Building Degradation Index with Variable Selection for Multivariate Sensory Data

Yueyao Wang[1], I-Chen Lee[2*], Yili Hong[1] and Xinwei Deng[1]

[1]Department of Statistics, Virginia Tech, Blacksburg, VA 24061

[2]Department of Statistics, National Cheng Kung University, Tainan, Taiwan 70101

[*]Corresponding author: I-Chen Lee, +886-6-2757575-53642, iclee@ncku.edu.tw

Abstract

The modeling and analysis of degradation data have been an active research area in reliability engineering for reliability assessment and system health management. As the sensor technology advances, multivariate sensory data are commonly collected for the underlying degradation process. However, most existing research on degradation modeling requires a univariate degradation index to be provided. Thus, constructing a degradation index for multivariate sensory data is a fundamental step in degradation modeling. In this paper, we propose a novel degradation index building method for multivariate sensory data with censoring. Based on an additive nonlinear model with variable selection, the proposed method can handle censored data, and can automatically select the informative sensor signals to be used in the degradation index. The penalized likelihood method with adaptive group penalty is developed for parameter estimation. We demonstrate that the proposed method outperforms existing methods via both simulation studies and analyses of the NASA jet engine sensor data.

Key Words: Adaptive LASSO; General Path Model; Prognostics; Sensor Selection; Splines; System Health Monitoring.

Nomenclature and Abbreviations

cdf	cumulative distribution function
pdf	probability density function
LEV	largest extreme value distribution
LASSO	least absolute shrinkage and selection operator
k-CV	k-fold cross validation
FNR	false negative error rates
FPR	false positive error rates
TER	total error rates, summation of FPR and FNR
DI-VS	the proposed degradation index with variable selection
DI-NVS	degradation index without variable selection
DI-VSL	degradation index with variable selection and linear form
DI-KSL	degradation index proposed by [1]
$\boldsymbol{x}(t)$	the history of multi-channel sensor signals from a unit
$u(t)$	cumulative exposure
$f_j[x_j(t); \boldsymbol{\beta}_j]$	the effect of jth signal on the degradation process
$h(z)$	a mapping function that ensures a positive exposure
n, p, m	number of units, number of sensors, and number of splines
U	a random failure threshold
α, $\tilde{\alpha}$	the target failure threshold and the practical failure threshold
$\log(\alpha)$, σ	location and scale parameters of a location-scale distribution
t_i, δ_i	failure time and censoring indicator for unit i
$\phi(\cdot)$, $\Phi(\cdot)$	pdf and cdf of a location-scale distribution
$\boldsymbol{\beta}$	the coefficient vector of spline basis in the model
L_i, l_i	the likelihood and log-likelihood function of unit i
ω_j	the weights of jth sensor in the adaptive group LASSO penalty
σ_l	the lower bound for the scale parameter σ
λ	tuning parameter of the adaptive group LASSO penalty
η, γ	hyper-parameters in the objective function
z_p	p quantile of a standard location-scale distribution

1 Introduction

1.1 Background

Degradation data have been widely used in reliability engineering for reliability and system health assessment. There are many examples of products and systems that provide degradation data, such as the loss of light output from a light-emitting diode (LED) array, the power output decrease of photovoltaic arrays, and the vibration from a worn bearing in a wind turbine. The data type is typically a repeated measurement of the degradation index (e.g., the loss of light output from an LED array) with a monotone increasing (decreasing) trend. The general path model is one class of methods for degradation modeling. In the typically modeling framework of the general path models, a soft failure occurs when the degradation level reaches a predefined failure threshold. The stochastic process model framework is also popular in the degradation literature. In the stochastic process models, the distribution of the degradation incremental levels is modeled by a Gaussian distribution or other distributions such as the gamma or inverse Gaussian distribution. Thus, in existing research on degradation modeling, the degradation index is needed and can be measured over time.

Different from traditional degradation data, modern sensor technology allows one to collect multi-channel sensor data that are related to an underlying degradation process. This is very common in many modern engineering systems. One example is the motivating data in our study, the multi-channel jet engine data ([2]). In the jet engine data, multiple sensors such as temperatures and pressures of module parts are recorded while the engine is operating. The details about the dataset are introduced in Section 5.1. In such multi-channel sensor data, any single channel may not be sufficient to represent the underlying degradation process. Without a degradation index, most existing methods in the aforementioned frameworks will not be applicable in the analysis of such sensor signal data. Thus, building a degradation index is an important step in utilizing the sensor data in degradation analysis.

There are several key considerations when building the degradation index based on multi-sensory data. In real engineering applications, it is typically complicated how each sensor signal reflects the overall degradation. In such a case, a linear form for the effect of each sensor signal may not be adequate, which motivates us to consider nonlinear functional forms of the individual sensor signal. In most cases, not all sensors collected are useful in representing the underlying degradation process. Thus, it is important to automatically select more useful and relevant sensors to build the degradation index. Besides, censored data are quite common in reliability engineering applications. It is ideal to use both exact failure and censored time data in the training of the model. In addition, when considering the prediction accuracy, the risks of being false positive and false negative are quite different, especially for failures of important systems such as jet engines. An asymmetric loss function is desirable during the training of the degradation index model.

Motivated by these considerations, this paper aims to develop a flexible method for constructing a degradation index from multi-channel signals with automatic variable selection while accounting for censored data and nonlinear relationships.

1.2 Literature Review and Contribution of This Work

Regarding the general path models for degradation data, the classic reference book is Meeker and Escobar [3]. For the stochastic process models for degradation data, the Wiener process, gamma process, and the inverse-Gaussian process have been used (e.g., [4]). Ye and Xie [5] provided a comprehensive review of degradation models. These models are suitable for the cases of one-dimensional degradation data. On the other hand, Meeker and Hong [6] and Hong et al. [7] outlined some opportunities for using sensor data in reliability modeling and analysis.

Regarding the recent development of degradation modeling, Zheng et al. [8] considered the joint modeling of degradation data and lifetime data using the proportional hazards model. Wang et al. [9] developed a Wiener process model to describe heterogeneity in degradation data. Chen et al. [10] developed an integration method of multi-source accelerated degradation testing for reliability evaluation. Duan et al [11] proposed an adaptive monitoring scheme based on the hidden Markov model to predict the faults of systems with hidden degradation processes. Kumar et al. [12] proposed a health indicator based on the state-space model to access the degradation process. Wang et al. [13] constructed a stochastic multi-phase model for multi-component systems. In summary, the modeling and analysis of degradation data is an active area, which is carried out with the availability of degradation measurements.

In the area of degradation index building, Liu et al. [14] proposed a data-level fusion model for developing composite health indices for degradation modeling and prognostic analysis. Follow-up work includes [15, 16, 17, 18]. Recently, Kim et al. [1] proposed a latent linear model that constructs a health index via multiple sensors and selects informative sensors. Wei et al. [19] proposed a dynamic conditional variational autoencoder to learn the health index. Kim et al. [20] developed a sensor selection framework that can be applied to neural network-based models and improved in interpretability of neural network models. Li et al. [21] developed a data-model interactive prediction method for multi-sensor monitored stochastic degrading devices. However, existing methods can not handle censored data and conduct variable selection at the same time, which is the gap that this paper aims to fill.

Regarding variable selections, the least absolute shrinkage and selection operator (LASSO) penalty for regression-type problems was studied in [22]. Zou [23] developed the adaptive LASSO to ensure variable selection consistency, and Yuan and Lin [24] considered the group LASSO for efficient variable selections with meaningful interpretation. Under the context of this paper, we focus on the adaptive group LASSO to select the important sensors.

This paper proposes a novel framework based on the cumulative exposure model to build the degradation index with multivariate sensors, which is applicable to many engineering systems equipped with sensors. The contributions of this paper are unique from existing literature. Specifically, the proposed framework can include censored failure time information to train the model, which can preserve the information provided by the data. The proposed framework can automatically select the most informative sensors related to the degradation process using the adaptive group LASSO penalty. To enable sufficient flexibility in the non-linear relationship between sensors and degradation path, spline-based methods are used to describe the contribution of each sensor signal in the cumulative exposure.

1.3 Overview

The rest of this paper is organized as follows. Section 2 introduces the framework for degradation index building based on time-to-event data with multivariate signals. Section 3 presents the parameter estimation with variable selection. Section 4 uses simulation to study the performance of the proposed methods. The motivating example is used to illustrate the developed method in Section 5. Section 6 contains conclusions and discusses areas for future research.

2 Building Degradation Index

2.1 Degradation Index

Consider sensor data with p degradation signals. Let $\boldsymbol{x}(t) = \{[x_1(s), \cdots, x_p(s)]' : 0 \leq s \leq t\}$ be the collection of information for the p signals from a unit, where $x_j(s)$ is the jth the dynamic covariate information at time s, $j = 1, \ldots, p$. For the jet engine data, those $x_j(\cdot)$'s can be signals from various sensors and recorded operating conditions. We use the cumulative exposure model (aka, the cumulative damage model) to construct the degradation index (e.g., see [25]). The cumulative exposure model is useful when the damage is accumulative, which is the case for many engineering systems. For example, the damage to an automobile tire is accumulative due to wear out.

The cumulative exposure $u(t)$ for the covariate history $\boldsymbol{x}(t)$ is defined as,

$$u(t) = \int_0^t h\left\{\sum_{j=1}^{p} f_j[x_j(s); \boldsymbol{\beta}_j]\right\} ds, \tag{1}$$

where $f_j[x_j(t); \boldsymbol{\beta}_j]$ represents the nonlinear effect function of the signal $x_j(t)$ on the degradation index and $\boldsymbol{\beta}_j$ are parameters that represent the influence of the covariate on the cumulative exposure. Note that $f_j(\cdot; \boldsymbol{\beta}_j)$ is defined on the range of the covariate, not on time t. Here, $h(z)$ maps the effect to a positive exposure, and the integral is from 0 to t, which guarantees that $u(t)$ is monotonically increasing and utilizes all the information in the history up to time t. In this paper, we use the function $h(z) = \log[1 + \exp(z)]/\log(2)$, which maps the input $z \in (-\infty, \infty)$ to an output that takes value $h(z) \in (0, \infty)$. In literature, functions like $h(z) = \exp(z)$ are used, but the function $h(z)$ used here is numerically more stable.

When modeling the nonlinear effect of the jth signal $f_j(\cdot; \boldsymbol{\beta}_j)$ in $u(t)$, it is desirable to make the function form flexible enough to capture potential nonlinearity in sensors' effect. Therefore, we use a non-negative spline function, called M-splines (e.g., [26]). For the jth signal, let $f_j[x_j(t); \boldsymbol{\beta}_j] = \sum_{k=1}^{m} \beta_{jk}\gamma_{jk}[x_j(t)]$, where $\{\gamma_{jk}[x_j(t)] : k = 1, \ldots, m\}$ are spline basis of the M-spline of order three with $(m - 3)$ interior knots and $\boldsymbol{\beta}_j = (\beta_{j1}, \ldots, \beta_{jm})'$ are the coefficients of the basis. Let $\boldsymbol{\beta} = (\boldsymbol{\beta}'_1, \ldots, \boldsymbol{\beta}'_p)'$ be the parameters for all p signals. Figure 1(a) shows the basis of the M-spline of order 3 with 7 interior knots ($m = 10$). The magnitude of $\boldsymbol{\beta}$ can be used to identify which signals are more important for the degradation index $u(t)$. Although we used the M-splines for implementation, other splines such as the B-splines can also be used, because most spline basis functions are quite flexible.

Note that $u(0) = 0$, and $u(t)$ is always monotonically increasing, which are the two properties of $u(t)$ that satisfy the characteristics of a degradation index as introduced in Section 1.1. Thus, in this paper, we propose to use $u(t)$ as a degradation index.

2.2 Modeling Time to Failure and Degradation Index

Based on the cumulative exposure model, a unit fails at time T when the cumulative exposure reaches a random threshold U (e.g., [25]). That is

$$U = u(T), \tag{2}$$

where the function $u(\cdot)$ is defined in (1). We model $\log(U)$ by the largest extreme value (LEV) distribution (e.g., [3]) with the location parameter $\log(\alpha)$ and the scale parameter $\sigma > 0$. The cumulative distribution function (cdf) and the probability density function (pdf) of U are

$$G_U(u; \alpha, \sigma) = \Phi_{\text{LEV}}\left[\frac{\log(u) - \log(\alpha)}{\sigma}\right] \text{ and } g_U(u; \alpha, \sigma) = \frac{1}{\sigma u}\phi_{\text{LEV}}\left[\frac{\log(u) - \log(\alpha)}{\sigma}\right],$$

where $\Phi_{\text{LEV}}(x) = \exp[-\exp(-x)]$, and $\phi_{\text{LEV}}(x) = \exp[-x - \exp(-x)]$. As an illustration, Figure 1(b) shows the pdf of LEV distributions with $\alpha = \exp(5)$ and $\sigma = 0.01$, 0.03, and 0.1.

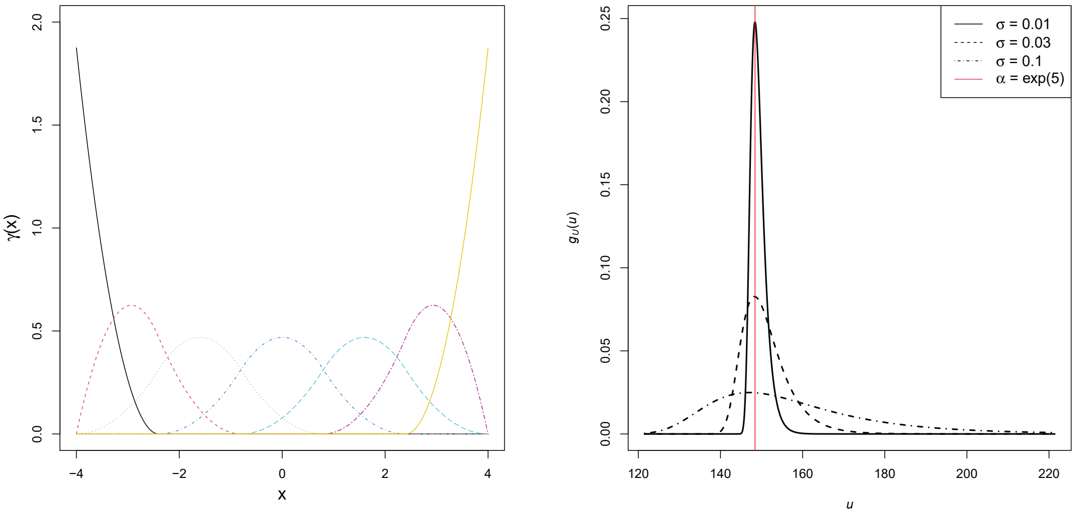
The parameter α in the LEV distribution can be used as the *target failure threshold* for the degradation index in (1), which is pre-fixed. This is because we want those failed units with their degradation indexes $u(t)$ centered around the failure threshold when they fail. The scale parameter σ serves as a measurement of how small the difference between $u(t)$ and the threshold α is if the unit is failed. Because we do not observe U, in practice, we use the following threshold rule. That is, if $u(t) \geq \alpha$, we say a unit fails, and if $u(t) < \alpha$, we say the unit is surviving (i.e., the operation status is normal). Through re-scaling, we can map the degradation index to any range that is desirable for the particular application. For example, using $100u(t)/\alpha$, one can map the normal range of the degradation index into $[0, 100]$.

For building the degradation index, a suitable property of the LEV distribution is its skewness to the right as shown in Figure 1(b). Thus, we allow the larger difference between $u(t)$ and α on the positive side so that when making predictions, we can potentially reduce false negative error (i.e., falsely predicting failed units as censored). This is desirable because, in our study, we focus on important equipment like the jet engine. Falsely predicting a failed unit as censored (i.e., false negative) can result in great losses or even severe accidents. Note that other location-scale distributions with the right skewness could also be considered. Through the relationship between T and U as shown in (2), the cdf and the pdf of T are, $G_T(t;\boldsymbol{\beta}) = \Phi_{\text{LEV}}\left(\{\log[u(t)] - \log(\alpha)\}/\sigma\right)$, and $g_T(t;\boldsymbol{\beta}) = \{u'(t)/[\sigma u(t)]\}\phi_{\text{LEV}}\left(\{\log[u(t)] - \log(\alpha)\}/\sigma\right)$, where the derivative of of $u(t)$ is $u'(t) = h\left(\sum_{j=1}^{p}\sum_{k=1}^{m}\beta_{jk}\gamma_{jk}[x_j(t)]\right)$.

3 Parameter Estimation

3.1 Log-likelihood with Adaptive Group LASSO Penalty

We first introduce some notation for the data. From the sensor data, there are n units with p degradation signals. Three sets of observable data are taken into consideration, which include failure-time data, censoring indicator, and multivariate degradation signals. Let t_i be the failure time of the ith unit and let $\boldsymbol{x}_i(t) = \{[x_{i1}(s), \cdots, x_{ip}(s)]' : 0 \leq s \leq t\}$ be the collection



(a) M-spline Basis (b) Probability Density Function

Figure 1: The plots show the construction of the M-spline of order 3 with 7 interior knots (a), and the pdf of LEV distributions with $\alpha = \exp(5)$ and various values of σ (b).

of p signals for the ith unit prior to and at time t. Here $x_{ij}(s)$ is the jth observed dynamic covariate information at time s, $i = 1, \ldots, n$, $j = 1, \ldots, p$, and $0 \leq s \leq t_i$. The observed censoring indicator of the ith unit is denoted by δ_i, where δ_i equals to 1 if the ith unit fails and 0 otherwise. Then, the collected information from unit i is denoted by $\{t_i, \delta_i, \boldsymbol{x}_i(t_i)\}$, where $i = 1, \ldots, n$, and let $\boldsymbol{H} = \{\{t_i, \delta_i, \boldsymbol{x}_i(t_i)\} : i = 1, \ldots, n\}$ be the collection of the data. Note that for training the degradation index, lifetime data with failures are necessary for our methods.

To estimate the parameters $\boldsymbol{\beta}$, the likelihood for the ith unit is expressed as,

$$L_i(\boldsymbol{\beta}) = \left(\left[\frac{u'(t_i)}{\sigma u(t_i)}\right] \phi_{\text{LEV}}\left\{\frac{\log[u(t_i)] - \log(\alpha)}{\sigma}\right\}\right)^{\delta_i} \times \left(1 - \Phi_{\text{LEV}}\left\{\frac{\log[u(t_i)] - \log(\alpha)}{\sigma}\right\}\right)^{1-\delta_i}.$$

Note that this is not an actual likelihood, because the "response" $u(t_i)$ is unknown. Instead, the log-likelihood function serves as a loss function for the estimation purpose. The log-likelihood function for the ith unit is

$$\begin{aligned} l_i(\boldsymbol{\beta}) = \quad & \delta_i \left\{\log[u'(t_i)] - \log(\sigma) - \log[u(t_i)] - \log\left[\frac{\alpha}{u(t_i)}\right]^{1/\sigma} - \left[\frac{\alpha}{u(t_i)}\right]^{1/\sigma}\right\} \\ & + (1 - \delta_i) \log\left(1 - \exp\left\{-\left[\frac{\alpha}{u(t_i)}\right]^{1/\sigma}\right\}\right). \end{aligned} \qquad (3)$$

Hence, the overall log-likelihood function is $l(\boldsymbol{\beta}|\boldsymbol{H}) = \sum_{i=1}^{n} l_i(\boldsymbol{\beta})$. Then, we can obtain the maximum likelihood (ML) estimates of $\boldsymbol{\beta}$ by maximizing the overall log-likelihood function.

Note that $\boldsymbol{\beta}$ are the unknown parameters, and α and σ are set as constants because the "response" $u(t_i)$ is unknown. From Figure 1(b) and the perspective of the likelihood function, the smaller value of σ is, the larger value of the pdf of a failure is. Besides, inside the $\phi_{\text{LEV}}(\cdot)$ and $\Phi_{\text{LEV}}(\cdot)$, we have the $\log(u) - \log(\alpha) = \log(u/\alpha)$ term. In this case, α and $\boldsymbol{\beta}$ can take different combinations of values so that u/α keeps the same, which yields the same likelihood. Thus, it is necessary to let α and σ be given constants to avoid identifiability issues.

As discussed in Section 2.2 and from (3), it can be seen that $u(t_i) \approx \alpha$ if the ith unit is a failure, and $u(t_i) < \alpha$ if the ith unit is censored. That is, the role of α is to set the target failure threshold. Then, the role of σ is to measure how close the difference between $u(t_i)$ and α. So for the value of σ, ideally, we want σ to be a small enough value to allow the degradation index of failed units to end close to the target failure threshold α. For the value of α, in theory, we can set any value for the target failure threshold α. Then $\boldsymbol{\beta}$ can adjust correspondingly to provide a degradation index between 0 and α. In applications, setting α close to the mean of failure times helps the convergence of the estimation algorithm.

Although multiple sensors are available to assess the degradation process, not every sensor collected has a significant contribution. So we integrate variable selection in the model to find out informative sensors. Since there are multiple M-splines basis to represent one sensor, we want to penalize coefficient parameters associated with one sensor simultaneously when that sensor does not contribute. Therefore, we adopt the adaptive group LASSO method in [27] to conduct variable selection. The adaptive group LASSO approach penalizes parameters in the same group simultaneously. In our model, the parameters in M-splines for the same variable are treated to be in the same group. That is, $\boldsymbol{\beta}_i$ and $\boldsymbol{\beta}_j$ are in different groups for any $i \neq j$. From the perspective of variable selection, the jth sensor variable has no effect on $u(t)$ if all elements in $\boldsymbol{\beta}_j$ are significantly small. The adaptive group LASSO considers the penalties on different grouped parameters have different effects. Let ω_j be a given weight of the penalty for the jth variable, where $\omega_j \geq 0$ and $j = 1, \ldots, p$. Then, the penalized negative log-likelihood function is

$$\mathcal{L}(\boldsymbol{\beta};\lambda) = -l(\boldsymbol{\beta}|\boldsymbol{H}) + \lambda \sum_{j=1}^{p} \omega_j ||\boldsymbol{\beta}_j||_2, \tag{4}$$

where $\lambda \geq 0$ is a tuning parameter and $||\boldsymbol{\beta}_j||_2 = \sqrt{\sum_{k=1}^{m} \beta_{jk}^2}$ is the L_2 norm of the vector $\boldsymbol{\beta}_j$. Typically, the weights are given by setting

$$\omega_j = \begin{cases} ||\widetilde{\boldsymbol{\beta}}_j||_2^{-\gamma} & \text{if } ||\widetilde{\boldsymbol{\beta}}_j||_2 > 0 \\ \infty & \text{if } ||\widetilde{\boldsymbol{\beta}}_j||_2 = 0, \end{cases} \tag{5}$$

where $\widetilde{\boldsymbol{\beta}}_j$ is an estimate of $\boldsymbol{\beta}_j$ and $\gamma \geq 0$ is a hyper-parameter. Here we follow the practice in [27] and define $\infty \cdot 0 = 0$. That means the model does not select sensor j if its coefficient estimates L_2 norm is zero (i.e., $||\widetilde{\boldsymbol{\beta}}_j||_2 = 0$).

3.2 Optimization of Objective Function

The optimization of (4) is challenging. Here we discuss some strategies used in the optimization of the objective functions. To optimize the objective function, we use the Nelder-Mead algorithm in the R package *nloptr*. Due to the model complexity and non-convexity, there exist multiple local optimal of the coefficients. However, in our study, the main focus is to predict the status accurately. Therefore, even though there are local optimal points, it is of less concern as long as the optimization allows us to predict units' status accurately.

One thing to notice is the influence of σ value in the optimization procedure. If σ is prefixed at a small value (e.g., 0.01) at the beginning of the optimization, the algorithm could be easily trapped at local optima. As shown in Figure 1(b), when σ is small, the pdf of LEV is highly concentrated around the location parameter $\log(\alpha)$. That means a unit with $\log[u(t_i)]$ at the event time that is close to the threshold $\log(\alpha)$ has a high probability. While a unit which degradation index at the event time is far away from the threshold has almost zero probability, thus, its contribution to the likelihood function is small. During the optimization process, with a small σ, it is possible that the $\boldsymbol{\beta}$ is updated to an estimation that some units' degradation paths get almost 0 probability. So the contribution of these units to the objective function is neglected in the following updates of the $\boldsymbol{\beta}$ estimation. Only units with $\log[u(t_i)]$

that are relatively close to $\log(\alpha)$ have the chance to further move close to the threshold.

One approach to avoid the local optima is to set a larger value for σ at the beginning of the optimization process, and then decrease it gradually to the prefixed lower bound. By setting σ to a large value, say $\sigma = 1$, the information of all units is equally treated, regardless of the distance between the value of $\log[u(t_i)]$ and the location parameter. After a certain number of iterations, the unit's $\log[u(t_i)]$ moves closer to $\log(\alpha)$, then we can decrease the value of σ by a small amount and update $\boldsymbol{\beta}$ estimation. Repeating this step until σ decreased to the fixed constant can help to avoid the local optima problem. Instead of manually determining a sequence of σ to decrease, we add it to the optimization parameters.

Let σ_l be the prefixed lower bound of σ. We consider the transformation $\log(\sigma^*) = \log(\sigma - \sigma_l)$. Thus, $\sigma = \exp[\log(\sigma^*)] + \sigma_l$. Then we optimize $\log(\sigma^*)$ and $\boldsymbol{\beta}$ simultaneously. This transformation can always impose a lower bound for the estimation of σ. Although we include $\log(\sigma^*)$ in the parameter estimation, the purpose is not to obtain an estimation of $\log(\sigma^*)$. The reason is that σ is not identifiable and it always becomes smaller to allow a larger likelihood. Via the iterations, it will get to its lower bound eventually. Thus, we include σ in the parameter estimation so that it can smoothly decrease and help to avoid the local optima of $\boldsymbol{\beta}$ estimation.

With a larger value of σ in the early stage of the iterations, the benefit of the asymmetric property of the LEV distribution is not evident. We introduce the following remedy to ensure the estimates of $u(t_i)$ are moving towards the right direction during the early stage of the optimization iterations. As discussed above, when building a degradation index, it is desirable that $u(t_i) = \alpha$ if ith unit fails and $u(t_i) < \alpha$ if ith unit is censored. Thus, we further impose those two constraints on the objective function. That is, we modify the objective function as,

$$\mathcal{M}\left(\boldsymbol{\beta}, \lambda\right) = \mathcal{L}(\boldsymbol{\beta}; \lambda) + \eta \sum_{i=1}^{n} \delta_i \{[\alpha - u(t_i)]\}^2 + (1 - \delta_i)\{[u(t_i) - \alpha]_+\}^2. \tag{6}$$

The positive part function is $[u(t_i) - \alpha]_+ = \max\left(u(t_i) - \alpha, 0\right)$ and the penalty η is non-negative. Therefore, to encourage the estimated $u(t)$ satisfying the degradation index characteristic (i.e., $u(t_i) = \alpha$ if ith unit fails and $u(t_i) < \alpha$ if ith unit is censored) as well as perform variable selection, we work with the objective function $\mathcal{M}\left(\boldsymbol{\beta}, \lambda\right)$ as shown in (6).

3.3 Determining Tuning Parameter

In parameter estimation, we need to determine the tuning parameter λ in the objective function (6). The k-fold cross-validation (k-CV) approach is used. Because the main goal of our degradation index model is to accurately predict the status of testing units, especially for the failed units, we use both the false negative error rate and total error rate as the criterion to select the tuning parameters. Let FNR and FPR be the averaged false negative and positive error rates across the k folds, respectively. Then the averaged total error rate is $\mathrm{TER} = \mathrm{FNR} + \mathrm{FPR}$. For a sequence of values for λ, denoted by $\{\lambda_1, \ldots, \lambda_q\}$, the corresponding error rates are $\{\mathrm{TER}_1, \ldots, \mathrm{TER}_q\}$, and $\{\mathrm{FNR}_1, \ldots, \mathrm{FNR}_q\}$. Let $k_f = \arg\min_b \mathrm{FNR}_b$ be the index of tuning parameter that minimizes FNR. We want to select the tuning parameter λ so that FNR is minimized, while TER is kept at a relatively low level. That means we do not want to sacrifice FPR to achieve the smallest FNR. Therefore, the selected tuning parameter is λ_s, of which the index s is determined by

$$s = \begin{cases} \arg\min_b \mathrm{FNR}_b & \text{if } \mathrm{TER}_{k_f} \leq 0.2, \\ \arg\min_b \mathrm{TER}_b & \text{otherwise.} \end{cases} \tag{7}$$

In this way, when a λ minimizes FNR at the cost of FPR, we will switch to the λ that minimizes TER to achieve a balance between FPR and FNR.

The parameters η, γ, and σ_l are treated as hyper-parameters. We do not tune η because the role of the η penalty term is to help $u(t)$ move towards α at the beginning stage. After $u(t)$ is close to α, the effect of the asymmetric property of LEV kicks in and the shrinkage of σ towards σ_l serves the same role. Therefore, we only include η penalty term to help the algorithm converge and set a moderate large value for η. In particular, η is set to be 5. We fix the hyper-parameter $\gamma = 2$ in the calculation of the weights in (5), which is as a common practice.

The lower bound of scale parameter σ_l is set to be 0.01 in this paper. In practice, the σ_l can be set by users. Ideally, the smaller value of σ_l has better performance of the degradation index. However, the smaller value takes more computation time. In our cases, we set it as

$\sigma_l = 0.01$, because when $\alpha = \exp(5)$, the 99% tolerance interval of the LEV distribution will be $(145.9, 156.5)$, using the quantile of the LEV distribution. This interval is $(-2\%, 5\%)$ around the target $\exp(5)$, which is narrow enough. Regarding the selection of the number of splines, the strategy is setting it large enough to allow flexibility. Then we use the adaptive group LASSO penalty to regularize the estimation and let the penalty shrink those less important sensors.

3.4 Parameter Estimation Procedure

In this section, we describe how to obtain the estimates $\widehat{\boldsymbol{\beta}}$ based on the training set, using the adaptive group LASSO procedure. With initial estimates $\widetilde{\boldsymbol{\beta}}_j$, we obtain the weights ω_j, as in (5) for $j = 1, \cdots, p$. We first apply the k-CV in Section 3.3 to obtain the tuning parameter λ_s as in (7). The average of the $\boldsymbol{\beta}$ estimates from each fold in k-CV can also provide a warm starting point for the final estimate of $\boldsymbol{\beta}$. Similar ideas are also used in literature (e.g., [28]). With the best selected λ_s, the warm starting point for $\boldsymbol{\beta}$, and the entire training set, we apply the adaptive group LASSO procedure to obtain the final estimates of $\boldsymbol{\beta}$ in (6), which is denoted by $\widehat{\boldsymbol{\beta}}$. The statistical inference and variable selection results can be obtained based on $\widehat{\boldsymbol{\beta}}$.

The question remains how to find the initial estimates $\widetilde{\boldsymbol{\beta}}_j$'s. Huang et al. [27] suggested that one can use the group LASSO procedure (i.e., without adaptive weights) to find the initial estimates of parameters. For the group LASSO estimates, the objective function in (6) is simplified by setting $\omega_j = 1, j = 1, \cdots, p$. Similarly, we apply the k-CV to find the best tuning parameter $\widetilde{\lambda}_s$ for the group LASSO, and use the averaged estimates as the warm starting points for the final estimates of the group LASSO procedure, denoted by $\widetilde{\boldsymbol{\beta}}_j, j = 1, \cdots, p$.

For the initial values of the k-CV of the group LASSO procedure, we have to use cold starting points as we do not have much information about the parameters at this step. We randomly select the starting points that satisfy some desirable properties in the degradation scenario. The details of selecting starting points are given in Supplementary Section 1.

4 Simulation Study

In order to evaluate the performance of the proposed degradation index building method, we use various simulated scenarios to compare our model with the model without variable selection and the model that assumes linear sensor effect. We investigate the average times that our model properly selects variables and the average prediction accuracy (i.e., correctly predicts the status of a unit as a failure or censored one).

4.1 Simulation Setup and Procedure

In this simulation study, we want to generate datasets similar to the jet engine data to demonstrate the performance of our degradation index building framework. To generate the simulation data, we first consider the signal sensors. In the simulation study, we generate 10 sensor signals similar to the jet engine signals within time interval $[0, 350]$. We assume each signal is a function of time with some variations. That is $X_j(t) = g_j(t) + \epsilon_j(t), j = 1, \ldots, 10$. The function $g_j(t)$ can take forms such as constant, linear, quadratic, log functions, and the error term $\epsilon_j(t)$ follows a normal or a uniform distribution. Supplementary Figure 1 presents the example of simulated signals for 10 units. The signals can be increasing, decreasing, or randomly fluctuate over time. After obtaining the sensor information, the basis functions of the M-spline with 2 interior knots are constructed based on the signals. So we have 50 coefficient parameters for the simulated data. To test the variable selection capability of our method, we assume that 5 out of 10 signals cause the units to fail, and the rest 5 have no effect on the failure process. Hence, we set the values of parameters of the first 5 signals to have effects on the degradation index. Among the five signals with effect, we assume the second and fourth are linear functions of time, the first and the fifth are quadratic functions of time, and the third one is assumed to follow a normal distribution. With the true parameter coefficients $\boldsymbol{\beta}$ and the simulated sensor signal history, we can compute $u(t)$ using (1).

The next step is to generate the failure-time data. Let the time to failure be $T = \min\{C, 350\}$, where C follows a Weibull distribution. The Weibull shape and scale parameters are determined to ensure the proportion of failed units does not exceed 90%. Here we set the upper bound for T to be 350 to mimic the jet engine data. We set the failure threshold as $\alpha = \exp(5)$, and the censoring indicator is defined as $\delta = 1$ if $u(T) \geq \exp(5)$ and $\delta = 0$ if $u(T) < \exp(5)$.

To test the procedure under different situations, we consider various n and $\boldsymbol{\beta}$ to control the number of total units and effects degree of covariates. The number of units is chosen as $n = 50, 100, 150, 200, 250, 300$. Because we have 5 splines for each covariate, the corresponding $\boldsymbol{\beta}_j$ is of length 5 for $j = 1, \ldots, 10$. Assume that the last 5 signals do not affect the degradation process, so $\boldsymbol{\beta} = (\boldsymbol{\beta}_1', \boldsymbol{\beta}_2', \boldsymbol{\beta}_3', \boldsymbol{\beta}_4', \boldsymbol{\beta}_5', \mathbf{0}_{25}')'$. We consider four scenarios and the coefficients are listed in Supplementary Table 1, which are (A): Contributions of effective covariates $(x_1(t), \ldots, x_5(t))$ are on the same magnitude; (B): The signals with quadratic function forms (i.e., $x_1(t)$ and $x_5(t)$) have larger effects; (C): The signals with linear function forms (i.e., $x_2(t)$ and $x_4(t)$) have larger effects; and (D): Only the random term (i.e., $x_3(t)$) has larger effect.

4.2 Method Comparisons

In order to better understand the performance of the proposed model and procedure, we compare the proposed model with the model without variable selection, and with the model considering only linear relationship for the covariate effect.

For the model without variable selection, λ is set to 0 in (6). The objective function is,

$$\mathcal{M}(\boldsymbol{\beta}) = -l(\boldsymbol{\beta}|\boldsymbol{H}) + \eta \sum_{i=1}^{n} \delta_i \{[\alpha - u(t_i)]\}^2 + (1 - \delta_i)\{[u(t_i) - \alpha]_+\}^2.$$

If we assume the covariate effect is in a linear form, then degradation index becomes $\widetilde{u}(t) = \int_0^t h \left\{ \sum_{j=1}^{p} x_j(s) \beta_j \right\} ds$. We also use adaptive LASSO to perform variable selection and the objective function is the same as (6) but with $u(t)$ replaced by $\widetilde{u}(t)$.

4.3 Simulation Results

We generate simulation data based on the procedure described in Section 4.1 and apply three different models to the simulated data. We denote our proposed model as DI-VS, the model without variable selection as DI-NVS, and the model assumes linear sensor effect as DI-VSL.

In this simulation, the target failure threshold is set to be $\alpha = \exp(5)$. For each sample size n and coefficient parameter vector $\boldsymbol{\beta}$, we repeat the trial 200 times. For each trial, we split the simulated data into 80% training set and 20% testing set. The tuning parameter is selected by 5-fold cross-validation on the training set.

Regrading to the predictions of unit status, although the target failure threshold is α, in practice, due to training errors. If we use the target failure threshold α as the threshold, some failures with $u(t) < \alpha$ are reported to be surviving units. In practice, one can use a threshold that is slightly smaller than α, which typically yields better classification results. We call this the *practical threshold* $\widetilde{\alpha}$. One can choose $\widetilde{\alpha}_p = \exp\left[\log(\alpha) + z_p\sigma\right]$, where z_p is the quantile function of the standard LEV distribution. In the simulation study, the classification uses the $\widetilde{\alpha}_{0.01}$ threshold. The practical threshold separates the two categories better than the target failure threshold. For the purpose of prediction, we suggest using the practical threshold for predicting the status of testing units.

Figure 2 shows the average FNR, FPR, and TER as the number of units increases. Although in general FNR decreases for all three models, DI-VS has the most consistent performance across various scenarios and the number of units. When the number of units is small (i.e., less than 100), the FNR of DI-NVS is the largest among the three models, which shows the benefit to consider variable selection, especially when n is relatively small. The FNR for DI-VSL is large in Scenarios A and D, even when the number of units is large. For FPR, in Scenarios B, C, and D, FPR does not change a lot with the size increases. In Scenario A, DI-NVS has the smallest FPR across the number of units while DI-VSL has the largest. The results show that ignoring nonlinear relationships can lead to larger errors in some scenarios. With respect to the TER, DI-VS has the smallest errors for almost all scenarios.

Regrading the variable selection capability, Figure 3 shows the average number of correctly specified variables, effect variables, and no effect variables identified by the model versus the number of units (n) for different methods and scenarios. The actual number of effective variables is 5, no-effect variables in the model is 0, and correctly specified variables are 10. Ideally, we want the model to include all 5 effective variables and zero no-effect variable. The number of correctly specified variables includes the number of effective variables that

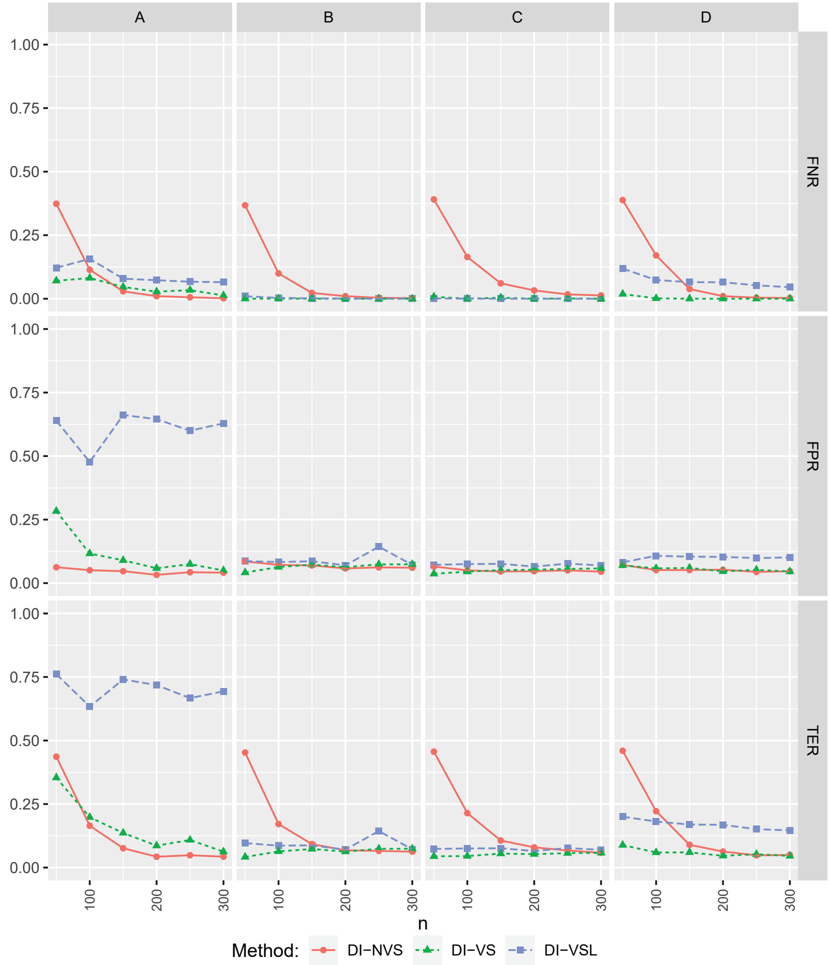


Figure 2: Average FNR, FPR and TER versus number of units (n) for different methods and scenarios.

remained in the model and no-effect variables excluded from the model, which should be 10. Compared to DI-VSL, DI-VS tends to include more effective signals in the model when the number of units is large in Scenarios A and D. For Scenarios B and C, the number of effect signals remained in the model is close to DI-VSL when the number of units is large. For signals that have no effect on the underlying degradation process, DI-VS can exclude more signals across all four scenarios and various numbers of units. In general, the DI-VS model has more correctly specified variables across different simulation scenarios compared to the DI-VSL model.

Overall, the simulation results show that our proposed model has better accuracy in predicting a unit status for various scenarios and numbers of units. It can also exclude signals with no effect from the model. Supplementary Section 3 also contains some further simulation results which show that the proposed method can select the most informative sensors and is robust to different censoring rates.

5 Application

5.1 The Illustrative Application

In this section, we apply the developed methods to the jet engine sensor data ([2]). We first provide a brief introduction to the jet engine data. For illustration, we use a subset that contains 200 units with 100 failures and 100 surviving units. The failure-time data give the cycles to failure for failed units and time in service for surviving units. The multi-channel sensor data give time-varying signals with cycles for all 200 units. The dataset includes 21 sensor outputs that measure the system's physical and functional conditions. There are 8 sensors that record temperature and pressure at the fan inlet and different outlets and 8 sensors that capture various fan speeds and coolant bleed in the simulation model. Besides, the measurements of pressure ratio, fuel flow ratio, bypass ratio, burner fuel-air ratio, and bleed enthalpy are also provided in the dataset. In the jet engine simulation data, there are 16 multi-channel sensors after removing those with constant signals. Figure 5 illustrates the cycles to failure for 20 units with their corresponding 16 multi-channel signals.

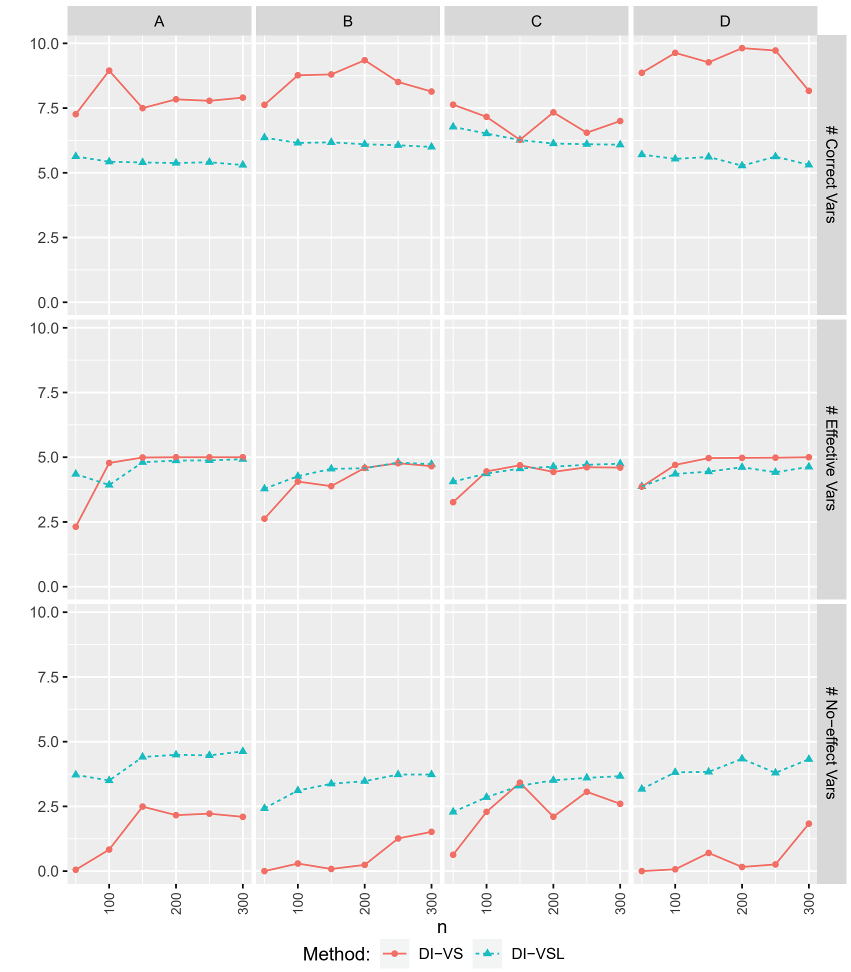


Figure 3: Average number of correctly specified variables, effect variables, and no effect variables identified by the model versus number of units (n) for different methods and scenarios.

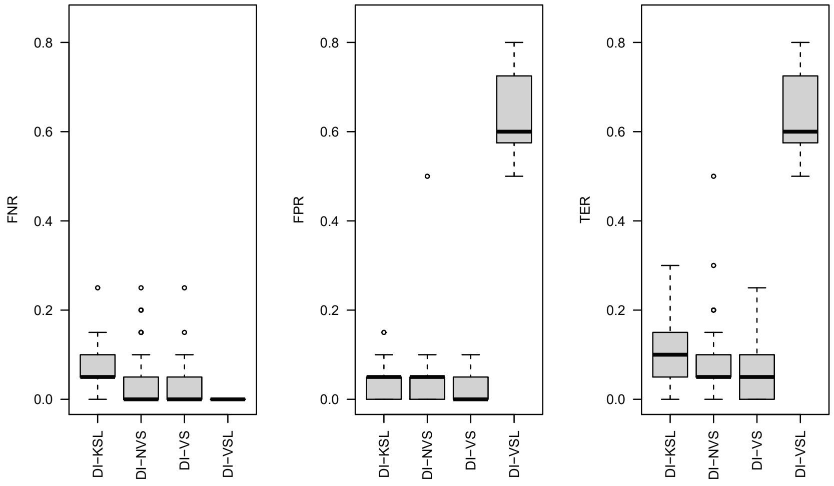
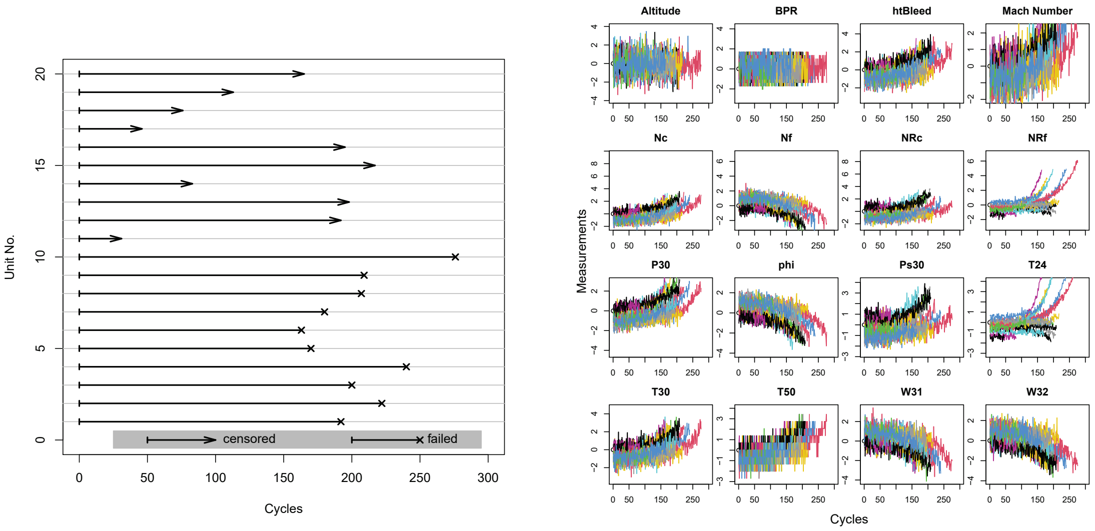


Figure 4: The boxplot of error rates with jet engine data with 50 replicates for different methods with $\widetilde{\alpha}_{0.01}$ practical threshold.



(a) Time to Event

(b) Multi-channel Signals

Figure 5: Plot of a subset of the time to event (a), and multi-channel signals of the 20 units presented by different colors (b).

Except for the model introduced in Section 4.2, in the real application, we also compare our model with existing models in literature [1], which is denoted as DI-KSL. The details of this method are introduced in Supplementary Section 4.1. The prediction ability of each model are investigated and the results are summarized in Section 5.2.

5.2 Result Comparisons

For the model comparison, we randomly split the dataset into 80% training set and 20% testing set while keeping the proportion of failed units as 50% in both training and testing sets. We apply each of the four models DI-VS, DI-NVS, DI-VSL, and DI-KSL to the training set and then use the trained model to predict unit status in the test set. In DI-VS and DI-NVS model, we use 10 spline basis to represent each of the 16 sensors, so in total, we have 160 coefficient parameters. The target failure threshold is set as $\alpha = \exp(5)$. The tuning parameter is selected by 5-fold cross-validation. When the sample size is relatively small (e.g., 200 in this application), it is generally not recommended to go with a large number of folds. Thus, five was a reasonable choice. For DI-KSL, we only use the failed units in the training set to build the model due to the model property. For each method, we repeat the splitting

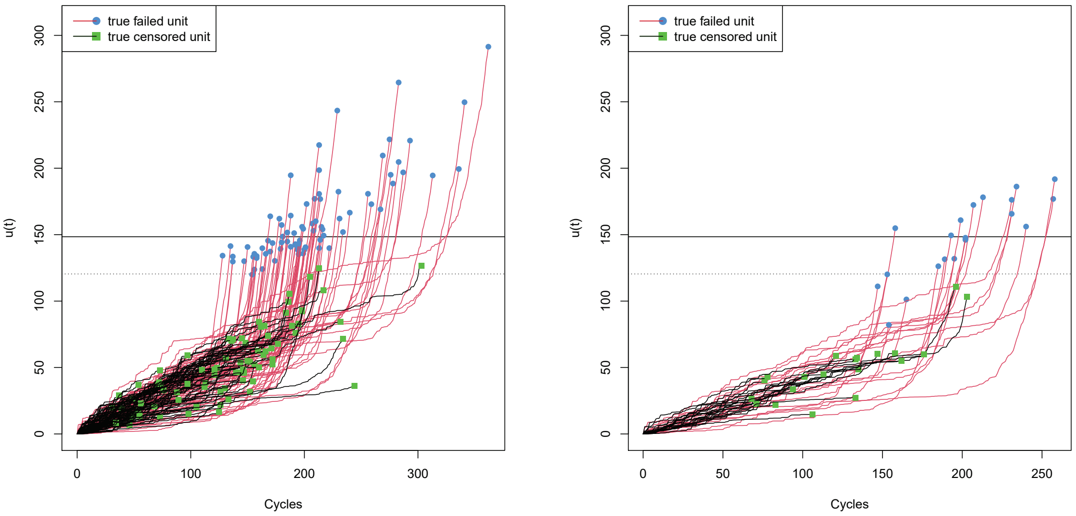
50 times and the results are summarized as follows.

Figure 4 presents the boxplot of the prediction errors over 50 splits on the testing set of four models and Table 1 provides the average prediction error over repetitions. We can see that DI-VS has the lowest averaged total error and FPR among the four models. The DI-VSL has zero FNR. However, its FPR is unusually large. One potential reason is that DI-VSL fails to capture the nonlinear trend in the data. This can cause the estimation of $\log(\sigma^*)$ to fail to shrink as expected, which results in the model having errors on one side. The DI-KSL model has the second smallest average FPR. However, its FNR is 4% larger than our proposed model DI-VS. One possible reason is that the DI-KSL model neglects the censored unit's information when training the model. Except for the DI-VSL model, the other three models have a similar inter quantile range for prediction errors.

Substituting the estimates into (1), the degradation index over time can be obtained for an individual unit. Figure 6 presents the degradation index built with our proposed framework DI-VS from one split. In this plot, in the testing set, all censored units are below $\widetilde{\alpha}_{0.01}$ threshold and most failed units are over that threshold. We can see from the plot that using a practice failure threshold allows more true failed units' $u(t)$ to reach the threshold.

Regrading to the variable selection, in the above jet engine data analysis, all 16 sensors are kept by the DI-VS model. However, as we have already seen in the simulation study, when the number of units in the model becomes larger, DI-VS tends to keep more variables in the model even some of them are no-effect. Therefore, we take a small subset of the jet engine data and test the model's variable selection ability when n is small. Supplementary Figure 6(a) shows the proportion of each variable being excluded over 20 replicates when the number of units changes from 40 to 80 with 10 increment. We can see that when the number of units is relatively small, some sensors such as NRf, altitude, and Nf are excluded from the model with high probability. When $n = 40$, each variable is excluded from the model at least once. However, as n increases, there are variables that can be remained in the model over all 20 replicates.

To better understand the sensors' effects on the degradation path, we use the accumulated local effects (ALE) plot in [29] for visualization. ALE plot is a visualization approach to present



(a) The $u(t)$ on Training Set

(b) The $u(t)$ on Testing Set

Figure 6: The degradation index for training and testing set with one split of the jet engine data. The horizontal lines represent different thresholds. The solid line is $\alpha = \exp(5)$ and the dotted line is $\widetilde{\alpha}_{0.01}$.

Table 1: The average FNR, FPR, and TER for the DI-VS, DI-KSL, DI-VSL, DI-NVS models based on 50 data splits. Here the practical threshold is $\widetilde{\alpha}_{0.01}$.

Model	FNR	FPR	TER
DI-VS	0.030	0.026	0.057
DI-KSL	0.070	0.034	0.104
DI-VSL	0.000	0.645	0.645
DI-NVS	0.040	0.040	0.080

the predictors' main and secondary-order effects in complex black box supervised learning models. We provide the technical details and the ALE plots in Supplementary Section 4.2. The ALE plot in Supplementary Figure 6(b) shows that the temperature at LPT outlet (T50) and pressure at HPC outlet tend to have a constant influence when the measurements are low and a larger effect on the damage level when measurements increase, while the coolant bleed (W31) has decreasing effect before a certain point then the effect becomes constant.

6 Conclusions and Areas for Future Research

In this paper, motivated by the jet engine multi-channel sensory data, we propose a new framework to build the degradation index based on the cumulative exposure model. The framework can handle censored data and conduct variable selection automatically. The comprehensive simulation studies and jet engine data analysis show that our approach has flexibility and advantages. It is also demonstrated that the performance of the proposed framework is more robust than other models. The DI-VS model has consistently good prediction accuracy regardless of the dataset size and scenarios.

The flexibility of the proposed framework makes it applicable to many complex engineering systems equipped with multiple sensors. In practice, based on the knowledge about different applications, the way to model the impact of sensors $f_j(x_j)$, and how to conduct the nonlinear transformation $h(z)$ can be adjusted accordingly. Regarding the usage of the proposed degradation index, this paper focuses on the product status prediction. Other potential usage includes the product health assessment and prediction for the remaining useful life.

There are a few limitations of the proposed framework. One of the limitations is that the model tends to keep no-effect sensors when the number of units n is large. A possible reason is that we use the log-likelihood function in the objective function. When the number of units is large, as long as the sensor can provide a little contribution to the likelihood, adding up these contributions of n units can lead to large minimization of the objective function. It will be interesting to study the adaptive elastic net penalties or involve the number of units in the penalty term. Another limitation is that the proposed model can only handle numerical covariates. It will be useful to extend the current framework to deal with categorical variables.

There are some other future directions for the proposed model. In this paper, we consider an additive way to model all sensors' impact. However, in reality, the way that sensors influence the degradation process can be much more complex. It will be interesting to allow sensors interactions in the model. Besides, this proposed framework requires strict monotonicity of the degradation index. If the Wiener process is used, then there needs not to be such a strict requirement for monotonicity as long as there is an overall trend. It will be interesting to study degradation index building with less requirement on monotonicity. Multiple degradation characteristics modeling has been popular (e.g., [30, 31]). Building multiple degradation indexes for data with multiple degradation characteristics is also worth investigating.

Supplementary Material

The following supplementary materials are available online.

Additional details: Additional details for the algorithm, simulation study, and data analysis (pdf file).

Acknowledgment

The authors thank the editor, associate editor, and referees, for their valuable comments that helped improve the paper significantly. The authors acknowledge the Advanced Research Computing program at Virginia Tech for providing computational resources. The work by Hong was partially supported by National Science Foundation Grant CMMI-1904165 to Virginia Tech.

References

[1] M. Kim, C. Song, and K. Liu, "A generic health index approach for multisensor degradation modeling and sensor selection," *IEEE Transactions on Automation Science and Engineering*, vol. 16, pp. 1426–1437, 2019.

[2] A. Saxena and K. Goebel, "PHM08 challenge data set," tech. rep., NASA Ames Prognostics Data Repository, Moffett Field, CA, 2008.

[3] W. Q. Meeker and L. A. Escobar, *Statistical Methods for Reliability Data.* John Wiley & Sons, 1998.

[4] Z. Ye and N. Chen, "The inverse Gaussian process as a degradation model," *Technometrics*, vol. 56, pp. 302–311, 2014.

[5] Z. Ye and M. Xie, "Stochastic modeling and analysis of degradation for highly reliable products," *Applied Stochastic Models in Business and Industry*, vol. 31, pp. 16–32, 2015.

[6] W. Q. Meeker and Y. Hong, “Reliability meets big data: Opportunities and challenges, with discussion,” *Quality Engineering*, vol. 26, pp. 102–116, 2014.

[7] Y. Hong, M. Zhang, and W. Q. Meeker, “Big data and reliability applications: The complexity dimension,” *Journal of Quality Technology*, vol. 50, pp. 135–149, 2018.

[8] H. Zheng, X. Kong, H. Xu, and J. Yang, “Reliability analysis of products based on proportional hazard model with degradation trend and environmental factor,” *Reliability Engineering & System Safety*, vol. 216, p. 107964, 2021.

[9] Z. Wang, Q. Zhai, and P. Chen, “Degradation modeling considering unit-to-unit heterogeneity-a general model and comparative study,” *Reliability Engineering & System Safety*, vol. 216, p. 107897, 2021.

[10] W.-B. Chen, X.-Y. Li, and R. Kang, “Integration for degradation analysis with multi-source adt datasets considering dataset discrepancies and epistemic uncertainties,” *Reliability Engineering & System Safety*, vol. 222, p. 108430, 2022.

[11] C. Duan, Y. Li, H. Pu, and J. Luo, “Adaptive monitoring scheme of stochastically failing systems under hidden degradation processes,” *Reliability Engineering & System Safety*, vol. 222, p. 108322, 2022.

[12] A. Kumar, C. Parkash, G. Vashishtha, H. Tang, P. Kundu, and J. Xiang, “State-space modeling and novel entropy-based health indicator for dynamic degradation monitoring of rolling element bearing,” *Reliability Engineering & System Safety*, vol. 222, p. 108356, 2022.

[13] H. Wang, H. Liao, and X. Ma, “Stochastic multi-phase modeling and health assessment for systems based on degradation branching processes,” *Reliability Engineering & System Safety*, vol. 222, p. 108412, 2022.

[14] K. Liu, N. Z. Gebraeel, and J. Shi, “A data-level fusion model for developing composite health indices for degradation modeling and prognostic analysis,” *IEEE Transactions on Automation Science and Engineering*, vol. 10, pp. 652–664, 2013.

[15] X. Fang, K. Paynabar, and N. Gebraeel, “Multistream sensor fusion-based prognostics model for systems with single failure modes,” *Reliability Engineering & System Safety*, vol. 159, pp. 322–331, 2017.

[16] C. Song, K. Liu, and X. Zhang, “Integration of data-level fusion model and kernel methods for degradation modeling and prognostic analysis,” *IEEE Transactions on Reliability*, vol. 67, pp. 640–650, 2018.

[17] A. Chehade, C. Song, K. Liu, A. Saxena, and X. Zhang, “A data-level fusion approach for degradation modeling and prognostic analysis under multiple failure modes,” *Journal of Quality Technology*, vol. 50, pp. 150–165, 2018.

[18] C. Song and K. Liu, “Statistical degradation modeling and prognostics of multiple sensor signals via data fusion: A composite health index approach,” *IISE Transactions*, vol. 50, pp. 853–867, 2018.

[19] Y. Wei, D. Wu, and J. Terpenny, “Learning the health index of complex systems using dynamic conditional variational autoencoders,” *Reliability Engineering & System Safety*, vol. 216, p. 108004, 2021.

[20] M. Kim, J.-R. C. Cheng, and K. Liu, “An adaptive sensor selection framework for multisensor prognostics,” *Journal of Quality Technology*, vol. 53, pp. 566–585, 2021.

[21] T. Li, X. Si, H. Pei, and L. Sun, “Data-model interactive prognosis for multi-sensor monitored stochastic degrading devices,” *Mechanical Systems and Signal Processing*, vol. 167, p. 108526, 2022.

[22] R. Tibshirani, “Regression shrinkage and selection via the Lasso,” *Journal of the Royal Statistical Society, Series B*, vol. 58, pp. 267–288, 1996.

[23] H. Zou, “The adaptive Lasso and its oracle properties,” *Journal of the American Statistical Association*, vol. 101, pp. 1418–1429, 2006.

[24] M. Yuan and Y. Lin, “Model selection and estimation in regression with grouped variables,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 68, pp. 49–67, 2006.

[25] Y. Hong and W. Q. Meeker, “Field-failure predictions based on failure-time data with dynamic covariate information,” *Technometrics*, vol. 55, pp. 135–149, 2013.

[26] J. O. Ramsay, “Monotone regression splines in action,” *Statistical Science*, vol. 3, pp. 425–441, 1988.

[27] J. Huang, J. L. Horowitz, and F. Wei, “Variable selection in nonparametric additive models,” *Annals of Statistics*, vol. 38, pp. 2282–2313, 2010.

[28] R. Mazumder, J. H. Friedman, and T. Hastie, “Sparsenet: Coordinate descent with nonconvex penalties,” *Journal of the American Statistical Association*, vol. 106, pp. 1125–1138, 2011.

[29] D. W. Apley and J. Zhu, “Visualizing the effects of predictor variables in black box supervised learning models,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 82, pp. 1059–1086, 2020.

[30] G. Fang, R. Pan, and Y. Hong, “Copula-based reliability analysis of degrading systems with dependent failures,” *Reliability Engineering & System Safety*, vol. 193, p. 106618, 2020.

[31] Z. Saberzadeh and M. Razmkhah, “Reliability of degrading complex systems with two dependent components per element,” *Reliability Engineering & System Safety*, vol. 222, p. 108398, 2022.