Deep Bayesian survival analysis of rail useful lifetime

Zeng, Cheng; Huang, Jinsong; Wang, Hongrui; Xie, Jiawei; Zhang, Yuting

DOI
10.1016/j.engstruct.2023.116822

Publication date
2023

Document Version
Final published version

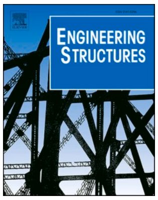
Published in
Engineering Structures

Citation (APA)
Zeng, C., Huang, J., Wang, H., Xie, J., & Zhang, Y. (2023). Deep Bayesian survival analysis of rail useful lifetime. *Engineering Structures*, *295*, Article 116822. https://doi.org/10.1016/j.engstruct.2023.116822

Contents lists available at ScienceDirect

Engineering Structures

journal homepage: www.elsevier.com/locate/engstruct



Deep Bayesian survival analysis of rail useful lifetime

Cheng Zeng [a], Jinsong Huang [a,*], Hongrui Wang [b], Jiawei Xie [a], Yuting Zhang [a]

[a] *Discipline of Civil, Surveying and Environmental Engineering, The University of Newcastle, Callaghan, NSW 2308, Australia*
[b] *Section of Railway Engineering, Delft University of Technology, Delft 2628CN, the Netherlands*

ARTICLE INFO

Keywords:
Rail useful lifetime
Survival analysis
Deep neural networks
Bayesian inference

ABSTRACT

Reliable estimation of rail useful lifetime can provide valuable information for predictive maintenance in railway systems. However, in most cases, lifetime data is incomplete because not all pieces of rail experience failure by the end of the study horizon, a problem known as censoring. Ignoring or otherwise mistreating the censored cases might lead to false conclusions. Survival approach is particularly designed to handle censored data for analysing the expected duration of time until one event occurs, which is rail failure in this paper. This paper proposes a deep Bayesian survival approach named BNN-Surv to properly handle censored data for rail useful lifetime modelling. The proposed BNN-Surv model applies the deep neural network in the survival approach to capture the non-linear relationship between covariates and rail useful lifetime. To consider and quantify uncertainty in the model, Monte Carlo dropout, regarded as the approximate Bayesian inference, is incorporated into the deep neural network to provide the confidence interval of the estimated lifetime. The proposed approach is implemented on a four-year dataset including track geometry monitoring data, track characteristics data, various types of defect data, and maintenance and replacement (M&R) data collected from a section of railway tracks in Australia. Through extensive evaluation, including Concordance index (C-index) and root mean square error (RMSE) for evaluating model performance, as well as a proposed CW-index for evaluating uncertainty estimations, the effectiveness of the proposed approach is confirmed. The results show that, compared with other commonly used models, the proposed approach can achieve the best concordance index (C-index) of 0.80, and the estimated rail useful lifetimes are closer to real lifetimes. In addition, the proposed approach can provide the confidence interval of the estimated lifetime, with a correct coverage of 81% of the actual lifetime when the confidence interval is 1.38, which is more useful than point estimates in decision-making and maintenance planning of railroad systems.

1. Introduction

All railway tracks would experience a certain degree of degradation over time. This degradation is particularly critical in Australia because numerous railway tracks carry heavy haul trains with axle loads up to 40 t. After a certain period, fatigue and other failure mechanisms may cause rail failures and ultimately end the useful lifetime of the rail, resulting in high costs and intensive maintenance, and even derailment. Therefore, early estimating of the rail useful lifetime is important to plan maintenance, optimize costs, and proactively prevent rail failures.

Previous research on rail degradation and rail useful lifetime estimation can be grouped into two categories: 1) large region-based models for statistical degradation determination. 2) segment-based models for predicting the location and time that rail failures are likely to occur.

Orringer [1] employed a stochastic process to develop a deterioration model that describes the probability of rail having defects at a particular period over a large region. Similar work was done by Zhao et al. [2], in which a combined probabilistic-based model was proposed to analyse the risk of derailment at a particular time. In addition, Jeong and Gordon [3] constructed a risk assessment model to forecast the occurrence of rail breaks between two consecutive inspections. A fuzzy logic model was developed by Vesković et al. [4] for predicting the frequency of rail break occurrence on some large sections of railway tracks. All previously mentioned models are large region-based models. However, rail degradation varies in degree at different locations. A model that estimates rail useful lifetime for large regions may lose generality and provide estimations with undesirable errors for rails at a specific location, which makes targeted predictive maintenance difficult.

Recent research interests are towards building segment-based

[*] Corresponding author.
E-mail address: jinsong.huang@newcastle.edu.au (J. Huang).

https://doi.org/10.1016/j.engstruct.2023.116822
Received 28 March 2023; Received in revised form 29 June 2023; Accepted 25 August 2023
Available online 8 September 2023

models to predict where and when rail failures will occur. For example, Dick et al. [5] proposed a multivariate statistical model to predict locations where rail failures were most likely to occur in two years. Further, by utilizing machine learning techniques, Schafer and Barkan [6] proposed a neural network-based model to improve the performance for predicting the locations of potential rail failure based on the same dataset used by Dick et al. [5]. Recently, Zhang et al. [7] used a tree-based machine learning technique to estimate the risk of rail failures at a certain location. Ghofrani et al. [8] applied an ensemble-based machine learning model to analyse the risk of rail failures. As one can see, very few studies have explored the estimation of rail useful lifetime. The most related work was done by Bai et al. [9]. They built a predictive model based on Markov stochastic processes to estimate the rail useful lifetime. The lifetime data used in the modelling were derived from severely defective rails and broken rails events. However, in most of the rail segments, rail failures are not always observed. For example, some rail segments did not experience a failure before the end of the study horizon, but they could potentially fail at a future date. Such cases are known as right censored. Ignoring or otherwise mistreating the censored cases might lead to false conclusions [10].

Survival approach is particularly designed to handle censored data for analysing the length of time until one event of interest occurs, such as a patient's death or mechanical system failure. In survival approach, the time to event is characterized by the survival model, which represents the probability that an individual is still alive at a certain time, or in this paper, a rail segment is still working safely. There are some popular survival models such as Cox proportional hazards (Cox) model, log-logistic model, and Weibull accelerated failure time (Weibull) model. All models attempt to represent the hazard rate (the probability of failure in a very small time interval) as a function of lifetime and observed data, which are also statistically called covariates. Ghofrani et al. [11] used a Weibull model to forecast the risk of service failures in railway tracks. Chi et al. [12] used Cox model, Weibull model, and log-logistic model to analyse the time to failure of the high-speed railway train wheelsets. Alemazkoor et al. [13] developed a mixed-survival model to estimate wheel wear rates. Extensive research has demonstrated the potential of survival approach in lifetime estimation for vehicle components in railway systems. However, no previous studies have exploited the potential of survival approach for rail useful lifetime modelling. This lack of exploration is due to several reasons, all of which pose major challenges.

In classical survival models, it is assumed that the hazard rate is linearly related to the covariates. However, in many applications, this assumption might be too simplistic. Therefore, a more complex family of survival models is necessary to properly capture nonlinear relationships in covariates. The fast developments in artificial intelligence have enabled researchers to utilize deep neural networks to build models that offer improved accuracy and flexibility in modelling the relationship between targeted tasks and covariates [14]. For example, Sresakoolchai and Kaewunruen [15], Sresakoolchai and Kaewunruen [16], and Sresakoolchai et al. [17] used deep learning techniques to estimate the condition of critical components of railway structure and vehicle, achieving good prediction performance. In the case of survival approach, deep neural networks are also deployed lately. Katzman et al. [18] used neural networks to model the non-linear relationship between covariates and the risk of a clinical event in the framework of survival approach. Lee et al. [19] used neural networks to directly learn the distribution of patients' survival times. Additionally, Giunchiglia et al. [20] proposed a parametric survival model that employed recurrent neural networks for medical practice.

The deep neural network-based survival models have gained undisputed success, especially the one developed by Katzman et al. [18], which has shown its strength in many applications with good performance [21,22]. However, these deep neural network-based survival models provide only point estimates of the hazard rates and thus cannot properly convey uncertainty in the estimations. Overly confident estimations might lead to unreliable decisions and potentially severe consequences, particularly in safety–critical industries like railway transportation, which involves substantial risks to both economy and personal safety. As such, it is necessary to properly consider the uncertainties in deep neural network-based survival models. Bayesian deep learning provides an appropriate way for measuring uncertainty [23,24]. In 2016, Gal and Ghahramani [25] proposed a practical Bayesian deep learning method named Monte Carlo dropout, which is a stochastic regularization technique. In this method, the uncertainty in deep neural networks can be estimated and confidence interval of estimations can be provided [26,27]. This method has been successfully applied in various fields such as image segmentation [28], object detection [29,30], and active learning [31].

This paper proposes a deep Bayesian survival approach, named BNN-Surv, to properly handle censored data for rail useful lifetime modelling. To capture the non-linear relationship between covariates and rail useful lifetime, a multi-layer neural network is used to represent the hazard rate in survival model. To consider and quantify uncertainty, Monte Carlo dropout, regarded as the approximate Bayesian inference, is incorporated into the deep neural network-based survival model to provide the confidence interval of the hazard rate as well as estimated rail useful lifetime. The proposed approach is demonstrated on a section of railroads in Australia. Track geometry monitoring data, track characteristics data, various types of defects data, and M&R data are used for model development. Through extensive evaluation, including C-index and RMSE for evaluating model performance, as well as a proposed CW-index for evaluating uncertainty estimations, the effectiveness of the proposed approach is confirmed. The key contributions of this study are summarized as follows:

1) To the best of our knowledge, this is the first effort to use deep neural network combined with Monte Carlo dropout as a survival approach. It is verified that, compared to the classical survival approach, integrating deep neural network into the survival approach can achieve better performance and the estimated rail useful lifetimes are closer to the real values. In addition, incorporating Monte Carlo dropout can provide the confidence interval of the estimated rail useful lifetime.
2) For the first time, the survival approach is used for rail useful lifetime modelling. This allows the censored data collected by the railroads can be properly considered in the model development.
3) The proposed model in this study can serve as a valuable tool for rail useful lifetime modelling, which helps railroads to make informed decisions and optimize predictive maintenance.

2. Methodology

2.1. Problem statement

Given a set of covariates $\mathbf{x}$ and rail useful lifetimes T and labels E, this study aims to model the length of time until rail failure occurs. Specifically, the covariate is defined as a matrix of $n \times m$, n being the number of observations and m being the number of covariates, such as representations of track conditions and track characteristics. Rail useful lifetime T denotes the time interval that the rail maintains its normal condition until failure. Label E represents whether the rail useful lifetime is observed. If a failure has happened, the rail useful lifetime T is fully known, and the label is $E = 1$. If a failure has not happened, the rail useful lifetime T is partially observed, i.e., only known that the rail does not fail until the end of the study horizon. In such a case, the data is labelled as $E = 0$ and is called censored data. The standard regression methods consider the censored data as a type of missing data and usually discard them, which may introduce bias in the model. As such, to adequately handle censored data, the use of the survival approach is crucial.

2.2. Survival approach

Survival approach [32] aims to model the distribution of rail useful lifetime T with censored data consideration, where T can be described by the probability density function $f(t)$ and cumulative distribution function $F(t)$. The probability that rail failure occurs before a certain time t, can be written as

$$F(t) = \mathbb{P}(T \leq t) = \int_0^t f(\tau)d\tau \tag{1}$$

The opposite case, i.e., the probability that a rail failure does not occur at t is called survival probability $S(t)$

$$S(t) = \mathbb{P}(T > t) = S(t) = 1 - F(t) \tag{2}$$

The survival probability in the survival approach is time-dependent and commonly described using the hazard rate $h(t)$. The hazard rate $h(t)$ represents the probability that a rail failure will occur in a very small-time interval, provided that the failure has not occurred before that particular time interval.

One of the most commonly used survival models is the Cox model [33], which offers a semi-parametric description of the hazard rate in continuous time. This model is based on the proportional hazards assumption, where the ratio of hazard rates between two observations is constant and only depends on the covariate values $\mathbf{x}$, as can be seen from Eq. (3).

$$h(t|\mathbf{x}_i) = h_0(t)\exp(g(\mathbf{x}_i)), g(\mathbf{x}_i) = (\boldsymbol{\beta}^{\mathrm{T}}\mathbf{x}_i) \tag{3}$$

In Eq. (3), the hazard rate is divided into two components $h_0(t)$ and $g(\mathbf{x})$. The non-parametric baseline hazard, represented by $h_0(t)$, varies over time and is consistent across all observations. Meanwhile, the partial hazard, denoted by $g(\mathbf{x})$, is time-invariant and varies with covariate values. The partial hazard is expressed as a linear function of the covariates, in which vector $\boldsymbol{\beta}$ represents the coefficients for the observation. The non-parametric baseline hazard is typically modelled with the Breslow estimator, according to Lin [34]. The parametric component $g(\mathbf{x})$ is determined by maximizing the Cox partial likelihood L as follow:

$$L = \prod_{i:E_i=1} \frac{h_0(t_i)\exp[g(\mathbf{x}_i)]}{\sum_{j\in\Re(t_i)} h_0(t_i)\exp\left[g\left(\mathbf{x}_j\right)\right]} = \prod_{i:E_i=1} \frac{\exp[g(\mathbf{x}_i)]}{\sum_{j\in\Re(t_i)} \exp\left[g\left(\mathbf{x}_j\right)\right]} \tag{4}$$

where t_i, E_i, and $\mathbf{x}_i$ are the respective lifetime, label indicator, and covariates for the i th sample. The risk set $\Re(t_i)$ is the set of samples that are still at risk of rail failure at time t_i. As can be observed in Eq. (4), the partial likelihood considers probabilities only for those samples that have experienced rail failure ($E = 1$) and does not explicitly consider probabilities for those samples that are censored ($E = 0$). But the information that censored data contained is preserved in the partial likelihood, i.e., a sample that is censored after the i th lifetime is part of the risk set used to compute L_i even though this sample is censored later. More description of censored data in survival approach can be found in Kleinbaum et al. [35].

2.3. Deep neural network-based survival approach

However, the relationship between covariates and partial hazard is restricted to linear in Cox models, which is often not the case in many practical scenarios [18,36]. Therefore, a more complex family of survival models is necessary to properly capture nonlinearity in the data, offering greater flexibility in modelling the relationship between covariates and partial hazard. Deep neural network is a highly popular modelling technique and has been frequently utilized in literature owing to its capability of fitting highly complex, nonlinear functions. In the case of survival approach, deep neural network is also deployed lately.

Among them, the model proposed by [18] is one of the most popular deep neural network-based survival models, showing outstanding performance in many applications [21,22]. In Katzman et al. [18], the partial hazard is estimated through a multi-layer perceptron (MLP), which comprises two fully connected layers. The parameterization of the partial hazard $g(\mathbf{x})$ is rather straightforward by using the neural network $f_{net}(\cdot)$ to replace linear function $\beta^{\mathrm{T}}\mathbf{x}$ as

$$g(\mathbf{x}_i) = f_{net}(\mathbf{w}, \mathbf{x}_i) \tag{5}$$

where $\mathbf{w}$ denotes the weights of the neural network. To train this neural network, the loss function is set to be the average negative log partial likelihood, where the partial likelihood is similar to that is used in the Cox model.

$$\begin{aligned}\log L &= \log\left(\prod_{i=1}^{n} \frac{\exp[\beta X_i(t_i)]}{\sum_{j\in\Re(t_i)}\exp\left[\beta X_j(t_i)\right]}\right) \\ &= \sum_{i:E_i=1}\left(\beta X_i(t_i) - \log\sum_{j\in\Re(T_i)}\exp\left[\beta X_j\left(t_j\right)\right]\right)\end{aligned} \tag{6}$$

$$L_{loss} = -\frac{1}{n_{E=1}}\sum_{i:E_i=1}\left(\beta X_i(t_i) - \log\sum_{j\in\Re(T_i)}\exp\left[\beta X_j\left(t_j\right)\right]\right) \tag{7}$$

Note that the deep neural network employed in survival approach can vary in the number of hidden layers and units, depending on the specific problem. The type of network can also be customized depending on the structure of covariates. For example, Lee et al. [37] used recurrent neural networks to deal with longitudinal data and Li et al. [36] used attention-based neural networks to process time series for survival analysis. As the structure of covariates in this study is relatively basic and contains neither images nor time series, multiple fully connected layers are therefore used as the deep neural network backbone.

2.4. Monte Carlo dropout as a Bayesian approximation

Although the deep neural network-based survival model is capable of modelling the useful lifetime, it does not offer a confidence interval for each estimation. This implies that the uncertainty in the deep neural network-based survival model cannot be considered. A useful way to represent uncertainty is through Bayesian methods, which involve placing a prior distribution over model parameters and marginalizing them given new observations to obtain an updated distribution [38,39]. In this case, approximate inference methods such as Markov Chain Monte Carlo (MCMC) and variational inference are required. But inferring the posterior distribution in the context of deep neural networks poses a great challenge due to the large number of model parameters involved. Even MCMC method can be impractical because of slow convergence and immense computational costs. To address this issue, Monte Carlo (MC) dropout is introduced [25], which is one of the most popular approximate Bayesian inference methods in practice due to its simplicity and no loss of accuracy.

For a set of training samples with covariates $\mathbf{x} = \{x_1, .., x_n\}$, corresponding outputs $\mathbf{y} = \{y_1, .., y_n\}$, and the weights of an L-layer neural networks $\mathbf{w} = \{w_1, .., w_{node}\}_{i=1}^{L}$, the aim of Bayesian inference is to determine the posterior distribution $p(\mathbf{w}|\mathbf{x}, \mathbf{y})$. Hence, the prediction of an output y^* for an unknown sample with covariate x^* can be made through the posterior distribution over the space of weights given the training samples:

$$p(y^*|x^*, \mathbf{x}, \mathbf{y}) = \int p(y^*|x^*, \mathbf{w})p(\mathbf{w}|\mathbf{x}, \mathbf{y})d\mathbf{w} \tag{8}$$

As the true posterior distribution is intractable in most cases, variational inference (VI) is often applied to deal with this issue. The idea of VI is to approximate the true posterior distribution with the variational distribution $q_\theta(\mathbf{w})$ with parameters θ. This can be achieved by minimizing the Kullback-Leibler (KL) divergence $KL(q_\theta(\mathbf{w})\|p(\mathbf{w}|\mathbf{x}, \mathbf{y}))$, which is a measure of the similarity between the two distributions. The minimization objective for VI can be written as:

$$L_{VI} = -\int q_\theta(\mathbf{w})\log p(\mathbf{y}|\mathbf{x},\mathbf{w})d\mathbf{w} + KL(q_\theta(\mathbf{w})\|p(\mathbf{w})) \tag{9}$$

where the integral term denotes the negative partial log-likelihood function with respect to the expectation of the posterior distribution. The *KL* term serves to alleviate overfitting.

The minimization objective for a neural network with dropout applied $L_{dropout}$ can be expressed as

$$L_{dropout} = \frac{1}{N}\sum_{i=1}^{N} l(y_i, \widehat{y}_i) + \lambda\sum_{i=1}^{L}\|\mathbf{w}_i\|_2^2 \tag{10}$$

where $\widehat{y}$ denotes the output of the neural network and $l(\cdot,\cdot)$ denotes the loss function.

According to the derivation in Gal and Ghahramani [25], the integral term in Eq. (9) can be approximated through Monte Carlo integration with respect to **w** because the Monte Carlo sampling process of parameters **w** from Bernoulli distribution is identical to performing dropout on the neural network layers. Meanwhile, the *KL* term in Eq. (9) corresponds to an L_2 regularization term by some weight decay in dropout networks. Thus, Eq. (9) and Eq. (10) can be rewritten as:

$$\frac{\partial}{\partial\theta}\widehat{L}_{dropout}(\theta) = \frac{1}{N}\frac{\partial}{\partial\theta}\widehat{L}_{VI}(\theta) \tag{11}$$

Eq. (11) shows that optimizing the neural network with dropout operations is equivalent to performing approximate inference within a probabilistic framework for the model.

By replacing the posterior $p(\mathbf{w}|\mathbf{x},\mathbf{y})$ with its variational approximation $q_\theta(\mathbf{w})$, the prediction of output y^* for an unknown sample with covariate x^* can be further calculated as

$$p(y^*|x^*,\mathbf{x},\mathbf{y}) = \int p(y^*|x^*,\mathbf{w})q_\theta(\mathbf{w})d\mathbf{w} \approx \frac{1}{M}\sum_{m=1}^{M} p(y^*|x^*,\widehat{\mathbf{w}}_m) \tag{12}$$

Owing to the mathematical proofs in Gal and Ghahramani [25], the predictive distribution of output can be approximated by collecting the results of *M* times stochastic forward passes through the model during the test process. As a result, the uncertainty within the model can be estimated.

2.5. The proposed deep Bayesian survival model

Combining the aforementioned survival approach, deep neural network, and Monte Carlo dropout, a deep Bayesian survival model, named BNN-Surv is designed for rail useful lifetime modelling as depicted in Fig. 1. The structure of the model follows a configurable feed-forward deep neural network structure: the input to the network is the covariates **x**, consisting of monitoring data, track characteristics data, defects data, and M&R data, which are explained in section 3. The network propagates the inputs through a number of hidden layers with weights **w**. To fully learn the nonlinear relationships in covariates, multiple fully connected layers are constructed. The final layer is a single node that performs a linear function of the learned hidden representations. The output of the last layer is the estimated partial hazard $g(\mathbf{x})$. The architecture facilitates the network in learning potentially nonlinear relationships between covariates and partial hazards. Furthermore, every fully connected layer is succeeded by an MC dropout layer, which serves as an approximation of the variational Bayesian inference to provide uncertainty estimation.

The pseudocode for training and testing the BNN-Surv is illustrated in Algorithm 1. Unlike using standard dropout, the specific operation of MC dropout is to randomly drop some neuron weights during the training and testing process. This can be seen as adding some Bernoulli noises to the original neural network. Because some neurons are randomly dropped during the testing process, different estimation results can be obtained for the same test data each time. After *M* estimations, the mean μ and standard deviation σ can be calculated for all the estimation results, and the final confidence interval can then be estimated. By doing this, approximate Bayesian inference is involved in the

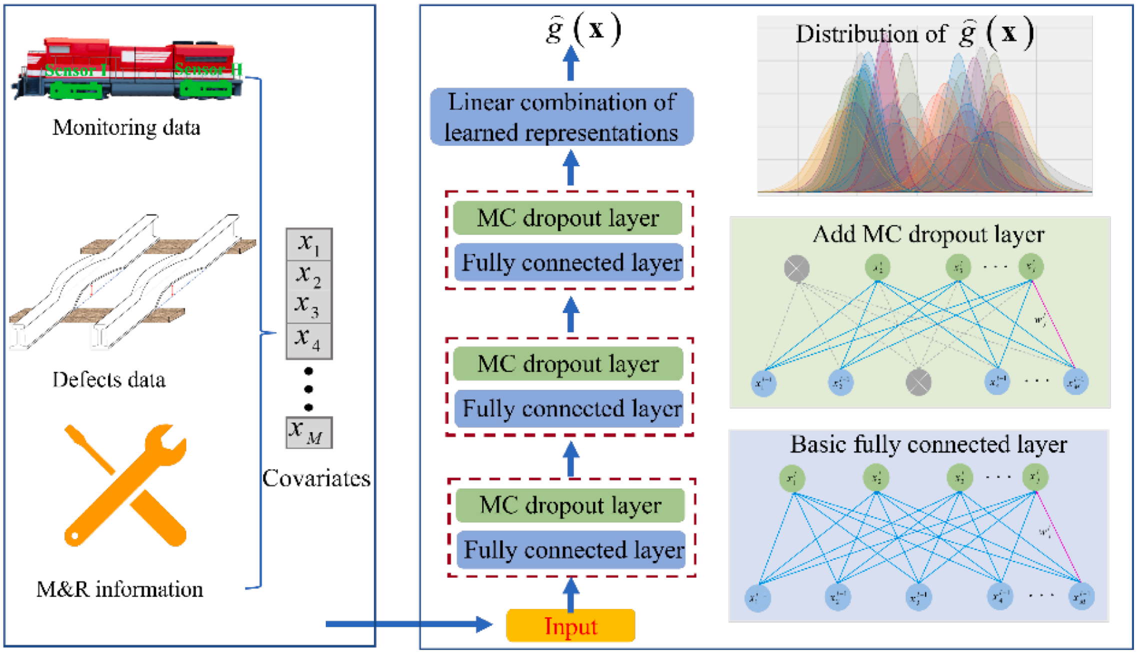


Fig. 1. The architecture of the proposed BNN-Surv model.

neural network.

Algorithm 1. Training and testing phases of BNN-Surv.

```
Input $n$ samples as the training set, $\mathbf{x}$ is input covariates.
Initialize weights $\mathbf{w}$ in network $f_{net}$.
Output $\hat{y}$ is the partial hazard.
Training phase:
  For each epoch do
    Turn on dropout
    Performing stochastic calculations and
        $\hat{y} = f_{net}(w, x_i)$
    Update weights $\mathbf{w}$ in the neural network $f_{net}(\cdot)$
    Restore the dropped neurons
    Compute the loss $L_{loss} = -\frac{1}{n_{E=1}}\sum_{i:E_i=1}\left(\beta X_i(t_i) - \log\sum_{j\in\Re(T_i)}\exp[\beta X_j(t_j)]\right)$
    Determine whether to stop the training process when the loss does not decrease.
  End for
Testing phase:
  For $i = 1, \ldots, M$ do
    Turn on dropout
    Performing stochastic calculations and
        $\hat{y}_i = f_{net}(w_{droppedi}, x_{testi})$
    Restore the dropped neurons
  End for
  $\mu = \frac{1}{M}\sum_{i=1}^{M}\hat{y}_i$
  $\sigma = \sqrt{\frac{1}{M}\sum_{i=1}^{M}(\hat{y}_i - \mu)}$
  Setting $n$ times the standard deviation $\sigma$
  Upper bound of the confidence interval for hazard function:
        $h(t)^{upper} = h_0(t)\exp(\mu + n\cdot\sigma)$
  Lower bound of the confidence interval for hazard function:
        $h(t)^{lower} = h_0(t)\exp(\mu - n\cdot\sigma)$
```

3. Deep Bayesian survival analysis for rail useful lifetime modelling

3.1. Data structure

The dataset available for the current study consists of track geometry monitoring data, track characteristics data, various types of defects data, as well as M&R data. The dataset is collected from a 150 km section of railway tracks in Australia during the period from 2016 to 2021.

3.1.1. Track geometry monitoring data

Track geometry monitoring data is collected from the special track geometry car at 1 m interval with corresponding locations. The track geometry data is collected in an almost constant time interval, every four months. Each time at every 1 m, there are more than 100 measurements, such as curvature, twist, wear, and so on. Using all measurements would result in a high dimensionality of covariates and might decrease the computational efficiency. Besides, some measurements are correlated with each other and might provide redundant information. Therefore, feature selection is performed based on the expert's knowledge and previous studies [40,41]. Table 1 shows the list and description of some selected track geometry monitoring data.

Table 1
The description of some selected track geometry monitoring data.

Geometry monitoring data	Description
Top offset at left rail (top_L)	5 m top chord with a 2 m/3m mid-point offset
Line offset at left rail (line_L)	10 m alignment chord with a centre mid-point offset.
Twist 2 *m*	Measures the change in superelevation over a 2 m interval by calculating the difference between the current xsuper and xsuper delayed by 2 m.
Gauge	The measured distance between the rail gauge points, expressed as a variation from the standard gauge.
Rail head loss	The percentage reduction in the cross-sectional area of the rail head compared to the selected template
Rail horizontal wear	The sum of horizontal wear at the (inner) gauge point and horizontal wear at the outer (field face) gauge point.
Rail vertical wear	Vertical wear at a point 16 mm in from the gauge point.

In addition to the direct use of these monitoring data, some new numerical measurements can be extracted from geometry monitoring data to better represent the condition of the tracks. TQI is one of these and has been widely used to comprehensively quantify the quality of the track condition [42]. There are different types of TQI depending on local standards. Since the study is based on the Australian railroads, the TQI calculation method recommended by the experts' knowledge of Australian Rail Track Corporation is used in this paper [43].

$$TQI = 0.5 \times \left(\sigma_{topleft} + \sigma_{topright} + \sigma_{lineleft} + \sigma_{lineright}\right) + \sigma_{twist} + \sigma_{xGauge} \tag{13}$$

$$\sigma_i = \sqrt{\frac{1}{N_l - 1}\sum_{j=1}^{N_l}\left(g_{ij} - \overline{g}_i\right)^2} \tag{14}$$

$$\overline{g}_i = \frac{1}{N_i}\sum_{j=1}^{N_i} g_{ij} \tag{15}$$

where σ_i is the standard deviation of a single geometry measurement (unit: mm). g_{ij} is the value of the geometry measurement i at location j on the railway tracks. N_i is the number of measurements i in the section of track.

3.1.2. Track characteristics data

Given that the studied railway network consists of heavy haul lines, it is essential to take into account the impact of tonnage on the rails. To facilitate modelling, information on the annual tonnage, which refers to the total weight of trains and freight passing each track section, is collected and analysed.

The insulated joint data needs to be considered when modelling the rail useful lifetime because the nearby rails usually experience failure more frequently according to the experience of patrol inspection staffs and maintenance engineers. In this study, the count of insulated joints is used.

3.1.3. Defects data

According to the literature [42], the existing defects on tracks might have effects on rail failures. So, it is natural to collect defect data for rail useful lifetime modelling. The defects data is acquired through patrol inspection, ultrasonic inspection, ground penetrating radar, and circuit signals with the date and location of occurrence, and defect type recorded. There are many types of defects recorded, such as transverse weld defects, formation failures, squats, and so on. Based on where the defects occur, the defects data is classified into three categories, which are rail defects, geometry defects, and formation defects. The details of the classification are shown in Table 2.

Table 2
The classification of track geometry data.

Defect type	Description
Rail defects	Weld defect
	Surface damage
	Squat and shelling
	Rail head split
	Rail head transverse crack
Geometry defects	Twist fault
	Top fault
	Gauge exceeds
Formation defects	Mud pumping
	Formation failure
	Ballast fouled

3.1.4. Maintenance and replacement data

Maintenance and replacement (M&R) activities would change the condition of the tracks and directly affect the rail useful lifetime. Thus, M&R data needs to be considered in modelling. However, there are different types of M&R activities, and they have different degrees of influence on the condition of tracks. For example, taking rail joint replacement and re-railing can be regarded as resetting the rail to brand-new condition, whereas, tamping and grinding are only partial and temporary repairs of the rail surface and formation. Thus, the M&R activities need to be classified based on the degrees of M&R and the components being maintained. In this study, according to the rail maintenance engineers' knowledge, the M&R activities are classified into four categories including renewal M&R, surface M&R, geometry M&R, and formation M&R. The details of classification are shown in Table 3.

3.2. Survival data processing

Unlike classical classification and regression approaches, survival analysis requires a dataset including the following three pieces of information: 1) observed covariates $\mathbf{x}$, 2) useful lifetime T, and 3) label E whether the lifetime is fully observed or partially observed. Therefore, a unique survival dataset needs to be constructed for model development. The detailed processing procedures are as follows:

1) Segment division: as this study attempts to perform a segment-based useful lifetime modelling rather than in a large region, the continuously studied track sections are divided into adjacent segments. According to the recommendation from Bai et al. [9], the length of segment is set to 1 km.
2) Useful lifetime: the real rail useful lifetime is the time interval from rail starting time to the failure time. Fig. 2 shows the timeline of a typical track segment. Since the actual time that the rail put into use was not accurately recorded, in this study, the first starting time is counted from the first time that the rail experiences a rail replacement (one of the maintenance activities). If rail failures occur n_i times in segment i, lifetime should be calculated as many as $n_i + 1$ times for that segment. The $(1, 2, \ldots, n_i)^{\text{th}}$ lifetime is the time interval from the time of rail replacement to the next rail failure that occurs. The $(n_i + 1)^{\text{th}}$ lifetime is not known, which is censoring. As one can see, the real rail useful lifetime is determined based on the replacement and rail failure records. The location and date of each actual replacement and rail failure event were provided by the Australia Railway Track Corporation.
3) Event indicator: when the lifetime is associated with a rail failure, the event indicator is set to 1. If no rail failure is observed until the end of the study horizon, the event indicator is set to 0.
4) Mapping covariates: all the aforementioned datasets explained in section 3.1 are mapped to the corresponding segment and lifetime. The covariates that are used in this study are summarized in Table 4.

Based on historical records, 526 samples are created, 194 of which have exact lifetimes, whereas the rest samples are censored. Each sample represents a 1 km segment of track, with actual lifetime, label, and covariates, where covariates include geometry monitoring data, track characteristics, defects data, and M&R data. Fig. 3 shows the distribution of some of the covariates and their relationships with one another. The histograms on the diagonal illustrate the distribution for each covariate, while the scatter plots on the upper and lower triangles show the relationships between two covariates. It can be seen from the scatter plots that there is no significant correlation between most of the covariates.

Table 3
The classification of M&R activities.

M&R category	Description
Renewal M&R	Rail joint replacement Rail defect removal Renewal Rerailing
Surface M&R	Grinding
Geometry M&R	Track reconditioning Undercutting
Formation M&R	Tamping Ballast cleaning Drainage works

4. Implementation and results

In this section, the proposed BNN-Surv is applied to a real-life case from the railway tracks in Australia to validate its effectiveness. Typical results obtained from the proposed survival approach are first analysed and explained. Then, the proposed survival approach is evaluated through two metrics, i.e., concordance index (C-index) and root mean square error (RMSE), and compared with three commonly used survival models, i.e., Cox, Weibull model, random survival forest models. The uncertainty estimation capability of the proposed approach is also discussed through a proposed metric that balances coverage probability and interval width.

4.1. Performance metrics

To evaluate the estimation performance of the survival model, two metrics are applied including concordance index and root mean square error.

4.1.1. Concordance index (C-index)

The concordance index (C-index), as proposed by Harrell [32], is a widely used metric for assessing the quality and efficiency of a survival model. It is a ranking-related score that assesses how close the ranking order of estimated lifetimes is to the ranking order of real lifetimes. The C-index is founded upon the assumption that segment with longer lifetime should be assigned a greater estimated lifetime than segment with shorter lifetime. The score ranges from 0 to 1, with a larger score indicating the better performance of the model. C-index is calculated as

$$C - index = \frac{1}{n} \sum_{i:E_i=1} \sum_{j:L_i^{real} < L_j^{real}} 1_{L(x_i) < L\left(x_j\right)} \tag{16}$$

where $L(x)$ denotes the estimated useful lifetime and L^{real} denotes the real observed lifetime. Eq. (16) counts the times a model estimates $L(x_i) < L\left(x_j\right)$ when observed $L_i^{real} < L_j^{real}$ holds true over the total number of comparable cases, which is represented by n.

4.1.2. Root mean square error (RMSE)

Another commonly used metric to evaluate the survival model is root mean square error (RMSE), which indicates the discrepancy between the estimated and real values.

$$RMSE = \frac{1}{n} \sum_{i=1}^{N} RMSE_i = \frac{1}{n} \sum_{i=1}^{N} \sqrt{\left[L_i - L_i^{real}\right]^2} \tag{17}$$

where L_i denotes the estimated useful lifetime and L_i^{real} denotes the real useful lifetime. RMSE reflects the deviation degree of the estimated and the real useful lifetime. So, the lower the RMSE, the better the modelling.

4.2. Model structural selection

Before performing the evaluation, it is necessary to determine the network structural parameters that produce the best results. In this

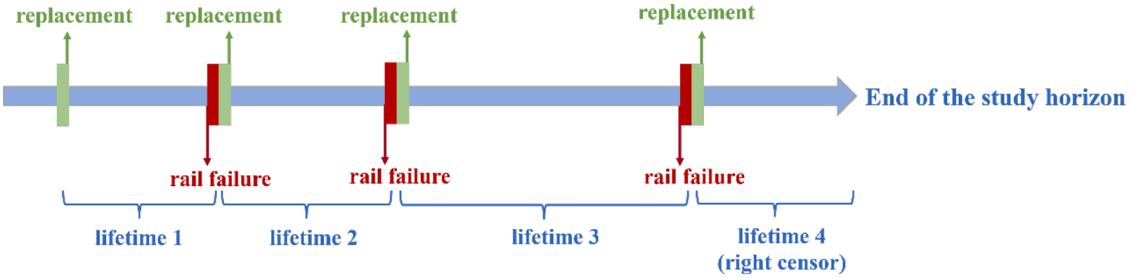


Fig. 2. Timeline of a typical track segment.

Table 4
Details on the description of the survival covariates.

No.	Covariates	Description
1	Rail head loss	The latest rail head loss before the end of lifetime.
2	Rail horizontal wear	The latest rail horizontal wear before the end of lifetime.
3	Rail vertical wear	The latest rail vertical wear before the end of lifetime.
4	Curvature	The latest curvature before the end of lifetime.
5	TQI	The latest TQI before the end of lifetime.
6	Tonnage	The annual tonnage on the segment
7	Insulated joint	Number of insulated joints exist in the segment
8	Rail break	Number of rail breaks occurred before the start of lifetime.
9	Rail defects	Number of rail defects reported before the end of lifetime.
10	Geometry defects	Number of geometry defects reported before the end of lifetime.
11	Formation defects	Number of formation defects reported before the end of lifetime.
12	Renewal M&R	Number of renewal M&R reported before the start of lifetime.
13	Surface M&R	Number of surface M&R reported before the end of lifetime.
14	Geometry M&R	Number of geometry M&R reported before the end of lifetime.
15	Formation M&R	Number of formation M&R reported before the end of lifetime.

study, grid search tuning is used to optimize the network structural parameters of the model. The network structural parameters include number of hidden layers, number of nodes in hidden layers, and learning rate. The tunning experiments are conducted on the dataset via 5-fold cross-validation. The mean of C-index and standard deviation (Std) of the five runs are used to measure the model's performance.

4.2.1. *Number of hidden layers*

The number of hidden layers plays a vital role in the performance of neural networks. However, increasing the number of hidden layers might not always guarantee performance improvement, which depends on the complexity of the problems that are being solved [44]. According to the previous studies [8,40,41,45], the values for the number of hidden layers for tunning are selected between 1 and 5 with the ReLU as activation function. Fig. 4 (a) shows the results of different numbers of hidden layers, where the left vertical axis indicates the mean of C-index and the right vertical axis indicates the Std of five runs. It can be seen from Fig. 4 (a) that as the number of hidden layers increases, C-index gradually increases whereas Std decreases, and there is an inflection point when the number of hidden layers is 3. After the inflection point, as the number of hidden layers increases, C-index gradually decreases whereas Std increases. It can be concluded that the number of hidden layers set to 3 gives optimal results.

4.2.2. *Number of nodes in hidden layers*

The values for the number of nodes for tunning are selected as 8, 16, 32, 64, 128, and 256 with other parameters fixed. The reason for using a power of 2 as the number of nodes is that the complexity of efficient algorithms is usually measured on the order of log base 2 [46]. Fig. 4 (b) shows the results of different numbers of nodes in hidden layers. It can be seen from Fig. 4 (b) that as the number of nodes increases, C-index gradually increases whereas Std decreases and there is an inflection point when the number of nodes is 32. After reaching the inflection point, there is a decrease in the C-index. When the number of nodes reaches 128, the C-index begins to rise again. But it can be seen from Fig. 4 (b) that the increase in the number of nodes does not get a significant performance improvement. Thus, considering the computational efficiency, the number of nodes in hidden layers is set to 32.

4.2.3. *Learning rate*

The value for learning rate can affect the updating speed of parameters during neural network training. In this paper, the values of learning rate for tunning are selected as 0.0001, 0.001, 0.01, and 0.1, which are commonly used in neural networks [45]. Fig. 4 (c) shows the results of different learning rates. It can be seen from Fig. 4 (c) that as the learning rate increases, C-index gradually increases whereas Std decreases, and there is an inflection point when the learning rate is 0.001. After the inflection point, as the learning rate increases, C-index gradually decreases whereas Std increases. Thus, it can be concluded that 0.001 is the optimal learning rate.

The network structure determined above has incorporated the MC dropout layer, which can provide confidence interval for the model output when the dropout is activated during test process. To evaluate the effect of MC dropout on model performance, a competing model is constructed, i.e., 'proposed model without MC dropout'. The effect of the MC dropout on the model performance is measured with the C-index. The comparative results show that the proposed model without MC dropout acquires the C-index of 0.769, while the proposed model with MC dropout can achieve the C-index of 0.802 under the same circumstance. This indicates that adding MC dropout not only can provide uncertainty estimation but also improve performance.

In general, the model is trained via stochastic gradient descent along with Adam optimizer and Eq. (7) as the loss function. The BNN-Surv model is both smooth and differentiable, thereby allowing for the model's parameters to be learned through standard backpropagation.

4.3. *Model implementation*

To train and evaluate the BNN-Surv model in this study, the dataset is randomly divided into two sets: the training set, which comprises 80% of the data, and the testing set, which comprises the remaining 20%. Once the BNN-Surv model has been trained, each test is repeated 500 times with the MC dropout on. In this way, the distribution of partial hazard $g(\mathbf{x})$ for each sample in the testing set can be estimated as shown in Fig. 5, where each colour represents a different sample.

According to Eq. (3), the survival curve (graphic representation of the survival probability S(t)) with confidence interval can be derived for each sample. Fig. 6 shows the typical survival curves for two different samples. The x-axis is the lifetime in day. The y-axis is the survival probability, in which 1.0 means 100% survival at a certain time and 0.0

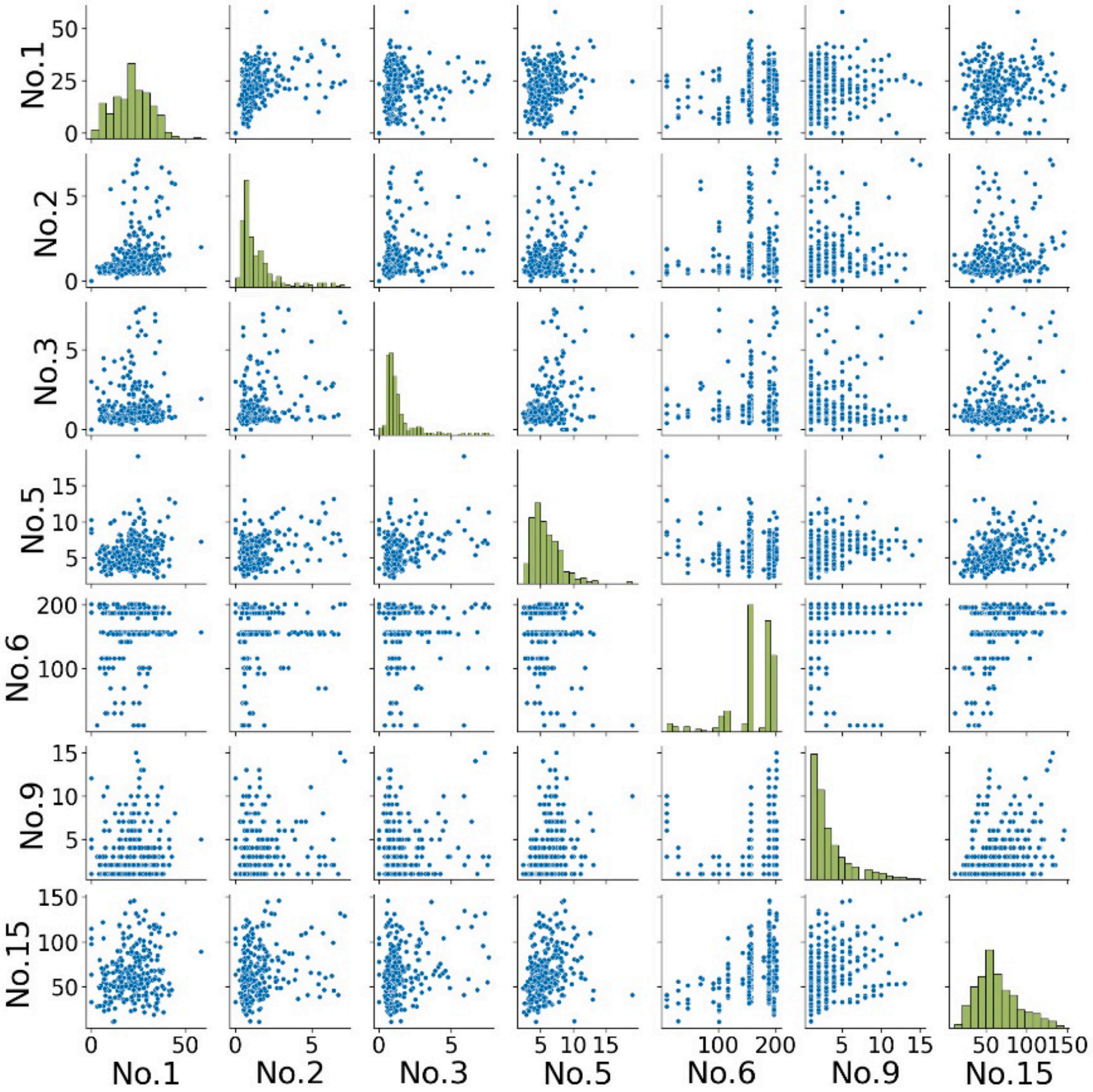
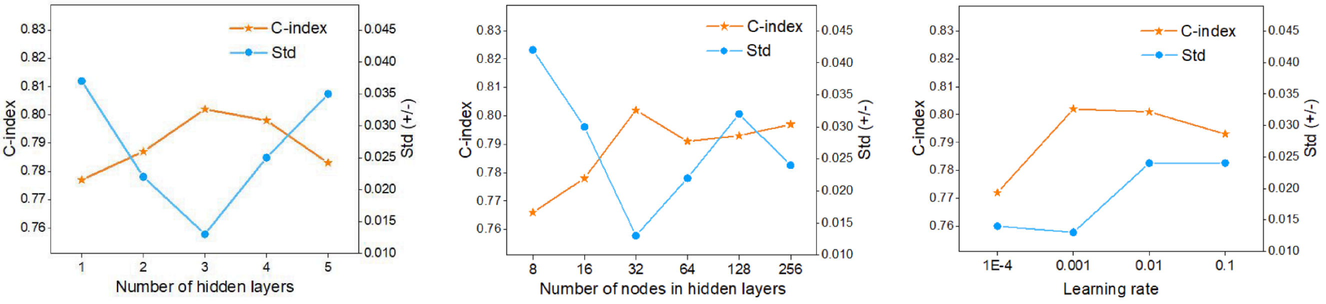


Fig. 3. Statistical distributions of some of the covariates and their relationships with one another. No. 1 is rail head loss; No. 2 is rail horizontal wear; No. 3 is rail vertical wear; No. 5 is TQI; No. 6 is tonnage; No. 9 is rail defects; No. 15 is formation M&R.



(a) Number of hidden layers

(b) Number of nodes in hidden layers

(c) Learning rate

Fig. 4. C-index and Std of different (a) Number of hidden layers, (b) Number of nodes in hidden layers, and (c) Learning rate.

means 0% survival at a certain time. From Fig. 6, one can estimate the probability that the rail still works safely at any certain time.

As can be seen in Fig. 6, the outputs of survival model are the time-dependent survival probabilities that are computed through the time-dependent hazard rates, rather than the exact lifetime values. Estimation of rail useful lifetime helps railroads to take timely maintenance to avoid catastrophic failure. Given the outputs of survival model, the useful lifetime can be estimated as

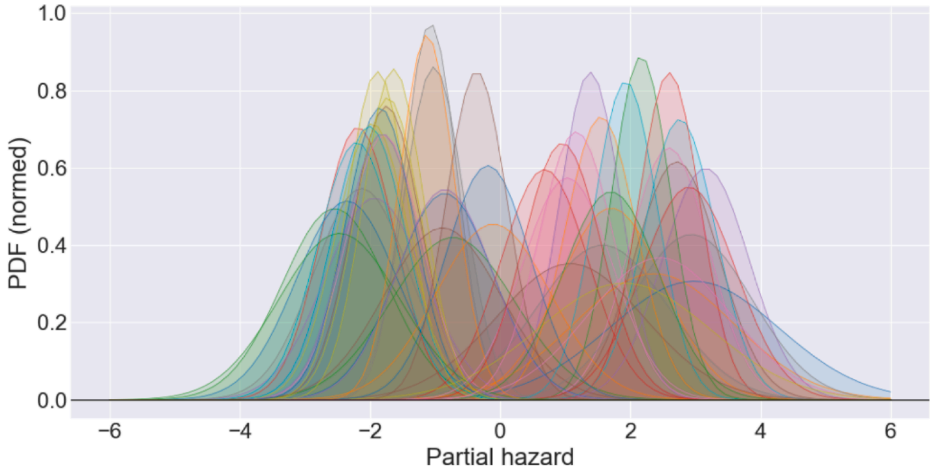


Fig. 5. The distribution of survival model's outputs (partial hazard $g(\mathbf{x})$) for the testing set. Each colour represents a different sample.

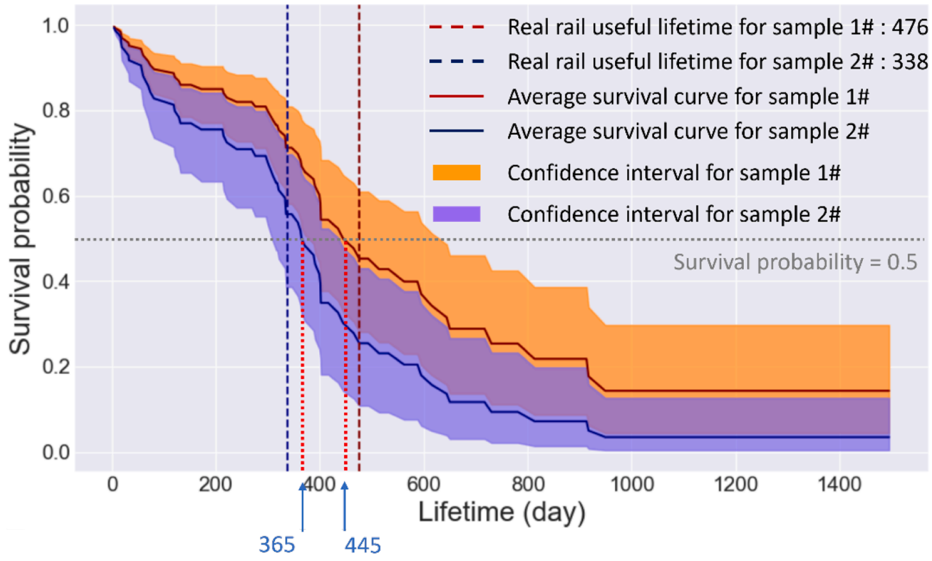


Fig. 6. The survival curves of two different samples.

$$L_i = \underset{t}{\mathrm{argmin}}\{S_i(t)\ \langle S_{threshold}\} \tag{18}$$

where $S_i(t)$ denotes the estimated survival probabilities over time t. $S_{threshold}$ is a threshold survival probability defining a risky operating situation. Typically, median or mean lifetime is used to represent the potential lifetime of components based on the assumption that components tend to fail when the survival probabilities are less than 0.5 [22]. Thus, in this study, the $S_{threshold}$ is set to be 0.5. For the two samples in Fig. 6, the survival probability of 0.5 corresponds to 445 days and 365 days, which are the estimated useful lifetimes. The real useful lifetimes of these two samples are with 476 days and 338 days, represented by vertical dash lines. It can be seen from Fig. 6 that the estimated lifetimes are very close to the real lifetimes.

4.4. Model evaluation

The C-index results for 500 times with the MC dropout activated are displayed in Fig. 7. It is observed that the C-index of the BNN-Surv model is roughly around 0.8 by a variance of 0.02. According to Steck et al. [47], a C-index value ranging from 0.6 to 0.7 typically indicates a well-fitted model, whereas a value closer to 0.5 indicates that the model does not predict the target value better than random chance. The C-index

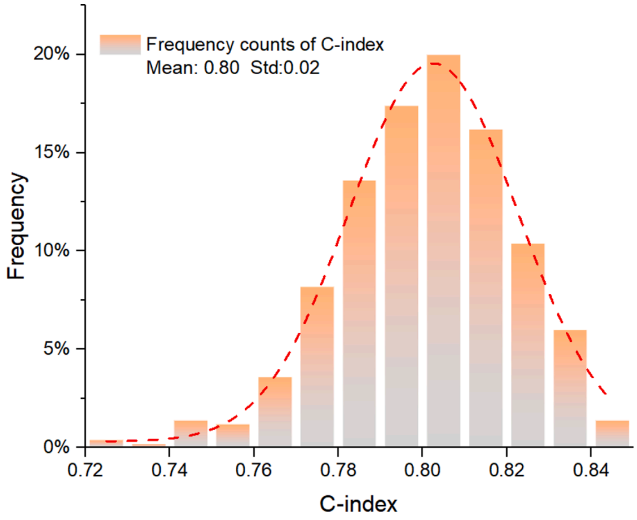


Fig. 7. The distribution of C-index results for 500 times with the MC dropout on.

results obtained by BNN-Surv indicate the proposed model has an excellent ability in ranking a sample's useful lifetime.

To demonstrate the superiority of the proposed BNN-Surv, three commonly used survival models in the field of transportation systems are applied for comparisons. Two of them are classical survival models including Cox and Weibull models, while the other one is a machine learning-based survival model, i.e., random survival forest model (RF-Surv) [48]. In addition, to verify the advantages of considering censored data in rail useful lifetime modelling, two data conditions are constructed to compare the performance considering censored data and not considering censored data. In this way, 8 survival models are created by combing two data conditions and four survival approaches. As the BNN-Surv approach generates the distribution of C-index results, the mean value of C-index results is used for comparison with other models.

Table 5 compares the performances of the different models, with the highest C-index highlighted in bold. From Table 5, it can be observed that the BNN-Surv model proposed in this study outperforms the other three survival models regarding two data conditions, which achieves the highest C-index of 0.8. The results demonstrate the BNN-Surv model's superior performance in modelling rail useful lifetime, possibly owing to significant non-linearities in the covariates, which can be better captured by a neural network as opposed to a linear model. Moreover, as the results are shown in Table 5, the Cox, Weibull, RF-Surv, and BNN-Surv models without considering censored data achieve the C-index of 0.64, 0.67, 0.65, and 0.69 respectively. When censored data is considered in the modelling, the C-index of Cox model is improved to 0.71, Weibull model is improved to 0.74, RF-Surv model is improved to 0.78, and BNN-Surv is improved to 0.80, showing noticeable improvement. This indicates that ignoring or otherwise mistreating the censored data might lead to undesirable results.

In general, the proposed BNN-Surv achieves a C-index of 0.8, indicating that the estimated useful lifetime ranking of most samples is consistent with the real one. The correct ordering of useful lifetime is of great importance for practical predictive maintenance. Based on the ranking results, asset managers can develop more economical and targeted maintenance plans.

To get an intuition of how accurate the useful lifetime estimation is by the proposed survival model, RMSE is used to measure the degree of error between the estimated lifetimes and the real values. The distribution of RMSE results is shown in Fig. 8. It can be seen from Fig. 8 that the mean value of the distribution is about 189 days, while the standard deviation of the distribution is 22 days.

In addition, the estimated lifetimes obtained by BNN-Surv are used to compare with the other three commonly used survival models, Cox, Weibull, and RF-Surv models. As the BNN-Surv approach generates the distribution of results, the average results of 500 times are used for comparison. Also, the samples only to be uncensored are chosen for comparison because they have real lifetimes.

Comparison results are shown in Fig. 9. The subplot shows the RMSEs for the four models. It can be seen from the subplots in Fig. 9 that BNN-Surv achieves the lowest RMSE of 189, which can demonstrate the proposed model's effectiveness. Compared to the other three models, the estimated lifetimes obtained by BNN-Surv are closer to real values. However, the RMSEs achieved by the four models are not very desirable, and for some samples, there is a significant difference between the estimated and real lifetimes. For example, for sample #8, the real lifetime is 1200 days, while the estimated lifetimes by all three models are below 300 days, which results in a difference larger than 900 days (almost 3 years). One possible reason is that the collected replacement data of some segments are inadequate and do not reflect the track condition well, thus causing a large bias in estimating the lifetimes of these samples. In this case, future efforts will focus on collecting the real-time monitoring data and integrating it into the model, which is expected to better reflect the rail condition and further improve the performance.

Table 5
C-index for considering censored data and not considering censored data based on Cox, Weibull, RF-Surv, and BNN-Surv models.

Data condition	Cox	Weibull	RF-Surv	BNN-Surv
Not considering censored data	0.64	0.67	0.65	0.69
Considering censored data	0.71	0.74	0.78	**0.80**

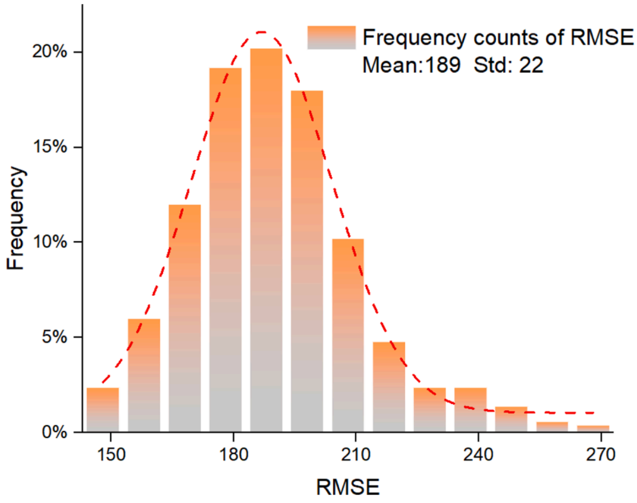


Fig. 8. The distribution of RMSE results for 500 times with the MC dropout on.

Previous study [9] proposed a predictive model based on Markov stochastic process to estimate the rail useful lifetime. In their paper, the difference between the estimated rail useful lifetime and real lifetime is used as the performance metric. The average difference achieved by Bai et al. [9] was 180 days. Whereas the proposed approach (BNN-Surv) in this paper achieves an average difference of 151 days. This shows that the proposed approach in this study performs better.

4.5. Uncertainty estimation

As aforementioned in section 4.4, accurate prediction of useful lifetime is extremely difficult, almost every sample has a difference between the estimated and real useful lifetime. By taking the uncertainty of model into account, the BNN-Surv can provide the confidence interval of the estimated lifetime, which is more appropriate than point estimation. It is obvious that enlarging the width of the confidence interval allows the estimation interval to cover more of the target lifetime, but it also accompanies by an increase in uncertainty. Therefore, it is necessary to find a metric to balance the coverage probability and the width of the confidence interval. Based on the evaluation metric used in Li et al. [49], a new metric is proposed in this study to evaluate the estimation performance and find the optimal interval width, called CW-index:

$$\mathrm{CW}\text{ - }\mathrm{index} = \mathrm{sigmoid}(C + W) \tag{19}$$

$$C = \frac{1}{N}\sum_{i=1}^{N}\xi_i(I(x_i), y_i) \tag{20}$$

$$\xi_i(I(x_i), y_i) = \begin{cases} 1 y_i \in I(x_i) \\ 0 \text{ otherwise} \end{cases} \tag{21}$$

$$W = \frac{1}{N}\sum_{i=1}^{N}\exp\left(I^l(x_i) - I^u(x_i)\right) \tag{22}$$

where C denotes the coverage probability, calculating the number of target lifetimes covered by the estimation interval and W denotes the normalized averaged width. The number of samples to be estimated is denoted by N and the envelope of the estimation interval is denoted by

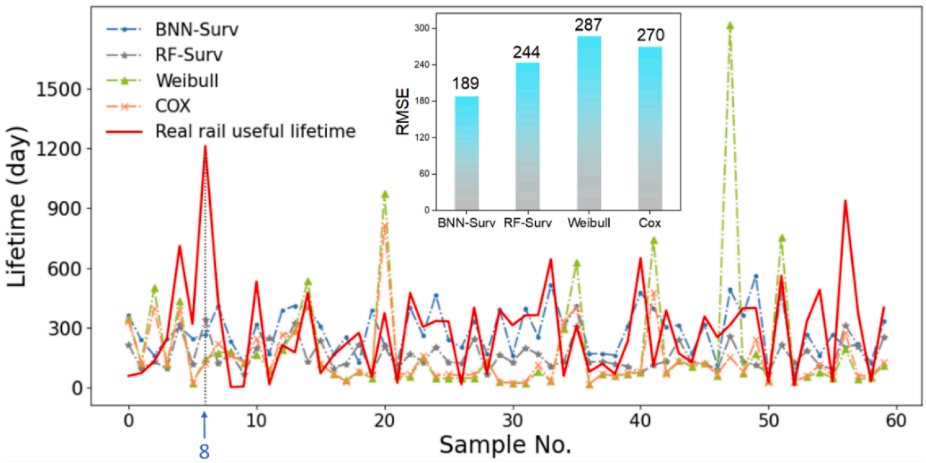


Fig. 9. Comparisons between the real lifetimes and estimated lifetimes obtained from the Cox, Weibull, RF-Surv, and BNN-Surv models.

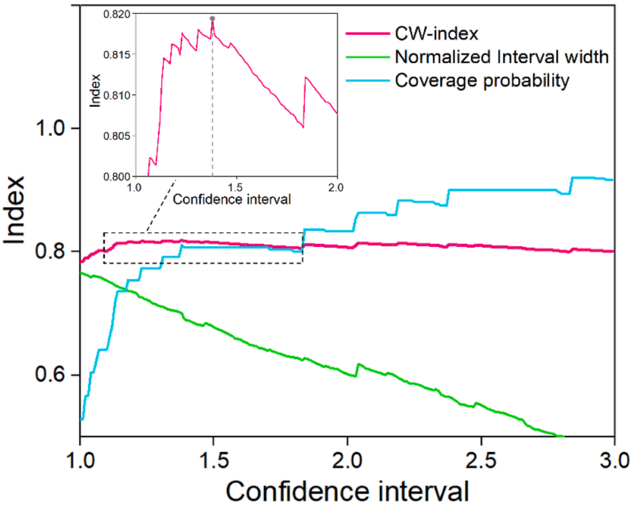


Fig. 10. The results of CW-index, coverage probability, and normalized averaged width under different confidence intervals.

$I(x_i)$. The upper and lower bounds of the estimation interval are denoted by $I^u(x_i)$ and $I^l(x_i)$, respectively. A sigmoid function is applied to the sum of C and W to ensure that the CW-index is a score within the range of 0 to 1. Thus, the larger the CW-index, the better the uncertainty estimation performance.

Fig. 10 shows the results of CW-index, coverage probability, and normalized averaged width at different confidence intervals. The coverage probability and CW-index are significantly affected by changes in the confidence interval. Specifically, the CW-index first increases and then decreases as the confidence interval increases. This pattern can be explained by the fact that the value of CW-index is predominantly governed by C when the confidence interval is small, and by W when the confidence interval is large. As can be seen in Fig. 10, the highest value of CW-index is achieved when the confidence interval falls between 1.2 and 1.4. Thus, for the subsequent analysis, a confidence interval of 1.38 is adopted, which equates to an estimation interval equal to the mean plus/minus 1.38 times the standard deviation.

The estimation results by BNN-Surv are shown in Fig. 11. The shadow area denotes the estimated useful lifetimes with a confidence interval of 1.38 from MC dropout uncertainty estimation. Points within the lower and upper envelopes of the interval possess varying probabilities of occurrence, where those closer to the mean of interval having higher probabilities and those farther away having lower probabilities. Furthermore, two subplots display the distribution of useful lifetime estimations for two samples (i.e., sample #3 and sample #30), where the star symbol denotes the real lifetime, for comparison with the probability distribution obtained from the BNN-Surv model.

As can be seen from Fig. 11, the average lifetimes generated by the BNN-Surv model are very close to the real lifetimes. 81% of the real

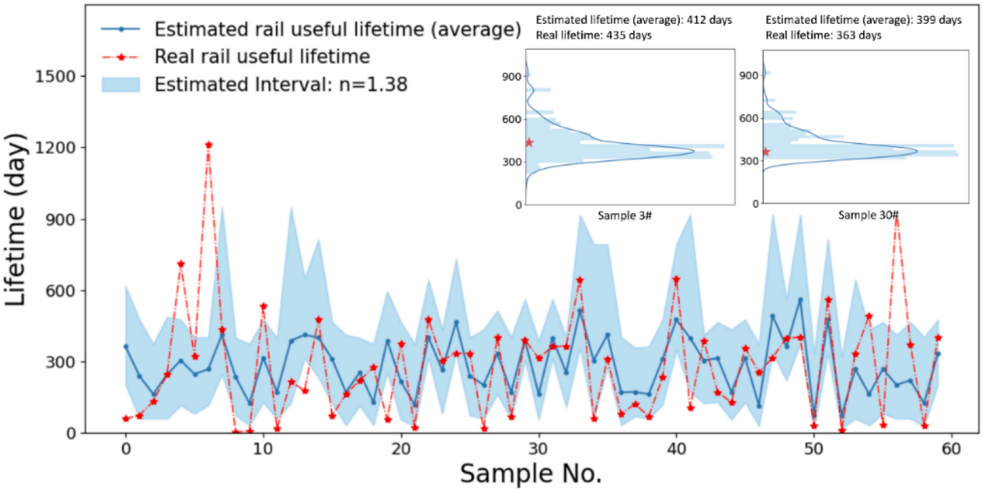


Fig. 11. The estimation results of rail useful lifetime by using BNN-Surv based on the confidence interval of 1.38.

lifetimes are within the envelope of the estimation interval. On a closer examination of the subplots, it is found that the distribution of estimated lifetimes follows an approximately normal distribution. The real useful lifetimes of sample #3 and sample #30 are bounded by confidence intervals. All of these results indicate that the BNN-Surv provides a more conservative and safer estimation compared to the point estimation. Even with some level of uncertainty, such conservative estimations are deemed acceptable in the railway transportation industry due to their significant implications for the economy and personal safety.

5. Conclusion

This paper proposes a deep Bayesian survival approach named BNN-Surv to properly handle censored data for rail useful lifetime modelling. The proposed BNN-Surv model uses a deep neural network as the hazard rate to capture the non-linear relationship between covariates and useful lifetime. To consider and quantify uncertainty in the model, Monte Carlo dropout, regarded as the approximate Bayesian inference, is incorporated into the deep neural network to provide the confidence interval of the estimated useful lifetime. The proposed approach is implemented on a four-year dataset including track geometry monitoring data, track characteristics data, various types of defects data, as well as M&R data collected from a section of railway tracks in Australia.

Extensive comparative studies are conducted to show the effectiveness of the proposed approach. The results obtained allow the following conclusions to be drawn:

1) The results show that considering the censored data significantly outperforms the case of not considering the censored data regarding the C-index. This demonstrates the importance of using survival approach to handle censored data for rail useful lifetime modelling.
2) By comparing with the commonly used survival models, i.e., Cox, Weibull, and RF-surv approaches, the proposed approach shows superior performance. The proposed BNN-Surv can achieve a C-index of 0.80, while Cox, Weibull, and RF-Surv only reach 0.71, 0.74, and 0.78 respectively. In terms of rail useful lifetime estimation, the estimated lifetimes obtained by BNN-Surv are also more approaching to the real lifetimes compared to Cox, Weibull, and RF-Surv. This superiority might indicate that there have significant non-linearities in the covariates that a neural network would benefit from.
3) Through uncertainty estimation, the confidence interval of 1.38 by the BNN-Surv has an 81% correct coverage rate of the real lifetimes. The results demonstrate that BNN-Surv is safer and more trustworthy than the point estimation. In railway transportation, which is related to huge economic and personal safety, this trustworthy rail useful lifetime estimation is extremely important.

By implementing the proposed approach, the rail useful lifetime of each segment can be estimated, which helps railroads to optimize predictive maintenance. For example, taking grinding planning or replacement of those segments have shorter estimated lifetimes. Although this paper has focused specifically on rail useful lifetime modelling, the proposed approach can also be adapted for modelling other products and mechanical components' lifetimes. The proposed approach in this study is a kind of data-driven model that would benefit from larger and more diverse datasets. In future work, additional datasets such as daily dynamic response monitoring data and GIS information are expected to collect, so that more complex relationships between data and rail useful lifetime with uncertainty can be modelled.

CRediT authorship contribution statement

Cheng Zeng: Methodology, Validation, Software, Visualization, Formal analysis, Writing – original draft. **Jinsong Huang:** Conceptualization, Supervision, Resources, Writing – review & editing. **Hongrui Wang:** Software, Writing – review & editing. **Jiawei Xie:** Methodology, Data curation, Investigation. **Yuting Zhang:** Conceptualization, Investigation.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgment

This study was supported by the Australian Government through the Australian Research Council's (ARC) Linkage Projects funding scheme (project LP160101254).

References

[1] Orringer O. Control of rail integrity by self-adaptive scheduling of rail tests. Federal Railroad Administration: United States; 1990.
[2] Zhao J, Chan AHC, Burrow MPN. Probabilistic model for predicting rail breaks and controlling risk of derailment. Transp Res Rec 2007;1995(1):76–83.
[3] Jeong DY, Gordon J. Evaluation of rail test frequencies using risk analysis. Joint Rail Conference 2009:23–30.
[4] Vesković S, Tepić J, Ivić M, Stojić G, Milinković S. Model for predicting the frequency of broken rails. Metalurgija 2012;51:221–4.
[5] Dick CT, Barkan CPL, Chapman ER, Stehly MP. Multivariate statistical model for predicting occurrence and location of broken rails. Transp Res Rec 2003;1825(1):48–55.
[6] Schafer D, Barkan C. A hybrid logistic regression/neural network model for the prediction of broken rails. Proceedings of the 8th World Congress on Railway Research, Seoul, Korea2008a.
[7] Zhang Z, Zhou K, Liu X. Broken rail prediction with machine learning-based approach. ASME/IEEE Joint Rail Conference: American Society of Mechanical Engineers; 2020. p. V001T08A14.
[8] Ghofrani F, Sun H, He Q. Analyzing risk of service failures in heavy haul rail lines: a hybrid approach for imbalanced data. Risk Anal 2022;42(8):1852–71.
[9] Bai L, Liu R, Wang F, Sun Q, Wang F. Estimating railway rail service life: A rail-grid-based approach. Transp Res A Policy Pract 2017;105:54–65.
[10] Kittaneh OA, El-Beltagy MA. Efficiency estimation of type-I censored sample from the Weibull distribution based on sup-entropy. Commun Stat Simul Comput 2017;46(4):2678–88.
[11] Ghofrani F, Chava NK, He Q. Forecasting risk of service failures between successive rail inspections: a data-driven approach. J Big Data Anal Transport 2020;2(1):17–31.
[12] Chi Z, Lin J, Chen R, Huang S. Data-driven approach to study the polygonization of high-speed railway train wheel-sets using field data of China's HSR train. Measurement 2020;149:107022.
[13] Alemazkoor N, Ruppert CJ, Meidani H. Survival analysis at multiple scales for the modeling of track geometry deterioration. Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit 2018;232(3):842–50.
[14] Sresakoolchai J, Kaewunruen S. Prognostics of unsupported railway sleepers and their severity diagnostics using machine learning. Sci Rep 2022;12:1–10.
[15] Sresakoolchai J, Kaewunruen S. Wheel flat detection and severity classification using deep learning techniques. Insight-Non-Destructive Testing and Condition Monitoring 2021;63(7):393–402.
[16] Sresakoolchai J, Kaewunruen S. Track geometry prediction using three-dimensional recurrent neural network-based models cross-functionally co-simulated with BIM. Sensors 2022;23:391.
[17] Sresakoolchai J, Hamarat M, Kaewunruen S. Automated machine learning recognition to diagnose flood resilience of railway switches and crossings. Sci Rep 2023;13:2106.
[18] Katzman JL, Shaham U, Cloninger A, Bates J, Jiang T, Kluger Y. DeepSurv: personalized treatment recommender system using a Cox proportional hazards deep neural network. BMC Med Res Method 2018;18:1–12.
[19] Lee C, Zame W, Yoon J, Van Der Schaar M. Deephit: A deep learning approach to survival analysis with competing risks. Proceedings of the AAAI Conference on Artificial Intelligence. 2018.
[20] Giunchiglia E, Nemchenko A, van der Schaar M. Rnn-surv: A deep recurrent model for survival analysis. Artificial Neural Networks and Machine Learning–ICANN 2018: 27th International Conference on Artificial Neural Networks, Rhodes, Greece, October 4-7, 2018, Proceedings, Part III 27: Springer; 2018. p. 23-32.
[21] Mantouka EG, Fafoutellis P, Vlahogianni EI. Deep survival analysis of searching for on-street parking in urban areas. Transport Res Part C Emerg Technol 2021;128:103173.

[22] Kostic B, Loft MP, Rodrigues F, Borysov SS. Deep survival modelling for shared mobility. Transport Res Part C Emerg Technol 2021;128:103213.

[23] Damianou A, Lawrence ND. Deep gaussian processes. Artificial Intelligence and Statistics: PMLR; 2013. p. 207-15.

[24] Hernández-Lobato JM, Adams R. Probabilistic backpropagation for scalable learning of bayesian neural networks. Int Conf Mach Learn PMLR 2015:1861–9.

[25] Gal Y, Ghahramani Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. Int Conf Mach Learn PMLR 2016:1050–9.

[26] Kendall A, Gal Y. What uncertainties do we need in bayesian deep learning for computer vision? Adv Neural Inf Proces Syst 2017;30.

[27] Leibig C, Allken V, Ayhan MS, Berens P, Wahl S. Leveraging uncertainty information from deep neural networks for disease detection. Sci Rep 2017;7:1–14.

[28] Kendall A, Badrinarayanan V, Cipolla R. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding. ArXiv Preprint ArXiv:151102680. 2017.

[29] Miller D, Nicholson L, Dayoub F, Sünderhauf N. Dropout sampling for robust object detection in open-set conditions. In: 2018 IEEE International Conference on Robotics and Automation (ICRA): IEEE; 2018. p. 3243–9.

[30] Nair T, Precup D, Arnold DL, Arbel T. Exploring uncertainty measures in deep networks for multiple sclerosis lesion detection and segmentation. Med Image Anal 2020;59:101557.

[31] Gal Y, Islam R, Ghahramani Z. Deep bayesian active learning with image data. International Conference on Machine Learning: PMLR; 2017. p. 1183-92.

[32] Harrell FE. Regression modeling strategies. Bios 2017;330:14.

[33] Cox DR. Regression models and life-tables. J Roy Stat Soc: Ser B (Methodol) 1972; 34:187–202.

[34] Lin DY. On the Breslow estimator. Lifetime Data Anal 2007;13(4):471–80.

[35] Kleinbaum DG, Klein M, Kleinbaum DG, Klein M. Introduction to survival analysis. Survival Analysis: A Self-Learning Text; 2012. p. 1–54.

[36] Li X, Krivtsov V, Arora K. Attention-based deep survival model for time series data. Reliab Eng Syst Saf 2022;217:108033.

[37] Lee C, Yoon J, Schaar MVD. Dynamic-DeepHit: A deep learning approach for dynamic survival analysis with competing risks based on longitudinal data. IEEE Trans Biomed Eng 2020;67(1):122–33.

[38] Beck JL, Katafygiotis LS. Updating models and their uncertainties. I: Bayesian statistical framework. J Eng Mech 1998;124:455–61.

[39] Karathanasopoulos N, Angelikopoulos P, Papadimitriou C, Koumoutsakos P. Bayesian identification of the tendon fascicle's structural composition using finite element models for helical geometries. Comput Methods Appl Mech Eng 2017;313: 744–58.

[40] Zeng C, Huang J, Xie J, Zhang Bo, Indraratna B. Prediction of mud pumping in railway track using in-service train data. Transp Geotech 2021;31:100651.

[41] Zeng C, Huang J, Wang H, Xie J, Huang S. Rail break prediction and cause analysis using imbalanced in-service train data. IEEE Trans Instrum Meas 2022;71:1–14.

[42] Mohammadi R, He Q, Ghofrani F, Pathak A, Aref A. Exploring the impact of foot-by-foot track geometry on the occurrence of rail defects. Transportation Research Part C: Emerging Technologies 2019;102:153–72.

[43] Australian Rail Track Corporation L. Performance Indicators 'track condition' 2015.

[44] Goodfellow I, Bengio Y, Courville A. Deep learning. MIT press; 2016.

[45] Feng S, Chen Z, Luo H, Wang S, Zhao Y, Liu L, et al. Tunnel boring machines (TBM) performance prediction: A case study using big data and deep learning. Tunn Undergr Space Technol 2021;110:103636.

[46] Vanhoucke V, Senior A, Mao MZ. Improving the speed of neural networks on CPUs. 2011.

[47] Steck H, Krishnapuram B, Dehing-Oberije C, Lambin P, Raykar VC. On ranking in survival analysis: Bounds on the concordance index. Advances in Neural Information Processing Systems. 2007;20.

[48] Ishwaran H, Kogalur UB, Blackstone EH, Lauer MS. Random survival forests. 2008.

[49] Li C, He Q, Wang P. Estimation of railway track longitudinal irregularity using vehicle response with information compression and Bayesian deep learning. Comput Aided Civ Inf Eng 2022;37(10):1260–76.