure data samples is increased. In contrast to existing data generation techniques which duplicate or randomly generate data, the proposed methodology is capable of generating new and realistic failure data samples. The methodology utilises the conditional generative adversarial network and auxiliary information pertaining to failure modes to control and direct the failure data generation process. The theoretical foundation of the methodology in a non-parametric setting is presented and we show that it holds in practice using empirical results. The methodology is evaluated in a real-world case study involving the prediction of air purge valve failures in heavy-trucks. Two prognostics models are developed using the gradient boosting machine and random forest classifiers. When these models are trained on the augmented training dataset, they outperformed the best solution previously proposed in the literature for the case study by a large margin. More specifically, costs due to breakdowns and false alarms are reduced by 44%.

INDEX TERMS Equipment prognostics, Expert knowledge, Generative modelling, Limited failure data, Physics of failure.

I. INTRODUCTION

PROGNOSTICS involve predicting the time to failure of equipment or predicting the probability that a piece of equipment operates without failure up to some future time (e.g. until the next inspection time or the end of current mission window) [1]. Prognostics is usually performed using expert knowledge, condition monitoring data and/or event data relating to past failures. Despite their popularity, the long-lasting problem with data-driven prognostics is that they rely on large amounts of historical failure data (i.e. run-to-failure data indicating past degradation patterns) to estimate prognostics model parameters [2]. Nevertheless, historical failure data are limited in real-world industrial scenarios due to three major reasons: (i) rare (yet adverse) failures; (ii) over-protective maintenance and replacement regimes; (iii) highly reliable equipment [3], [4]. This causes training datasets to be imbalanced, which makes it difficult for data-driven algorithms to estimate model parameters from degradation patterns and characterise system performance for prognostics modelling [4]. Hence, predictions produced by these models are associated with high uncertainty and therefore introduce additional costs due to under maintenance and over maintenance of equipment and false alarms.

The objective of this paper is to present a methodology for generating real-valued failure data so that the training datasets used for prognostics modelling can be augmented to include an increased number of realistic failure data samples. In the context of this work, "real-valued" data means data that realistically reflect the behaviour of the equipment of interest. Using theoretical and empirical results, we show that the proposed methodology has the potential to address the long-lasting problem of limited failure data availability for prognostics, and hence allow predictions produced by data-driven prognostics models to be associated with minimal uncertainty when real failure data are limited. To this end, we expanded the research presented in our conference paper which was presented at the 2019 IEEE International Conference on Prognostics and Health Management, San Francisco, USA [5].

When failure data are limited for data-driven prognostics, the use of physical model-based and knowledge-based prognostics solutions have been unsuccessful due to the following: physical model-based prognostics require the empirical estimation of physics parameters which is difficult and expensive [2]. Moreover, large amounts of historical failure data are still required for validating physical model-based prognostics solutions [2]. Knowledge-based prognostics involve obtaining domain knowledge and converting it into rules which is also difficult in most industrial scenarios [2]. More importantly, when the number of rules increases, knowledge-based prognostics solutions suffer from the combinatorial explosion problem and once built, they do not generalise into new situations that are not covered explicitly in their knowledge bases [2].

Existing techniques used to address the problem of limited failure data availability for data-driven prognostics include undersampling and oversampling techniques. Unfortunately, these techniques also have major shortcomings. Undersampling discards potentially useful non-failure data samples, hence, for instance, can degrade the discriminating power of a classifier [6]. Since random oversampling and advanced techniques such as the synthetic minority oversampling technique and adaptive synthesis involve duplicating existing failure data or randomly generating data, they do not introduce real-valued failure data samples [7], [8]. Recently, there has been an emergence of a few generative adversarial networks (GAN)-based oversampling techniques for failure prediction. The common shortcoming of these techniques is that they do not condition the noise being added to newly generated data samples which leads to different modes of data being generated (i.e. the failure data generation process is not controlled and directed) [8]. Hence, the fundamental problem of limited failure data availability for prognostics is not addressed sufficiently.

We aim to address this problem by developing a methodology that is capable of generating real-valued failure data. The methodology utilises auxiliary information available in the prognostics domain to condition the noise being added to newly generated data samples, thus the failure data generation process is controlled and directed [1]. In the context of this work, "auxiliary information" means the additional information that can be obtained from industrial scenarios and adds value to the understanding of failure dynamics of the equipment of interest (e.g. expert knowledge, physics of failure and information contains within maintenance records). The methodology estimates a generative model that captures the semantic features of the failure mode that needs predicting using real failure data, noise and more importantly, using auxiliary information pertaining to the failure mode. Then it uses the estimated generative model to generate new failure data by sampling from a joint distribution of noise and auxiliary information. The tool used for estimating the generative model is a conditional generative adversarial network (CGAN) [8].

Despite its success in the image recognition domain (see [8] and [9]), generating real-valued data for prognostics presents the following domain-specific research challenges: (i) systematically identifying auxiliary information pertaining to failure modes that is useful for controlling and directing the failure data generation process; (ii) systematically converting different kinds of auxiliary information that can be in complex and different forms (e.g. aural, visual, formulas and text entries) into a form that can be integrated into the failure data generation process to condition the noise. The research presented in this paper takes the first step towards overcoming these challenges, and thus developing a methodology for generating real-valued failure data for prognostics under the conditions of limited failure data availability.

Following the problem formulation and method for measuring the extent of limited failure data availability presented in our conference paper (see Sec. III in [5]), this paper commences by providing the theoretical foundation of the proposed methodology (Sec. II). The theoretical foundation is divided into three parts: the prerequisite and assumption required for the methodology are presented in Sec II-A; a detailed description of the methodology and suitable evaluation methods are provided in Sec II-B; the theoretical results of integral parts of the methodology are presented in Sec II-C. The methodology is evaluated in the Scania air purge valve prognostics problem and empirical results are discussed in Sec. III. The paper is concluded in Sec. IV.

II. THEORETICAL FOUNDATION

In this section, we use the value function $V(G, D)$ of the minimax game used to estimate a generative model using the CGAN (see Eq. 1). G and D are the generator and discriminator artificial neural networks of the CGAN architecture respectively. X is real failure data samples, Y is auxiliary information and Z is noise. A detailed description of the CGAN and its value function are provided in our conference paper (see Sec. II in [5]).

$$\min_G \max_D V(G, D) = \mathbb{E}_{x \sim p_{\text{data}}}[\log D(x \mid y)] + \mathbb{E}_{z \sim p_{\text{noise}}}[\log(1 - D(G(z \mid y)))] \quad (1)$$

A. PREREQUISITE AND ASSUMPTION

The prerequisite of the methodology is that auxiliary information pertaining to the failure mode (see Table 1) needs to be available in the industrial scenario in addition to condition monitoring and event data which are limited. This is reasonable in practice due to the following: (i) expert knowledge is available in most industrial scenarios since it is a popular information source used for maintenance decision making and risk analysis of industrial systems [1]; (ii) physical laws and equations that are developed using the physics of failure technique are well-documented in the literature (e.g. Paris' Law for fatigue crack propagation and Power Law for analysis of reliability of complex systems) [10]; (iii) inspection, replacement and maintenance activities are documented, stored and accessible in most industrial scenarios due to the advances in information and communication technology and cloud data storage model [11].

TABLE 1. Different Kinds of Auxiliary Information Available in the Prognostics Domain

Expert knowledge	• Equipment similarity information • Knowledge on failure causes and failure modes • Empirically validated rules • Known failure thresholds
Physics of failure	• Physical laws • Differential equations • Stochastic differential equations
Maintenance records	• Information contains within inspection records • Information contains within repair and replacement records

The following assumption is made in the current version of the methodology: the failure mode that needs predicting causes equipment to fail due to gradual degradation and is not a sudden failure. This is a reasonable assumption in practice since the dominant failure modes of industrial equipment cause the hazard rate to be increased with the equipment age [12]. By making this assumption we imply the following: (i) data-driven prognostics using condition monitoring and/or event data is feasible since the evolution of fault into failure causes monotonic trends in equipment condition and performance, and hence data-driven predictive algorithms can use these trends to estimate prognostics model parameters; (ii) since the failure is not random, predicting equipment failure is useful as maintenance before failure affects the probability that the equipment will fail in the next instance, hence downtime can be prevented [12].

B. DESCRIPTION OF THE METHODOLOGY

The proposed methodology for generating real-valued failure data consists of three phases (see Fig. 1). In the remainder of this section, these three phases are discussed in detail.

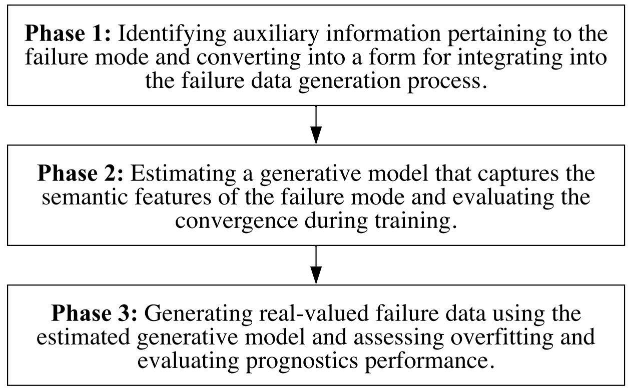

FIGURE 1. Diagram outlining the three phases of the proposed methodology for generating real-valued failure data.

1) Identifying auxiliary information pertaining to the failure mode and converting into a form for integrating into the failure data generation process

Previously in Table 1, we outlined different kinds of auxiliary information that might be typically available in the prognostics domain. The challenge is to identify pieces of auxiliary information that are useful for generating real-valued failure data for the failure mode that needs predicting. Here, expert knowledge provided in the literature and obtained from on-site maintenance engineers is used to identify auxiliary information that may potentially be useful for generating real-valued failure data. Then Phases two and three are iteratively performed in order to identify the ideal set of auxiliary information that leads to the satisfactory prognostics performance.

Auxiliary information pertaining to failure modes can be in complex and different forms. For example, aural, visual or text entries of expert knowledge and maintenance records, and mathematical equations of physics of failure. Thus, the challenge is to convert this information into a form that can be integrated into the failure data generation process for conditioning the noise.

In this methodology, auxiliary information is converted into vector representations. This can be further explained using the following example involving auxiliary information obtained from expert knowledge on failure causes and failure modes. Imagine that there is a set of outdoor electronic equipment (e.g. electronic circuits and joints inside distribution point boxes of broadband lines) that has failed. Over time maintenance engineers have gained knowledge that the failure rate of this set of equipment increases during the rainy season. This information can be further validated using maintenance records which reveal that water ingress (i.e. failure cause) caused corrosion and electrical shorts (i.e. failure modes) and led to the failure of the set of electronic equipment. This information can be used to generate real-valued failure data by informing the CGAN that a newly generated failure data sample may contain the patterns in historical rainfall data in the locations the equipment is placed at. Thus, the noise being added to newly generated data samples is conditioned on auxiliary information related to the past failures of the equipment.

In order to integrate auxiliary information into the failure data generation process, we first convert it into an abstract form. This allows equipment-specific information to be generalised to all the equipment that has failed under the failure mode that needs predicting. For instance, if the rainfall in a particular location where the target equipment placed at during their degradation period is recorded as the rainfall at the location where the equipment A, B and C placed at increased from 43 mm to 65 mm, once converted into the abstract form this information becomes some variable X increases. Thus, specific terms such as equipment A, B and C, rainfall and numerical thresholds are ignored. Then the abstracted information is converted into the statistical form by representing it as some continuous variable C. The continuous variable C can be converted into a distribution between some values y_0 and y_1. Finally, this distribution can be represented as a vector Y containing some values $\{y \in Y \mid y_0 < y < y_1, \text{and y increases}\}$.

2) Estimating a generative model that captures the semantic features of the failure mode and evaluating the convergence during training

We first structure the historical condition monitoring and/or event data that will be used for prognostics modelling as shown in Fig. 2. The historical data are divided into three datasets: (i) training dataset (referred to as the original training dataset) which includes data for training prognostics models; (ii) validation dataset which is used for hyperparameter tuning; (iii) testing dataset which is used to evaluate prognostics models on previously unseen data.

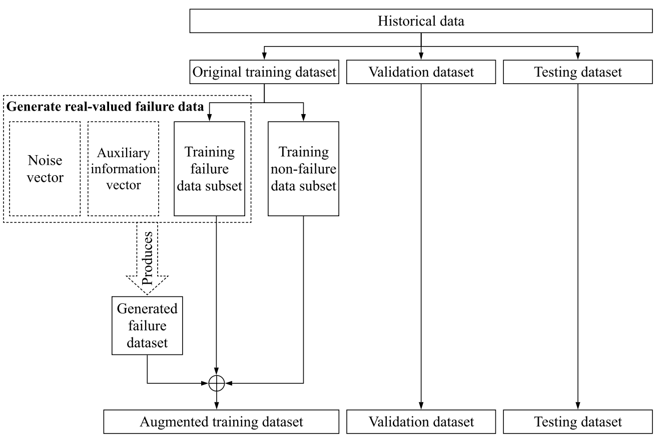

FIGURE 2. Diagram depicting how historical condition monitoring and/or event data are structured in the proposed methodology. The original training dataset is augmented by combining generated real-valued failure data samples. The validation and testing datasets are left unchanged for hyperparameter tuning and comparative evaluation of prognostics models respectively.

The objective of generating real-valued failure data is to augment the original training dataset so that the number of failure samples available for training prognostics models is increased. To this end, as shown in Fig. 2 we first divide the original training dataset into two subsets containing failure data (referred to as the training failure data subset) and non-failure data (referred to as the training non-failure data subset). The training failure data subset is used to estimate a generative model that captures the semantic features of the failure mode using noise and auxiliary information vectors. After the new dataset containing real-valued failure data samples is generated, it is combined with the two subsets to obtain the augmented training dataset. The validation and testing datasets are left unchanged for hyperparameter tuning and comparative evaluation of prognostics models.

In order to estimate the generative model, the CGAN architecture presented in Fig. 3 is implemented using theoretical aspects and the value function $V(G, D)$ of CGAN. In the proposed methodology, the generator and discriminator are two deep neural networks (DNNs). The first step is to combine the noise vector Z with the auxiliary information vector Y into the joint distribution $Pr(Z, Y)$. This is used as the input to the generator DNN. Then data samples in the training failure data subset X are combined with the auxiliary information vector Y into the joint distribution $Pr(X, Y)$. This is used as the input to the discriminator DNN.

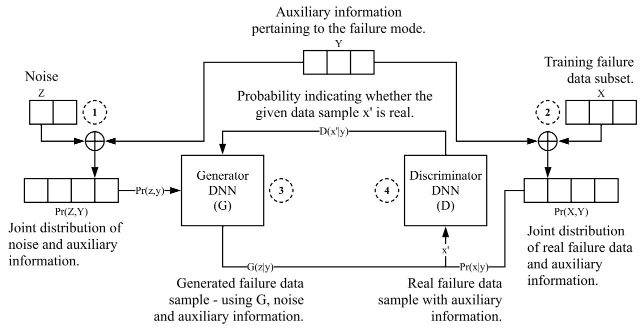

FIGURE 3. Diagram depicting the architecture of the conditional generative adversarial network implemented for the proposed methodology. The generator and discriminator are deep neural networks (DNNs).

The objective of the generator G is to learn to fool the discriminator D into believing that a generated failure data sample is real (i.e. the generated failure data sample is sampled from the real failure data distribution) during each training step. Thus, the generator produces a fake failure data sample $G(z \mid y)$ by conditioning the noise z on auxiliary information y. More specifically, the generator aims to minimise its loss function $\log(1 - D(G(z \mid y)))$ (i.e. learn to fool the discriminator the most). The objective of the discriminator D is to detect whether a given failure data sample is real. Finally, the discriminator produces a probability $D(x' \mid y)$ indicating how much it believes the given failure data sample x' is real. More specifically, the discriminator tries to minimise its loss function $\log(D(x' \mid y))$ (i.e. learn to discriminate between real and fake failure data samples better). Hence, during the training period the generator is allowed to converge the generated failure data distribution to the real failure data distribution. At the end of the training, that is, when the generated failure data distribution is converged to the real failure data distribution (also known as the Nash equilibrium of the minimax game), the generator is capable of generating new and realistic failure data samples that the discriminator cannot discriminate as real or fake. Thus, the generator DNN has now captured the semantic features of the failure mode, and hence it can be used for generating real-valued failure data in the next phase.

In order to evaluate the convergence of generated failure data distribution to the real failure data distribution, we use the Kolmogorov-Smirnov test (K-S test) and measure the similarity between the two distributions as the number of training steps increases. The K-S test provides a statistical test framework for identifying whether two samples are drawn from the same distribution. The null hypothesis of this test is, the two distributions are statistically similar. Thus, the objective of this evaluation is to statistically identify whether the following null hypothesis can be rejected: the generated failure data distribution and real failure data distribution are statistically similar.

To this end, we perform the K-S test and observe how the p-value (probability value) changes as the number of training steps increases. In principle, we expect the p-value to monotonically increase and then converge to 1 which indicates there is no evidence against the null hypothesis, hence the hypothesis cannot be rejected. This means the generative model is capable of perfectly replicating the real failure data distribution. However, in practice, it is possible for the p-value to monotonically increase, but only converge to a value greater than 0.05 due to the uncertainty associated with model hyperparameter tuning and data noise. Nevertheless, a p-value greater than 0.05 indicates weak evidence against the null hypothesis, hence the hypothesis cannot be rejected still. This means the following statement holds: the generated failure data distribution and real failure data distribution are statistically similar. If this is the case, the generative model is converging the generated failure data distribution to the real failure data distribution as the number of training steps increases. Hence, it is capable of replicating the real failure data distribution.

The result of the aforementioned convergence evaluation can be biased if the generative model is overfitting to training failure data. To address this issue, in the next phase, we propose another evaluation method that identifies whether the result of the convergence evaluation is biased.

3) Generating real-valued failure data using the estimated generative model and assessing overfitting and evaluating prognostics performance

A new noise vector and the previously used auxiliary information vector are used as inputs to the estimated generative model to generate real-valued failure data. More specifically, given the joint distribution of noise and auxiliary information as the input, the generator DNN estimates a set of real-valued failure data samples. Once the real-valued failure data samples are generated, they are combined with the original training dataset to obtain the augmented training dataset as shown in Fig. 2. Finally, the generated failure data samples are evaluated using the following overfitting assessment and prognostics performance evaluation.

In order to identify whether the generative model is overfitting to training failure data, we use maximum mean discrepancy (MMD) to quantify the mean discrepancy between real failure data and generated failure data. MMD measures the distance between distributions to identify whether two data samples are generated from different distributions [13]. If the generative model is overfitting to training failure data, the MMD between generated failure data samples and training failure data samples is proven to be significantly lower than generated failure data samples and test failure data samples [14]. Thus, the objective of this evaluation is to statistically identify whether the following null hypothesis can be rejected: the generative model is not overfitting to training failure data. If this null hypothesis can be retained (i.e. p-value is greater than 0.05), the MMD between generated failure data samples and test failure data samples is at most as large as the MMD between generated failure data samples and training failure data samples. In other words, we intend to identify whether the following statement holds: generated failure data samples do not look more similar to the training failure data samples than they do to the test failure data samples, hence the generative model is not overfitting to training failure data.

In order to identify whether the generated failure data are real-valued and hence improve the predictive power, we compare the prognostics performance obtained when prognostics models are trained on the augmented training dataset to the benchmark prognostics performance obtained when they are trained on the original training dataset. The standard evaluation metrics such as correlation coefficient, accuracy and error rate are not suitable for evaluating prognostics models when failure data are limited since they will be biased to the majority class (i.e. non-failure data class) regardless of the minority class (i.e. failure data class) leads to the poor performance [15]. Precision and recall, however, are not affected by the majority class hence suitable for measuring predictive performance when the data are limited [15]. The precision is the fraction of correctly predicted failures among all the predicted instances that include actual failures and false alarms. The recall is the fraction of correctly predicted failures among all the actual failures. This means higher the precision lower the number of false alarms and higher the recall lower the number of undetected failures. Formally, the precision and recall are given by Eq. 2 and Eq. 3 respectively.

$$\text{Precision} = \frac{\text{Predicted failures}}{\text{Predicted failures} + \text{False alarms}} \tag{2}$$

$$\text{Recall} = \frac{\text{Predicted failures}}{\text{Predicted failures} + \text{Undetected failures}} \tag{3}$$

C. THEORETICAL RESULTS

As discussed in Sec. II-B, Phase two of the methodology involves estimating a generative model that has captured the semantic features of the failure mode that needs predicting. More specifically, conditioning generator and discriminator on auxiliary information pertaining to the failure mode allows the CGAN to estimate a generative model that is perfectly replicating the real failure data distribution (in principle).

Since this is integral to the proposed methodology, in the following, we provide the theoretical grounding for this claim.

We adopt the mathematical proof of GAN provided in [16] and extend it to include the joint probability distribution which allows the conditioning of noise on auxiliary information pertaining to the failure mode.

Using Lemma 1, we first theorise that the CGAN implemented for the methodology (see Fig. 3) allows estimating the optimal discriminator which can perfectly discriminate between real and generated failure data samples for the fixed generator.

Lemma 1: Given the joint probability distribution of real failure data $p_{\text{data}}(x, y)$ and generated failure data $p_G(x, y)$, where $\{x \in X\}$ consists of real and generated input failure data samples and $\{y \in Y\}$ is the auxiliary information vector, the optimal discriminator D^*_G for the fixed generator is,

$$D^*_G(x \mid y) = \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)} \tag{4}$$

Intuitively, Eq. 4 states that when the real failure data distribution and generated failure data distribution are given, the optimal discriminator should be able to identify the real failure data fraction. The proof of Lemma 1 is provided in the appendix.

Then using Lemma 1 and below Theorem 1, we theorise that the CGAN implemented for the methodology can be trained to estimate the optimal generator which can perfectly replicate the real failure data distribution, hence captures the semantic features of the failure mode that needs predicting.

Theorem 1: The training criterion $C(G) = \min_G \max_D V(G, D)$ achieves a unique global minimum if and only if the generated failure data distribution p_G is equal to the real failure data distribution p_{data}.

The proof of Theorem 1 is provided in the appendix. As shown in the proof, the training criterion $C(G)$ achieves a unique global minimum of $-2\log 2$ when the optimal discriminator $D^*_G(x \mid y) = 1/2$. And this unique global minimum is achieved only when $p_G = p_{\text{data}}$. This means at the unique global minimum, we have the optimal generator which is perfectly replicating the real failure data distribution, and hence it has captured the semantic features of the failure mode that needs predicting.

Finally, we show how a training algorithm can be used to achieve the unique global minimum of the training criterion $C(G)$. The generative model in the methodology is estimated by training the CGAN using minibatch stochastic gradient descent algorithm. The loss functions of discriminator and generator DNNs are obtained by decomposing the value function of the minimax game in CGAN. More specifically, decomposing the value function $V(G, D)$ into the loss functions of discriminator D generator G gives Eq. 5 and 6 respectively.

$$\max_D V(D) = \overbrace{\mathbb{E}_{x\sim p_{\text{data}}}[\log D(x \mid y)]}^{\text{Recognise real failure data samples better}} + \underbrace{\mathbb{E}_{z\sim p_{\text{noise}}}[\log(1 - D(G(z \mid y)))]}_{\text{Recognise generated failure data samples better}} \tag{5}$$

$$\min_G V(G) = \underbrace{\mathbb{E}_{z\sim p_{\text{noise}}}[\log(1 - D(G(z \mid y)))]}_{\text{Optimise G to fool the discriminator the most}} \tag{6}$$

The loss function of the discriminator is the total loss when recognising real failure data samples and generated failure data samples. Hence, for a minibatch of m examples $\{xy^{(1)}, ..., xy^{(m)}\}$ from the joint distribution of real failure data and auxiliary information pertaining to the failure mode $Pr(X, Y)$, and for a minibatch of m examples $\{zy^{(1)}, ..., zy^{(m)}\}$ from the joint distribution of noise and auxiliary information pertaining to the failure mode $Pr(Z, Y)$, the total loss of discriminator D is,

$$\frac{1}{m}\sum_{i=1}^{m}[\log D(xy^{(i)}) + \log(1 - D(G(zy^{(i)})))]$$

Similarly, using the loss function of the generator G given in Eq. 6, the total loss of G for a minibatch of m examples $\{zy^{(1)}, ..., zy^{(m)}\}$ from the joint distribution of noise and auxiliary information pertaining to the failure mode $Pr(Z, Y)$ is,

$$\frac{1}{m}\sum_{i=1}^{m}\log(1 - D(G(zy^{(i)})))$$

The loss functions are used as the objective functions of the minibatch stochastic gradient descent algorithm (see Algorithm 1). The discriminator DNN is executed twice per training step before it calculates the total loss: once for real failure data and once for generated failure data. The generator DNN is executed only once per training step. When the discriminator and generator losses are known, the gradients with regard to their parameters are calculated and backpropagated through the discriminator and generator DNNs to optimise model parameters.

In order to show that the training algorithm converges the generated failure data distribution p_G to the real-failure data distribution p_{data}, and hence reaches the unique global minimum provided in Theorem 1, we consider the equilibrium point of the minimax game. During the two-player minimax game, the discriminator D tries to maximise the value function $V(G, D)$ for a given generator G whilst G tries to minimise it for the optimal D. Thus, the objective is to reach a saddle point as illustrated in Fig. 4. This saddle point is the equilibrium point of the minimax game. Moreover, according to Theorem 1, we know that at the equilibrium point the training criterion $C(G)$ achieves the unique global minimum and at that point, we have the optimal discriminator $D^*_G = 1/2$, and a generative model that is perfectly replicating the real failure data distribution. Hence, if we are able to show that the value function $V(G, D)$ can reach the

Algorithm 1: Minibatch stochastic gradient descent training of the conditional generative adversarial network implemented for the proposed methodology.

for *number of training steps* **do**
  for *number of steps to apply to the discriminator* **do**
  • Sample a minibatch of m examples $\{zy^{(1)}, ..., zy^{(m)}\}$ from the joint distribution of noise and auxiliary information pertaining to the failure mode $p_G(z, y)$.
  • Sample a minibatch of m examples $\{xy^{(1)}, ..., xy^{(m)}\}$ from the joint distribution of real failure data and auxiliary information pertaining to the failure mode $p_{\text{data}}(x, y)$.
  • Update the discriminator model parameters by ascending its stochastic gradient:

$$\bigtriangledown\theta_d \frac{1}{m}\sum_{i=1}^{m}[\log D(xy^{(i)})+\log(1-D(G(zy^{(i)})))]$$

  end
  • Sample a minibatch of m examples $\{zy^{(1)}, ..., zy^{(m)}\}$ from the joint distribution of noise and auxiliary information pertaining to the failure mode $p_G(z, y)$.
  • Update the generator model parameters by descending its stochastic gradient:

$$\bigtriangledown\theta_g \frac{1}{m}\sum_{i=1}^{m}\log(1 - D(G(zy^{(i)}))]$$

end

saddle point through gradient descent, we have shown that Algorithm 1 converges the generated failure data distribution to the real-failure data distribution (i.e. $p_G = p_{\text{data}}$).

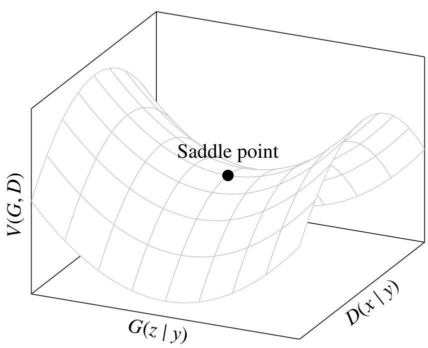

FIGURE 4. Plot depicting the saddle point of the value function $V(G, D)$ of minimax game. The objective of training the proposed CGAN architecture is to reach the saddle point and hence achieve the unique global minimum of the training criterion $C(G) = \min_G \max_D V(G, D)$.

In order to reach the saddle point, we apply gradient descent to discriminator D for each fixed point of p_G and get the optimum D for p_G as shown in the inner loop of Algorithm 1. Then keeping D fixed, we apply gradient descent to generator G and get closer to the saddle point (see the outer loop of Algorithm 1). Since partial derivatives of p_G at optimum D points include the saddle point, given enough capacity (i.e. computation power and training time) to discriminator and generator DNNs, we will eventually reach the saddle point using the training algorithm. Thus, Algorithm 1 converges the generated failure data distribution p_G to the real failure data distribution p_{data}, and hence reaches the unique global minimum provided in Theorem 1.

It must be noted that reaching the saddle point and hence achieving the optimal result for Theorem 1 can be impossible in practice due to the uncertainty associated with hyperparameter tuning and data noise. Nevertheless, the excellent performance of DNNs and empirical results presented in the next section show that this theoretical result holds to a highly satisfactory extent in practice [16].

III. CASE STUDY

The proposed methodology is used to address the problem of Scania air purge valve (APV) prognostics under the conditions of limited failure data availability. A detailed description of this prognostics problem and how the normalised Shannon entropy is used to measure the extent of limited failure data availability is provided in our conference paper (see Sec. V in [5]). Briefly, the prognostics problem is modelled as a binary classification task in which the challenge is to predict whether a truck faces an APV failure in the near future. The anonymised cost of an undetected APV failure is €500 (C_{FN}) and the anonymised cost of a false alarm is €10 (C_{FP}) [17]. The objective is to reduce the total cost of breakdowns and false alarms. Let m be the number of undetected APV failures and n be the number of false alarms, then the total cost of breakdowns and false alarms T_{Cost} is given by the following:

$$T_{\text{Cost}} = mC_{FN} + nC_{FP} \tag{7}$$

The positive class (i.e. data samples pertaining to APV failures) only covers 1.6% of the entire Scania training dataset, whereas the negative class (i.e. data samples not pertaining to APV failures) covers 98.4%. Thus, the Scania dataset is considered as a highly imbalanced dataset in the literature [18]. The normalised Shannon entropy of the dataset is 0.08 which also indicates a highly imbalanced dataset [5].

There are a few solutions already proposed in the literature for addressing the problem of Scania APV prognostics. The top three solutions and their performances are summarised in our conference paper (see Table II in [5]). In the remainder of this section, we show that using the proposed methodology for generating real-valued failure data one can obtain a far better result than all the existing solutions.

First a discussion on how the methodology is applied to address the problem of Scania APV prognostics under the conditions of limited failure data availability is provided. Then results obtained from convergence evaluation, overfitting assessment and prognostics performance evaluation introduced in Sec. II-B are discussed.

A. GENERATING REAL-VALUED FAILURE DATA FOR APV PROGNOSTICS

Before applying the methodology to generate real-valued failure data, we show that the Scania APV prognostics problem satisfies the prerequisite and assumption required for the methodology. The problem satisfies the prerequisite due to the following: similarity between trucks can be used as auxiliary information since the types of trucks and their purpose have an effect on degradation patterns of APV failures [17]. The problem satisfies the assumption due to the following: the failure mode that needs predicting is crack and it causes equipment to fail under degradation, hence the failure is not random [2].

In the remainder of this section, a discussion on how the three phases of the methodology are applied to address the problem of Scania APV prognostics under the conditions of limited failure data availability is provided.

1) Identifying auxiliary information pertaining to the failure mode and converting into a form for integrating into the failure data generation process

We employ similarity analysis to group similar trucks with APV failures in the training dataset and use these groups as auxiliary information to control and direct the failure data generation process. Since no information that can be used to group trucks (e.g. mileage, purpose, etc.) is provided with the dataset, clustering is used to identify natural groupings of trucks with APV failures. First, a subset D' that only contains failure data samples in the training dataset (i.e. the training failure data subset shown in Fig. 2) is created. Then k-means and hierarchical clustering algorithms are used to identify the natural groupings in D'.

Since prior knowledge about the number of clusters is not available, the average silhouette score is used to evaluate the quality of clustering. The best value of the average silhouette score is 1 and the worst value is -1. The negative values indicate that the samples are assigned to a wrong cluster and positive values indicate the samples are properly clustered. For both clustering algorithms, we obtained the best average silhouette scores when the number of clusters is two (also see [5]). The average silhouette scores obtained by k-means and hierarchical clustering algorithms are 0.55 and 0.53 respectively. Given the high diversity of Scania trucks in the dataset [17], it is reasonable to not expect the values obtained for the average silhouette score to be closer to 1 or the number of estimated clusters to match the ground truth.

In order to convert abstracted auxiliary information into a vector representation, we choose the class labels generated by the k-means algorithm since it has obtained the best average silhouette score. The class labels that represent the two groups with natural numbers 1 and 2 is a vector of natural numbers $Y = \{y \in \mathbb{N} | 1 \leq y \leq 2\}$.

2) Estimating a generative model that captures the semantic features of the failure mode and evaluating the convergence during training

In order to estimate a generative model that has captured the semantic features of APV failures in Scania trucks, we use the CGAN previously presented in Fig. 3. The generator G and discriminator D are DNNs. The Adaptive Moment Estimation (Adam) optimiser which is an extension to the stochastic gradient descent is used as the optimisation algorithm. The auxiliary information vector Y is the vector representation of class labels obtained in the previous phase. The training failure data subset X is the subset D' that was also obtained in the previous phase. The noise vector Z is Gaussian noise. Using these parameters as inputs to the CGAN, we train it to estimate the generative model using the Adam optimiser.

As previously discussed in the Theoretical Results section, the ability of the methodology to estimate a generative model which can replicate the real failure data distribution, and hence capture the semantic features of the failure mode is critical. In order to evaluate whether this theoretical result holds in practice, we evaluate the convergence during training using the K-S test introduced in Sec II-B. To reiterate, the objective of this test is to statistically identify whether the following null hypothesis can be rejected: the generated failure data distribution and real failure data distribution are statistically similar.

Since Scania dataset contains 170 features, principal component analysis (PCA) is used to reduce dimensionality. As shown in Table 2, PC-1 and PC-2 have a cumulative explained variance percentage of 74%. This means PC-1 and PC-2 alone capture 74% of information contains in the dataset. Hence, the generated failure data distribution and real failure data distribution are compared using these two principal components. The p-values observed for PC-1 and PC-2 K-S tests are 0.161 and 0.078 respectively. This indicates weak evidence against the null hypothesis. This means the null hypothesis cannot be rejected, thus the generative model converges the generated failure data distribution to the real failure data distribution.

TABLE 2. Comparison of Real and Generated Failure Data Distributions Using the Kolmogorov-Smirnov (K-S) Statistical Test

Principal component	Explained variance percentage	K-S test p-value
1	53%	0.161 ($>$ 0.05)
2	21%	0.078 ($>$ 0.05)

3) Generating real-valued failure data using the estimated generative model and assessing overfitting and evaluating prognostics performance

A new noise vector and the previously used auxiliary information vector are used as inputs to the estimated generative model to generate real-valued APV failure data. More specifically, given the joint distribution of noise and auxiliary information as the input, the generative model estimates a set of real-valued APV failure data samples. The lowest value for costs due to breakdowns and false alarms (T_{Cost}) is obtained when the number of generated failure data samples is 2000. The original training dataset is then augmented to include the newly generated failure data samples. The positive and negative sample ratio in the augmented training dataset is 3000:59000 compared to the 1000:59000 in the original training dataset. Moreover, the normalised Shannon entropy is now increased from 0.08 to 0.2, hence the extent of the limited failure data availability problem is reduced.

In order to evaluate the generated failure data samples, we use the overfitting assessment and prognostics performance evaluation introduced in Sec. II-B. First, we evaluate whether the generative model is overfitting to training failure data by identifying whether the following null hypothesis can be rejected: the generative model is not overfitting to training failure data. If this null hypothesis can be retained, the MMD between generated failure data samples and test failure data samples is at most as large as the MMD between generated failure data samples and training failure data samples. The p-value observed for this statistical test is 0.38 (> 0.05) which indicates weak evidence against the null hypothesis. This means the null hypothesis cannot be rejected, thus the generative model is not overfitting to training failure data.

The prognostics performance is evaluated using the random forest (RF) and gradient boosting machine (GBM) classifiers. RF-based prognostics solutions have been previously successful in predicting Scania APV failures using the same dataset (e.g. [17] and [18]). GBM is another popular ensemble method for developing classification-based prognostics solutions, especially when datasets are imbalanced [19]. We implemented two prognostics models using the GBM and RF classifiers. When trained on the original training dataset and evaluated on the testing dataset, these models obtained total costs (T_{Cost}) of €10750 and €11090 respectively.

The models are then trained on the augmented training dataset and evaluated on the testing dataset. Fig. 5 shows reliability-based confusion matrixes obtained for GBM and RF classifier-based prognostics models when trained on the augmented training dataset and evaluated on the testing dataset. The T_{Cost} achieved by the GBM and RF-based prognostics solutions are €5550 and €6050 respectively. Compared to the performance obtained when trained on the original training dataset, this is a 48% (GBM) and 46% (RF) reduction of T_{Cost}. More importantly, compared to the performance obtained by the best prognostics solution previously proposed in the literature (i.e. the benchmark), this is a 44% (GBM) and 39% (RF) reduction of T_{Cost}.

The reason for this cost reduction can be observed using Fig. 6. It can be observed that when the prognostics models are trained on the augmented training dataset, they achieve higher precisions and recalls compared to the benchmark. Note that precision and recall are normalised between 0 and 1, hence a slight increase in these metrics will lead to a high number of accurate predictions and a low number of false alarms.

	Will maintain	Will not maintain
Failures	TRUE POSITIVES (TP) 371 instances	FALSE NEGATIVES (FN) 4 instances €2000 loss
Non-failures	FALSE POSITIVES (FP) 405 instances €4050 loss	TRUE NEGATIVES (TN) 15220 instances

Random Forest

	Will maintain	Will not maintain
Failures	TRUE POSITIVES (TP) 369 instances	FALSE NEGATIVES (FN) 6 instances €3000 loss
Non-failures	FALSE POSITIVES (FP) 255 instances €2550 loss	TRUE NEGATIVES (TN) 15370 instances

Gradient Boosting Machine

FIGURE 5. Reliability-based confusion matrixes summarising the performance of gradient boosting machine (GBM) and random forest (RF) classifiers-based prognostics models. Compared to the benchmark, GBM and RF-based prognostics models produced 44% and 39% costs savings when trained on the augmented training dataset and evaluated on the testing dataset.

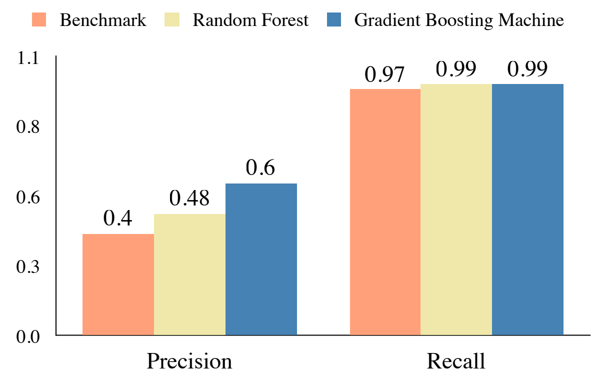

FIGURE 6. Plot summarising prognostics performance achieved by the gradient boosting machine (GBM) and random forest (RF) classifiers-based prognostics models when trained on the augmented training dataset and evaluated on the testing dataset. It can be observed that both prognostics models outperform the benchmark by achieving higher precisions and recalls.

IV. CONCLUSION

The problem of limited failure data availability for prognostics is long-lasting and challenging. Existing techniques used to address this problem have been unsuccessful since they either duplicate existing failure data or randomly generate data (i.e. the failure data generation process is not controlled and directed, hence leads to different modes of data being generated). The research presented in this paper starts to address this problem from a novel perspective by developing a methodology that is capable of generating new and realistic failure data samples.

The methodology estimates a generative model in a minimax game which captures the semantic features of failure modes using real failure data, noise and more importantly, using auxiliary information pertaining to the failure modes (e.g. expert knowledge, physics of failure and information contains within maintenance records). The utilisation of auxiliary information allows the methodology to condition the noise being added to newly generated data samples, thus the failure data generation process is controlled and directed.

Whilst theoretical results presented in the paper provide the formal grounding to the methodology, empirical results obtained using the real-world case study show that the theoretical results hold in practice. The methodology outperformed the best solution previously proposed in the literature for the real-world case study by producing 44% cost savings.

We introduced following methods for evaluating key aspects of the methodology: (i) convergence evaluation which statistically tests whether the generative model converges the generated failure data distribution to the real failure data distribution during the minimax game; (ii) overfitting assessment which statistically identifies whether the generative model is overfitting to training failure data during the minimax game; (iii) prognostics performance evaluation which uses precision and recall to identify the increase in predictive power of prognostics models when they are trained on the augmented training dataset which includes real and generated failure data.

The key components that are unique to the proposed methodology and enhance prognostics performance include the following: (i) integration of auxiliary information pertaining to failure modes to control and direct the failure data generation process; (ii) utilisation of a conditional generative adversarial network which provides the platform to estimate a generative model in a minimax game; (iii) estimation of a generative model that captures the semantic features of the failure mode that needs predicting using real failure data, noise and auxiliary information.

As per the future work, we intend to further develop and generalise the methodology using a diverse set of real-world case studies.

.

APPENDIX A PROOF OF LEMMA 1

Lemma: Given the joint probability distribution of real failure data $p_{\text{data}}(x, y)$ and generated failure data $p_G(x, y)$, where $\{x \in X\}$ consists of real and generated input failure data samples and $\{y \in Y\}$ is the auxiliary information vector, the optimal discriminator D^*_G for the fixed generator is,

$$D^*_G(x \mid y) = \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)}$$

Proof: Consider the value function $V(G, D)$ of the minimax game in CGAN,

$$V(G, D) = \mathbb{E}_{x \sim p_{\text{data}}}[\log D(x \mid y)] + \mathbb{E}_{z \sim p_{\text{noise}}}[\log(1 - D(G(z \mid y)))]$$

Using the law of unconscious statistician (LOTUS) theorem,

$$\mathbb{E}_{z \sim p_{\text{noise}}}[\log(1 - D(G(z \mid y)))] = \mathbb{E}_{x \sim p_G}[\log(1 - D(x \mid y))]$$

Therefore $V(G, D)$ can be written as,

$$V(G, D) = \mathbb{E}_{x \sim p_{\text{data}}}[\log D(x \mid y)] + \mathbb{E}_{x \sim p_G}[\log(1 - D(x \mid y))]$$

For a pair of continuous random variables X and Y with a joint probability distribution $Pr(x = X, y = Y)$, the expected value $\mathbb{E}$ can be found using an arbitrary function of the continuous variables $g(X, Y)$ such that,

$$\mathbb{E}[g(X, Y)] = \int_x \int_y p(x, y)\, g(x, y)\, dx\, dy$$

Therefore, $V(G, D)$ can be written as,

$$V(G, D) = \int_x \int_y p_{\text{data}}(x, y) \log D(x \mid y)\, dx\, dy + \int_x \int_y p_G(x, y) \log(1 - D(x \mid y))\, dx\, dy$$

Using the sum rule of integration,

$$V(G, D) = \int_x \int_y p_{\text{data}}(x, y) \log D(x \mid y) + p_G(x, y) \log(1 - D(x \mid y))\, dx\, dy$$

For clarity, we label the above equation as follows:

$$h = D(x \mid y),\, a = p_{\text{data}}(x, y),\, b = p_G(x, y)$$

Since the sample x given y is sampled over all the possible values, the integrals can be safely ignored for the remainder of the proof. Therefore $V(G, D)$ becomes,

$$f(h) = a \log h + b \log(1 - h)$$

The objective is to find the best value for $D(x \mid y)$ to maximise the value function $V(G, D)$. The maximum of $V(G, D)$ can be found as follows: (i) differentiate $V(G, D)$ w.r.t $D(x \mid y)$ and equate to zero to find the critical points; (ii) perform the second derivative test for local extrema (i.e. h is maximum if $f''(h) < 0$) to find the maximum. Hence, we first differentiate $f(h)$ w.r.t to h,

$$f'(h) = \frac{a}{h} - \frac{b}{1 - h}$$

Equating $f'(h)$ to zero to find the critical points,

$$\frac{a}{h} - \frac{b}{1 - h} = 0$$

Using calculus,

$$h = \frac{a}{a + b}$$

if $a + b \neq 0$. Finding $f''(h)$ to identify whether h is the maximum using second derivative test,

$$f''(\frac{a}{a + b}) = -\frac{a}{a/(a + b)^2} - \frac{b}{(1 - (a/a + b))^2}$$

Since $f''(h) < 0$, h is the maximum. Hence, for the fixed generator G the optimal discriminator $D^*_G(x \mid y) = h$. After substituting a and b labels in h with their corresponding values, $D^*_G(x \mid y)$ can be written as,

$$D^*_G(x \mid y) = \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)}$$

, concluding the proof. □

APPENDIX B PROOF OF THEOREM 1

Theorem: The training criterion $C(G) = \min_G \max_D V(G, D)$ achieves a unique global minimum if and only if the generated failure data distribution p_G is equal to the real failure data distribution p_{data}.

Proof: Assuming $p_G = p_{\text{data}}$ and using Lemma 1, the optimal discriminator $D^*_G(x \mid y)$ is,

$$D^*_G(x \mid y) = \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_{\text{data}}(x, y)} = \frac{1}{2}$$

Consider the integral form of the value function $V(G, D)$ introduced in Lemma 1,

$$V(G, D) = \int_x \int_y p_{\text{data}}(x, y) \log D(x \mid y) + p_G(x, y) \log(1 - D(x \mid y))\, dx\, dy$$

When $D(x \mid y) = D^*_G(x \mid y) = 1/2$, $C(G)$ is,

$$C(G) = \int_x \int_y p_{\text{data}}(x, y) \log \frac{1}{2} + p_G(x, y) \log(1 - \frac{1}{2})\, dx\, dy$$

As per the assumption made in the beginning of the proof, $p_G = p_{\text{data}}$ when $D^*_G(x \mid y) = 1/2$. Hence,

$$C(G) = \int_x \int_y p_{\text{data}}(x, y) \log \frac{1}{2} + p_G(x, y) \log(1 - \frac{1}{2})\, dx\, dy = -2 \log 2$$

This means $-2 \log 2$ is a candidate for the global minimum of the training criterion $C(G)$. However, we still need to prove that this is the only global minimum of $C(G)$ and it is achieved only when the generated failure data distribution p_G is equal to the real failure data distribution p_{data}. Hence, we first drop the assumption $p_G = p_{\text{data}}$. Again consider the integral form of the value function $V(G, D)$ introduced in Lemma 1,

$$V(G, D) = \int_x \int_y p_{\text{data}}(x, y) \log D(x \mid y) + p_G(x, y) \log(1 - D(x \mid y))\, dx\, dy$$

When $D(x \mid y) = D^*_G(x \mid y)$, $V(G, D)$ is,

$$V(G, D^*_G) = \int_x \int_y p_{\text{data}}(x, y) \log D^*_G(x \mid y) + p_G(x, y) \log(1 - D^*_G(x \mid y))\, dx\, dy$$

For the training criterion $C(G) = V(G, D^*_G)$ and since $D^*_G(x \mid y)$ is as per the equation proved in Lemma 1, $C(G)$ is,

$$C(G) = \int_x \int_y p_{\text{data}}(x, y) \log \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)} + p_G(x, y) \log(1 - \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)})\, dx\, dy$$

After using calculus to rearrange the second part of the integral,

$$C(G) = \int_x \int_y p_{\text{data}}(x, y) \log \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)} + p_G(x, y) \log \frac{p_G(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)}\, dx\, dy$$

Since we know $-2 \log 2$ is a candidate for the global minimum, we integrate this value into $C(G)$ by adding and subtracting $\log 2$ and multiplying by the joint probability densities. This do not change the equality of the equation since ultimately we are adding 0 to the equation. Therefore,

$$C(G) = \int_x \int_y (\log 2 - \log 2) p_{\text{data}}(x, y) + p_{\text{data}}(x, y) \log \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)} + (\log 2 - \log 2) p_G(x, y) + p_G(x, y) \log \frac{p_G(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)}\, dx\, dy$$

Using calculus $C(G)$ can be rearranged as,

$$C(G) = -\log 2 \int_x \int_y p_G(x, y) + p_{\text{data}}(x, y)\, dx\, dy + \int_x \int_y p_{\text{data}}(x, y)(\log 2 + log \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)}) + p_G(x, y)(\log 2 + \log \frac{p_G(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)})\, dx\, dy$$

The definition of probability densities states that integrating two probability distributions over their domain is equal to 1. Hence,

$$-\log 2 \int_x \int_y p_G(x, y) + p_{\text{data}}(x, y)\, dx\, dy = -\log 2(1 + 1) = -2 \log 2$$

Moreover, using the definition of the logarithm,

$$\log 2 + log \frac{p_{\text{data}}(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)} = log \frac{p_{\text{data}}(x, y)}{(p_{\text{data}}(x, y) + p_G(x, y))/2}$$

Similarly,

$$\log 2 + \log \frac{p_G(x, y)}{p_{\text{data}}(x, y) + p_G(x, y)} = \log \frac{p_G(x, y)}{(p_{\text{data}}(x, y) + p_G(x, y))/2}$$

Substituting above equalities into $C(G)$,

$$C(G) = -2log2 \int_x \int_y p_{\text{data}}(x, y) log \frac{p_{\text{data}}(x, y)}{(p_{\text{data}}(x, y) + p_G(x, y))/2} + p_G(x, y) log \frac{p_G(x, y)}{(p_{\text{data}}(x, y) + p_G(x, y))/2}\, dx\, dy$$

Using the sum rule of integration $C(G)$ can be written as,

$$C(G) = -2log2 \int_x \int_y p_{\text{data}}(x,y) log(\frac{p_{\text{data}}(x,y)}{(p_{\text{data}}(x,y)+p_G(x,y))/2}) \, dx \, dy + \int_x \int_y p_G(x,y) log(\frac{p_G(x,y)}{(p_{\text{data}}(x,y)+p_G(x,y))/2}) \, dx \, dy$$

Using the Kullback-Leibler divergence (KL),

$$C(G) = -2log2 + KL(p_{\text{data}}| \frac{p_{\text{data}}+p_G}{2}) + KL(p_G| \frac{p_{\text{data}}+p_G}{2})$$

Using the Jenson-Shannon divergence (JSD),

$$C(G) = -2log2 + 2.JSD(p_{\text{data}}||p_{\text{G}})$$

Rearranging $C(G)$ using calculus,

$$C(G) - (-2log2) = 2.JSD(p_{\text{data}}||p_{\text{G}})$$

JSD between two distributions is non-negative. Hence,

$$C(G) - (-2log2) \geq 0$$

Therefore, we have a unique global minimum for the training criterion $C(G)$ when JSD is equal to 0,

$$C(G)_{min} = -2log2$$

This means the training criterion $C(G)$ can achieve a unique global minimum of $-2\log 2$ when the optimal discriminator $D_G^*(x \mid y) = 1/2$. Since $D_G^*(x \mid y) = 1/2$ only when $p_G = p_{\text{data}}$, the global minimum is achieved only when the generated failure data distribution p_G is equal to the real failure data distribution p_{data}, concluding the proof. □

REFERENCES

[1] A. K. Jardine, D. Lin, and D. Banjevic, "A review on machinery diagnostics and prognostics implementing condition-based maintenance," Mechanical systems and signal processing, vol. 20, no. 7, pp. 1483–1510, 2006.

[2] Y. Peng, M. Dong, and M. J. Zuo, "Current status of machine prognostics in condition-based maintenance: a review," The International Journal of Advanced Manufacturing Technology, vol. 50, no. 1-4, pp. 297–313, 2010.

[3] E. Burnaev, "Rare failure prediction via event matching for aerospace applications," arXiv preprint arXiv:1905.11586, 2019.

[4] B. de Jonge and P. A. Scarf, "A review on maintenance optimization," European Journal of Operational Research, 2019.

[5] G. D. Ranasinghe and A. K. Parlikad, "Generating real-valued failure data for prognostics under the conditions of limited data availability," in 2019 IEEE International Conference on Prognostics and Health Management (ICPHM). IEEE, 2019, pp. 1–8.

[6] S. Alestra, C. Brand, E. Burnaev, P. Erofeev, A. Papanov, C. Bordry, and C. Silveira-Freixo, "Rare event anticipation and degradation trending for aircraft predictive maintenance," in 11th World Congress on Computational Mechanics, WCCM, 2014, pp. 6571–6582.

[7] G. M. Weiss, "Mining with rarity: a unifying framework," ACM Sigkdd Explorations Newsletter, vol. 6, no. 1, pp. 7–19, 2004.

[8] M. Mirza and S. Osindero, "Conditional generative adversarial nets," arXiv preprint arXiv:1411.1784, 2014.

[9] S. Gurumurthy, R. Kiran Sarvadevabhatla, and R. Venkatesh Babu, "Deligan: Generative adversarial networks for diverse and limited data," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 166–174.

[10] M. Pecht and J. Gu, "Physics-of-failure-based prognostics for electronic products," Transactions of the Institute of Measurement and Control, vol. 31, no. 3-4, pp. 309–322, 2009.

[11] M. Pecht, "Prognostics and health management of electronics," Encyclopedia of Structural Health Monitoring, 2009.

[12] A. K. Jardine and A. H. Tsang, Maintenance, replacement, and reliability: theory and applications. CRC press, 2005.

[13] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," in Advances in neural information processing systems, 2007, pp. 513–520.

[14] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (medical) time series generation with recurrent conditional gans," arXiv preprint arXiv:1706.02633, 2017.

[15] S. M. A. Elrahman and A. Abraham, "A review of class imbalance problem," Journal of Network and Innovative Computing, vol. 1, no. 2013, pp. 332–340, 2013.

[16] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in Advances in neural information processing systems, 2014, pp. 2672–2680.

[17] J. Biteus and T. Lindgren, "Planning flexible maintenance for heavy trucks using machine learning models, constraint programming, and route optimization," SAE International Journal of Materials and Manufacturing, vol. 10, no. 2017-01-0237, pp. 306–315, 2017.

[18] C. F. Costa and M. A. Nascimento, "Ida 2016 industrial challenge: Using machine learning for predicting failures," in International Symposium on Intelligent Data Analysis. Springer, 2016, pp. 381–386.

[19] Z. Wu, W. Lin, and Y. Ji, "An integrated ensemble learning model for imbalanced fault diagnostics and prognostics," IEEE Access, vol. 6, pp. 8394–8402, 2018.

GISHAN D. RANASINGHE is a doctoral student at the Institute for Manufacturing, Department of Engineering, University of Cambridge. In his research, he is developing a methodology for predicting equipment failure under the conditions of limited failure data availability and showing its impact on the predictive maintenance of industrial equipment. He holds bachelor's and master's degrees in Computer Science and Artificial Intelligence from the University of Nottingham.

Mr D. Ranasinghe is the recipient of Best Paper Award at the 2019 IEEE International Conference on Prognostics and Health Management, San Francisco, USA.

TONY LINDGREN is an associate professor at the Department of Computer and Systems Sciences, Stockholm University and a Consultant Research Scientist at the Scania Commercial Vehicles, Service Support Solutions. In 2006 he received his PhD degree in Computer and System Sciences. He has worked both in academia and industry since 2008, he is the inventor of numerous patents and the author of numerous scientific articles.

During his time at Scania, Dr Lindgren is working on applying his research to solve practical problems such as the maintenance optimisation of Scania heavy-trucks. The optimisation settings are for matching trucks with transport missions to create transport routes that maximise long-term profits for customers and optimising maintenance activities to align with the intended usage of trucks. He has a permanent position as lecturer at the Department of Computer and System Sciences at Stockholm University since 2012. His main interest is in the field of machine learning, artificial intelligence and constraint programming.

MARK GIROLAMI is a computational statistician having ten years of experience as a Chartered Engineer within IBM. In March 2019 he was elected to the Sir Kirby Laing Professorship of Civil Engineering (1965) within the Department of Engineering at the University of Cambridge where he also holds the Royal Academy of Engineering Research Chair in Data-Centric Engineering. Before joining the University of Cambridge, Professor Girolami held the Chair of Statistics in the Department of Mathematics at Imperial College London.

Professor Girolami was one of the original founding Executive Directors of the Alan Turing Institute the UK's national institute for Data Science and Artificial Intelligence, after which he was appointed as Strategic Programme Director at Turing. He also serves as the Editor-in-Chief of Statistics and Computing journal and the new open-access journal Data-Centric Engineering published by Cambridge University Press.

AJITH K. PARLIKAD is Reader in Asset Management at Cambridge University Engineering Department. He is based at the Institute for Manufacturing, where he is the Head of the Asset Management research group. He is a Fellow and Tutor at Hughes Hall.

Dr Parlikad leads research activities on engineering asset management and maintenance. His particular focus is examining how asset information can be used to improve asset performance through effective decision-making. He actively engages with industry through research and consulting projects. He is a member of the steering committees of the IFAC Working Group on Advanced Maintenance Engineering, Services and Technology and the UK Digital Twin Hub.

• • •