Wu, Shaomin (2021) *Relevance Vector Regression for Remaining Useful Life Prediction.* In: Mathematical Modeling in Physical Sciences, Social Sciences and Technology (icmm-21), 17-18 December, 2021, Organized by Department of Mathematics, Chaudhary Bansi Lal University, 17-18 Dec 2021, Bhiwani, India. (Unpublished)

Relevance Vector Regression for Remaining Useful Life Prediction

Prof. Shaomin Wu

Kent Business School
University of Kent

December 17, 2021

- This presentation is based on the paper: Wang, X., Jiang, B., Wu, S., Lu, N. and Ding, S., 2021. Multivariate Relevance Vector Regression based Degradation Modeling and Remaining Useful Life Prediction. *IEEE Transactions on Industrial Electronics*, doi: 10.1109/TIE.2021.3114724.
- I would like to thank my co-authors of the paper;
- I would also like to thank the organisers of *the International Conference in Mathematical Modeling in Physical Sciences, Social Sciences and Technology (icmm-21)* for their invitations.

Table of Contents

Table of Contents

Importance of RUL Prediction and our methods

- **Remaining Useful Life (RUL)** RUL is the length of an industrial item from its current time to the end of its useful life.
- **Importance of RUL estimation** It is needed in condition based maintenance, prognostics and health management.
- **Some methods for RUL estimation**
  - With lifetime data and other data, one can build a Cox regression model and then derive the probability distribution of the RUL,
  - One may also estimate the distribution of the first hitting time based on a degradation path, which will be adopted in this talk.

Importance of RUL Prediction and our methods

- **Remaining Useful Life (RUL)** RUL is the length of an industrial item from its current time to the end of its useful life.
- **Importance of RUL estimation** It is needed in condition based maintenance, prognostics and health management.
- **Some methods for RUL estimation**
  - With lifetime data and other data, one can build a Cox regression model and then derive the probability distribution of the RUL,
  - One may also estimate the distribution of the first hitting time based on a degradation path, which will be adopted in this talk.

Importance of RUL Prediction and our methods

- **Remaining Useful Life (RUL)** RUL is the length of an industrial item from its current time to the end of its useful life.
- **Importance of RUL estimation** It is needed in condition based maintenance, prognostics and health management.
- **Some methods for RUL estimation**
  - With lifetime data and other data, one can build a Cox regression model and then derive the probability distribution of the RUL,
  - One may also estimate the distribution of the first hitting time based on a degradation path, which will be adopted in this talk.

Importance of RUL Prediction and our methods

- **Remaining Useful Life (RUL)** RUL is the length of an industrial item from its current time to the end of its useful life.
- **Importance of RUL estimation** It is needed in condition based maintenance, prognostics and health management.
- **Some methods for RUL estimation**
  - With lifetime data and other data, one can build a Cox regression model and then derive the probability distribution of the RUL,
  - One may also estimate the distribution of the first hitting time based on a degradation path, which will be adopted in this talk.

Regression with OLS, Ridge, LASSO, ElasticNet

- **Ordinary least squares (OLS):** Given a set of data points, $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$, such that $\boldsymbol{x}_i (= (1, x_{i1}, \ldots, , x_{ip})^T) \in R^{n+1}$ is a feature vector for the ith case and $y_i \in R^1$ is a target output, one can build a linear regression model

$$y_n = \mathbf{w}^{\mathrm{T}} \mathbf{x} + \epsilon_n, ,$$

with the aim to minimise the error on the training datasets

$$\min \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i^T \boldsymbol{w}) \tag{1}$$

where $\boldsymbol{w} = (w_0, w_1, \ldots, w_p)^T$ is the weight vector.

- **Extensions:** Ridge, LASSO (least absolute shrinkage and selection operator), and ElasticNet are extensions of OLS, with an additional penalty on the weights that aims to maximise the generalisation.

Regression with OLS, Ridge, LASSO, ElasticNet

- **Ordinary least squares (OLS):** Given a set of data points, $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$, such that $\boldsymbol{x}_i (= (1, x_{i1}, \ldots, , x_{ip})^T) \in R^{n+1}$ is a feature vector for the ith case and $y_i \in R^1$ is a target output, one can build a linear regression model

$$y_n = \mathbf{w}^{\mathrm{T}}\mathbf{x} + \epsilon_n, ,$$

with the aim to minimise the error on the training datasets

$$\min \sum_{i=1}^{n} (y_i - \boldsymbol{x}_i^T \boldsymbol{w}) \tag{1}$$

where $\boldsymbol{w} = (w_0, w_1, \ldots, w_p)^T$ is the weight vector.

- **Extensions:** Ridge, LASSO (least absolute shrinkage and selection operator), and ElasticNet are extensions of OLS, with an additional penalty on the weights that aims to maximise the generalisation.

Support Vector Regression (SVR)

- **SVR** aims to minimize the weights,

$$\min \ \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} \tag{2}$$

$$|y_i - \boldsymbol{x}_i^T\boldsymbol{w}| \leq \epsilon \tag{3}$$

where $\boldsymbol{w}$ is the weight vector and ϵ is the error term.

- ϵ-**SVR** is given by

$$\min_{w,b,\xi,\xi^*} \quad \frac{1}{2}w^T w + C\left(\sum_{i=1}^{l}\xi_i + \sum_{i=1}^{l}\xi_i^*\right)$$

$$y_i - w^T\phi(x_i) - b \leq \epsilon + \xi_i,$$

$$w^T\phi(x_i) + b - y_i \leq \epsilon + \xi_i^*,$$

$$\xi_i, \xi_i^* \geq 0, i = 1, \ldots, l.$$

where C is the regularization parameter and balances the trade-off between the model complexity and empirical error, ξ_i and ξ_i^* are slack variables.

Support Vector Regression (SVR)

- **SVR** aims to minimize the weights,

$$\min \quad \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} \tag{2}$$

$$|y_i - \boldsymbol{x}_i^T\boldsymbol{w}| \leq \epsilon \tag{3}$$

  where $\boldsymbol{w}$ is the weight vector and ϵ is the error term.

- ϵ-**SVR** is given by

$$\begin{aligned}
\min_{w,b,\xi,\xi^*} \quad & \frac{1}{2}w^T w + C(\sum_{i=1}^{l}\xi_i + \sum_{i=1}^{l}\xi_i^*) \\
& y_i - w^T\phi(x_i) - b \leq \epsilon + \xi_i, \\
& w^T\phi(x_i) + b - y_i \leq \epsilon + \xi_i^*, \\
& \xi_i, \xi_i^* \geq 0, i = 1, \ldots, l.
\end{aligned}$$

  where C is the regularization parameter and balances the trade-off between the model complexity and empirical error, ξ_i and ξ_i^* are slack variables.

Limitations of SVR

- SVR has been used in many applications Nevertheless, they suffer some limitations:
  - **non-probabilistic:** SVR does not output probabilistic predictions;
  - C **and** ϵ**:** parameters of C and ϵ must be determined by cross-validation; and
  - **Mercer's condition:** the kernels must satisfy Mercer's condition. Relevance vector regression To overcome the above limitations, Tipping introduces a novel model: Relevance Vector Regression

Source: Tipping, M.E., 2001. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1(Jun), pp.211-244

Limitations of SVR

- SVR has been used in many applications Nevertheless, they suffer some limitations:
  - **non-probabilistic:** SVR does not output probabilistic predictions;
  - **C and ϵ:** parameters of C and ϵ must be determined by cross-validation; and
  - **Mercer's condition:** the kernels must satisfy Mercer's condition. Relevance vector regression To overcome the above limitations, Tipping introduces a novel model: Relevance Vector Regression

Source: Tipping, M.E., 2001. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1(Jun), pp.211-244

Limitations of SVR

- SVR has been used in many applications Nevertheless, they suffer some limitations:
  - **non-probabilistic:** SVR does not output probabilistic predictions;
  - **C and ϵ:** parameters of C and ϵ must be determined by cross-validation; and
  - **Mercer's condition:** the kernels must satisfy Mercer's condition. Relevance vector regression To overcome the above limitations, Tipping introduces a novel model: Relevance Vector Regression

Source: Tipping, M.E., 2001. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1(Jun), pp.211-244

Limitations of SVR

- SVR has been used in many applications Nevertheless, they suffer some limitations:
  - **non-probabilistic:** SVR does not output probabilistic predictions;
  - **C and ϵ:** parameters of C and ϵ must be determined by cross-validation; and
  - **Mercer's condition:** the kernels must satisfy Mercer's condition. Relevance vector regression To overcome the above limitations, Tipping introduces a novel model: Relevance Vector Regression

Source: Tipping, M.E., 2001. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1(Jun), pp.211-244

Relevance Vector Regression (RVR)

- Assume the relationship between $\{\boldsymbol{x}_n\}$ and $\{y_n\}_{n=1}^{N}$ is:

$$y_n = \mathbf{w}^{\mathrm{T}}\phi(\mathbf{x}) + \epsilon_n,$$

  where ϵ_n are i.i.d with the Gaussian distribution having mean-zero and variance σ^2.

- Due to the assumption of independence of the t_n, the likelihood of the complete data set can be written as

$$p(\mathbf{y}|\mathbf{w},\sigma^2) = (2\pi\sigma^2)^{-N/2}\exp\left\{-\frac{1}{2\sigma^2}\|\mathbf{y}-\mathbf{\Phi}\mathbf{w}\|^2\right\}, \quad (4)$$

  where $\mathbf{y} = (y_1 \dots y_N)^{\mathrm{T}}$, $\mathbf{w} = (w_0 \dots w_N)^{\mathrm{T}}$ and $\mathbf{\Phi}$ is the $N \times (N+1)$ 'design' matrix with $\mathbf{\Phi} = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_N)]^{\mathrm{T}}$, wherein $\phi(\mathbf{x}_n) = [1, K(\mathbf{x}_n,\mathbf{x}_1), K(\mathbf{x}_n,\mathbf{x}_2), \dots, K(\mathbf{x}_n,\mathbf{x}_N)]^{\mathrm{T}}$ and $K(.,.)$ is a kernel function.

RVR (cont'd)

- Tipping assumes a zero-mean Gaussian prior distribution over **w**:

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{i=0}^{N} \mathcal{N}(w_i|0, \alpha_i^{-1}), \tag{5}$$

  with $\boldsymbol{\alpha}$ a vector of $N+1$ *hyperparameters*.

- The suitable priors for α and α are Gamma distributions:

$$p(\boldsymbol{\alpha}) = \prod_{i=0}^{N} \text{Gamma}(\alpha_i|a, b),$$

$$p(\beta) = \text{Gamma}(\beta|c, d),$$

  with $\beta \equiv \sigma^{-2}$ and where $\text{Gamma}(\alpha|a, b) = \Gamma(a)^{-1} b^a \alpha^{a-1} e^{-b\alpha}$, with $\Gamma(a) = \int_0^\infty t^{a-1} e^{-t} dt$, the 'gamma function'.

RVR (cont'd)

- Tipping assumes a zero-mean Gaussian prior distribution over **w**:

$$p(\mathbf{w}|\boldsymbol{\alpha}) = \prod_{i=0}^{N} \mathcal{N}(w_i|0, \alpha_i^{-1}), \tag{5}$$

  with $\boldsymbol{\alpha}$ a vector of $N+1$ *hyperparameters*.

- The suitable priors for α and $\boldsymbol{\alpha}$ are Gamma distributions:

$$p(\boldsymbol{\alpha}) = \prod_{i=0}^{N} \text{Gamma}(\alpha_i|a, b),$$
$$p(\beta) = \text{Gamma}(\beta|c, d),$$

  with $\beta \equiv \sigma^{-2}$ and where $\text{Gamma}(\alpha|a, b) = \Gamma(a)^{-1} b^a \alpha^{a-1} e^{-b\alpha}$, with $\Gamma(a) = \int_0^\infty t^{a-1} e^{-t} dt$, the 'gamma function'.

Pros and Cons of RVR

- **Pros**
  - The relevance vector machine or RVM (Tipping, 2001) is a probabilistic regression model under the Bayesian framework;
  - compared to the SVR, the RVR results in a sparser model;
  - The kernels do not need to satisfy Mercer's condition;
- **Con** RVR only allows regression from multivariate inputs to a univariate output variable, but not to multiple variables. The operation of many engineering systems is influenced by multiple variables. For example, current and voltage are indispensable for electrical systems.

Pros and Cons of RVR

- **Pros**
  - The relevance vector machine or RVM (Tipping, 2001) is a probabilistic regression model under the Bayesian framework;
  - compared to the SVR, the RVR results in a sparser model;
  - The kernels do not need to satisfy Mercer's condition;
- **Con** RVR only allows regression from multivariate inputs to a univariate output variable, but not to multiple variables. The operation of many engineering systems is influenced by multiple variables. For example, current and voltage are indispensable for electrical systems.

Pros and Cons of RVR

- **Pros**
  - The relevance vector machine or RVM (Tipping, 2001) is a probabilistic regression model under the Bayesian framework;
  - compared to the SVR, the RVR results in a sparser model;
  - The kernels do not need to satisfy Mercer's condition;
- **Con** RVR only allows regression from multivariate inputs to a univariate output variable, but not to multiple variables. The operation of many engineering systems is influenced by multiple variables. For example, current and voltage are indispensable for electrical systems.

Pros and Cons of RVR

- **Pros**
  - The relevance vector machine or RVM (Tipping, 2001) is a probabilistic regression model under the Bayesian framework;
  - compared to the SVR, the RVR results in a sparser model;
  - The kernels do not need to satisfy Mercer's condition;
- **Con** RVR only allows regression from multivariate inputs to a univariate output variable, but not to multiple variables. The operation of many engineering systems is influenced by multiple variables. For example, current and voltage are indispensable for electrical systems.

Literature review and our methods

RVR has been used to predict the remaining useful life.

- **Literature review**
  - Some authors combine a set of univariate RVR models to output multiple variables, and
  - Some authors regard that the weight matrix is inducted by separating into a vector distribution;
- **Our methods:**
  - We propose a multivatiate RVR model (MRVR), in which the weight matrix is a matrix Gaussian distribution;
  - The hyperparameters of the MRVR model are estimated by Nesterov's Accelerated Gradient (NAG) method to obtain numerical solutions

Literature review and our methods

RVR has been used to predict the remaining useful life.

- **Literature review**
  - Some authors combine a set of univariate RVR models to output multiple variables, and
  - Some authors regard that the weight matrix is inducted by separating into a vector distribution;
- **Our methods:**
  - We propose a multivatiate RVR model (MRVR), in which the weight matrix is a matrix Gaussian distribution;
  - The hyperparameters of the MRVR model are estimated by Nesterov's Accelerated Gradient (NAG) method to obtain numerical solutions

Literature review and our methods

RVR has been used to predict the remaining useful life.

- **Literature review**
  - Some authors combine a set of univariate RVR models to output multiple variables, and
  - Some authors regard that the weight matrix is inducted by separating into a vector distribution;
- **Our methods:**
  - We propose a multivatiate RVR model (MRVR), in which the weight matrix is a matrix Gaussian distribution;
  - The hyperparameters of the MRVR model are estimated by Nesterov's Accelerated Gradient (NAG) method to obtain numerical solutions

Literature review and our methods

RVR has been used to predict the remaining useful life.

- **Literature review**
  - Some authors combine a set of univariate RVR models to output multiple variables, and
  - Some authors regard that the weight matrix is inducted by separating into a vector distribution;
- **Our methods:**
  - We propose a multivatiate RVR model (MRVR), in which the weight matrix is a matrix Gaussian distribution;
  - The hyperparameters of the MRVR model are estimated by Nesterov's Accelerated Gradient (NAG) method to obtain numerical solutions

Table of Contents

Multivariate RVR

- An MRVR is proposed as following

$$\boldsymbol{x}_{n+l} = \boldsymbol{W}\phi(\boldsymbol{x}_n) + \epsilon, \tag{6}$$

where

- $\boldsymbol{x}_{n+l} = [x_{1,n+l}, \cdots, x_{M,n+l}]^{\mathrm{T}} \in \mathbb{R}^M$ is the $l-$step forward prediction vector, and $1 < n + l \leq N$;
- $\phi(\boldsymbol{x}_n) = [\mathbf{1}, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_1), \cdots, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_N)] \in \mathbb{R}^{N+1}$ denotes a design vector, in which $\mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_j) \in \mathbb{R}$ is a kernel function,
- $\boldsymbol{W} \in \mathbb{R}^{M \times (N+1)}$ is a weight matrix of the design vector $\phi(\boldsymbol{x}_n) \triangleq \phi$, and
- ϵ is assumed to be a Gaussian distributed random error vector with the zero mean and a diagonal covariance matrix $\Sigma_0 = diag\{\sigma_1^2, \sigma_2^2, \cdots, \sigma_M^2\} \in \mathbb{R}^{M \times M}$, and $diag(\cdot)$ denotes a diagonal matrix.

Multivariate RVR

- An MRVR is proposed as following

$$\boldsymbol{x}_{n+l} = \boldsymbol{W}\phi(\boldsymbol{x}_n) + \boldsymbol{\epsilon}, \tag{6}$$

  where
  - $\boldsymbol{x}_{n+l} = [x_{1,n+l}, \cdots, x_{M,n+l}]^{\mathrm{T}} \in \mathbb{R}^M$ is the $l-$step forward prediction vector, and $1 < n + l \leq N$;
  - $\phi(\boldsymbol{x}_n) = [\mathbf{1}, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_1), \cdots, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_N)] \in \mathbb{R}^{N+1}$ denotes a design vector, in which $\mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_j) \in \mathbb{R}$ is a kernel function,
  - $\boldsymbol{W} \in \mathbb{R}^{M\times(N+1)}$ is a weight matrix of the design vector $\phi(\boldsymbol{x}_n) \triangleq \phi$, and
  - ϵ is assumed to be a Gaussian distributed random error vector with the zero mean and a diagonal covariance matrix $\Sigma_0 = diag\{\sigma_1^2, \sigma_2^2, \cdots, \sigma_M^2\} \in \mathbb{R}^{M\times M}$, and $diag(\cdot)$ denotes a diagonal matrix.

Multivariate RVR

- An MRVR is proposed as following

$$\boldsymbol{x}_{n+l} = \boldsymbol{W}\phi(\boldsymbol{x}_n) + \boldsymbol{\epsilon}, \qquad (6)$$

where

- $\boldsymbol{x}_{n+l} = [x_{1,n+l}, \cdots, x_{M,n+l}]^{\mathrm{T}} \in \mathbb{R}^M$ is the $l-$step forward prediction vector, and $1 < n + l \leq N$;
- $\phi(\boldsymbol{x}_n) = [\mathbf{1}, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_1), \cdots, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_N)] \in \mathbb{R}^{N+1}$ denotes a design vector, in which $\mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_j) \in \mathbb{R}$ is a kernel function,
- $\boldsymbol{W} \in \mathbb{R}^{M\times(N+1)}$ is a weight matrix of the design vector $\phi(\boldsymbol{x}_n) \triangleq \phi$, and
- ϵ is assumed to be a Gaussian distributed random error vector with the zero mean and a diagonal covariance matrix $\Sigma_0 = diag\{\sigma_1^2, \sigma_2^2, \cdots, \sigma_M^2\} \in \mathbb{R}^{M\times M}$, and $diag(\cdot)$ denotes a diagonal matrix.

Multivariate RVR

- An MRVR is proposed as following

$$\boldsymbol{x}_{n+l} = \boldsymbol{W}\phi(\boldsymbol{x}_n) + \boldsymbol{\epsilon}, \quad (6)$$

where

- $\boldsymbol{x}_{n+l} = [x_{1,n+l}, \cdots, x_{M,n+l}]^{\mathrm{T}} \in \mathbb{R}^M$ is the $l-$step forward prediction vector, and $1 < n + l \leq N$;
- $\phi(\boldsymbol{x}_n) = [\mathbf{1}, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_1), \cdots, \mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_N)] \in \mathbb{R}^{N+1}$ denotes a design vector, in which $\mathcal{K}(\boldsymbol{x}_n, \boldsymbol{x}_j) \in \mathbb{R}$ is a kernel function,
- $\boldsymbol{W} \in \mathbb{R}^{M\times(N+1)}$ is a weight matrix of the design vector $\phi(\boldsymbol{x}_n) \triangleq \phi$, and
- $\boldsymbol{\epsilon}$ is assumed to be a Gaussian distributed random error vector with the zero mean and a diagonal covariance matrix $\boldsymbol{\Sigma}_0 = diag\{\sigma_1^2, \sigma_2^2, \cdots, \sigma_M^2\} \in \mathbb{R}^{M\times M}$, and $diag(\cdot)$ denotes a diagonal matrix.

PDF of x_{n+1}

Probability Density Function (PDF) of $\boldsymbol{x}_{n+l}$ conditioned on $\boldsymbol{W}$ and Σ_0 can be written by

$$p(\boldsymbol{x}_{n+l}|\boldsymbol{W}, \Sigma_0) = (2\pi)^{-\frac{M}{2}}|\Sigma_0|^{-\frac{1}{2}} \times \exp\left(-\frac{1}{2}(\boldsymbol{x}_{n+l} - \boldsymbol{W}\phi)^{\mathrm{T}}\Sigma_0^{-1}(\boldsymbol{x}_{n+l} - \boldsymbol{W}\phi)\right), \quad (7)$$

here $|\cdot|$ is the determinant of a square matrix.

Prior and Posterior distributions of W

- **Prior distributions of W** To avoid the over-fitting problem of model (6), a prior matrix Gaussian distribution is assigned on the $M \times (N+1)$ dimension weight matrix $\boldsymbol{W}$, which is denoted as $\boldsymbol{W} \sim \mathcal{MN}_{M,N+1}(\mathbf{0}, \boldsymbol{\Psi}, \boldsymbol{\Gamma})$, which gives

$$p(\boldsymbol{W}|\boldsymbol{\Psi},\boldsymbol{\Gamma}) = (2\pi)^{-\frac{M(N+1)}{2}}|\boldsymbol{\Psi}|^{-\frac{N+1}{2}}|\boldsymbol{\Gamma}|^{-\frac{M}{2}} \times \operatorname{etr}\left(-\frac{1}{2}\boldsymbol{\Gamma}^{-1}\boldsymbol{W}^{\mathrm{T}}\boldsymbol{\Psi}^{-1}\boldsymbol{W}\right), \tag{8}$$

  where $\operatorname{etr}(\cdot)$ is the exponential of trace of a function of the trace of the matrix.

- **Posterior distributions of W** The posterior distribution of weight matrix $\boldsymbol{W}$ is matrix Gaussian, and its PDF is formulated in the following form.

$$p(\boldsymbol{W}|\boldsymbol{x}_{n+l},\Psi,\Gamma,\Sigma_0) = (2\pi)^{-\frac{M(N+1)}{2}}|\tilde{\Psi}|^{-\frac{N+1}{2}} \times|\tilde{\Gamma}|^{-\frac{M}{2}} \operatorname{etr}\left(-\frac{1}{2}\tilde{\Gamma}^{-1}(\boldsymbol{W}-\tilde{\mu})^{\mathrm{T}}\tilde{\Psi}^{-1}(\boldsymbol{W}-\tilde{\mu})\right). \tag{9}$$

Prior and Posterior distributions of $\boldsymbol{W}$

- **Prior distributions of $\boldsymbol{W}$** To avoid the over-fitting problem of model (6), a prior matrix Gaussian distribution is assigned on the $M \times (N+1)$ dimension weight matrix $\boldsymbol{W}$, which is denoted as $\boldsymbol{W} \sim \mathcal{MN}_{M,N+1}(\mathbf{0}, \boldsymbol{\Psi}, \boldsymbol{\Gamma})$, which gives

$$p(\boldsymbol{W}|\boldsymbol{\Psi}, \boldsymbol{\Gamma}) = (2\pi)^{-\frac{M(N+1)}{2}} |\boldsymbol{\Psi}|^{-\frac{N+1}{2}} |\boldsymbol{\Gamma}|^{-\frac{M}{2}} \times \operatorname{etr}\left(-\frac{1}{2}\boldsymbol{\Gamma}^{-1}\boldsymbol{W}^{\mathrm{T}}\boldsymbol{\Psi}^{-1}\boldsymbol{W}\right), \tag{8}$$

  where $\operatorname{etr}(\cdot)$ is the exponential of trace of a function of the trace of the matrix.
- **Posterior distributions of $\boldsymbol{W}$** The posterior distribution of weight matrix $\boldsymbol{W}$ is matrix Gaussian, and its PDF is formulated in the following form.

$$p(\boldsymbol{W}|\boldsymbol{x}_{n+l}, \boldsymbol{\Psi}, \boldsymbol{\Gamma}, \boldsymbol{\Sigma}_0) = (2\pi)^{-\frac{M(N+1)}{2}} |\tilde{\boldsymbol{\Psi}}|^{-\frac{N+1}{2}} \times |\tilde{\boldsymbol{\Gamma}}|^{-\frac{M}{2}} \operatorname{etr}\left(-\frac{1}{2}\tilde{\boldsymbol{\Gamma}}^{-1}(\boldsymbol{W} - \tilde{\boldsymbol{\mu}})^{\mathrm{T}}\tilde{\boldsymbol{\Psi}}^{-1}(\boldsymbol{W} - \tilde{\boldsymbol{\mu}})\right). \tag{9}$$

Parameter Estimation

- **Prediction distribution** The distribution of the predicted $\mathbf{x}_{k+l}$, based on the former prediction $\mathbf{x}_{n+l}$, can be obtained by

$$p\left(\mathbf{x}_{k+l} | \mathbf{x}_{n+l}\right) = \iiiint p\left(\mathbf{x}_{k+l} | \boldsymbol{W}, \boldsymbol{\Sigma}_0\right) p\left(\boldsymbol{W} | \mathbf{x}_{n+l}, \boldsymbol{\Psi}, \boldsymbol{\Gamma}, \boldsymbol{\Sigma}_0\right) \times p\left(\boldsymbol{\Psi}, \boldsymbol{\Gamma}, \boldsymbol{\Sigma}_0 | \mathbf{x}_{n+l}\right) \mathrm{d}\boldsymbol{W} \mathrm{d}\boldsymbol{\Psi} \mathrm{d}\boldsymbol{\Gamma} \mathrm{d}\boldsymbol{\Sigma}_0. \tag{10}$$

- **Marginal likelihood function** The marginal likelihood function $p\left(x_{n+l} | \Psi, \Gamma, \Sigma_0\right)$ can be obtained by integrating over the weight parameters W as

$$p\left(x_{n+l} | \Psi, \Gamma, \Sigma_0\right) = \int p\left(x_{n+l} | W, \Sigma_0\right) p(W | \Psi, \Gamma) \mathrm{d}W = \int p\left(x_{n+l} | \mathrm{vec}\left(W^{\mathrm{T}}\right), \Sigma_0\right) p(\mathrm{vec}\left(W^{\mathrm{T}}\right) | \Psi, \Gamma) \mathrm{d}\,\mathrm{vec}\left(W^{\mathrm{T}}\right). \tag{11}$$

Parameter Estimation

- **Prediction distribution** The distribution of the predicted $\boldsymbol{x}_{k+l}$, based on the former prediction $\boldsymbol{x}_{n+l}$, can be obtained by

$$p\left(\boldsymbol{x}_{k+l}|\boldsymbol{x}_{n+l}\right) = \iiiint p\left(\boldsymbol{x}_{k+l}|\boldsymbol{W}, \boldsymbol{\Sigma}_0\right) p\left(\boldsymbol{W}|\boldsymbol{x}_{n+l}, \boldsymbol{\Psi}, \boldsymbol{\Gamma}, \boldsymbol{\Sigma}_0\right) \times p\left(\boldsymbol{\Psi}, \boldsymbol{\Gamma}, \boldsymbol{\Sigma}_0|\boldsymbol{x}_{n+l}\right) \mathrm{d}\boldsymbol{W} \mathrm{d}\boldsymbol{\Psi} \mathrm{d}\boldsymbol{\Gamma} \mathrm{d}\boldsymbol{\Sigma}_0. \quad (10)$$

- **Marginal likelihood function** The marginal likelihood function $p\left(\boldsymbol{x}_{n+l}|\boldsymbol{\Psi}, \boldsymbol{\Gamma}, \boldsymbol{\Sigma}_0\right)$ can be obtained by integrating over the weight parameters $\boldsymbol{W}$ as

$$p\left(\boldsymbol{x}_{n+l}|\boldsymbol{\Psi}, \boldsymbol{\Gamma}, \boldsymbol{\Sigma}_0\right) = \int p\left(\boldsymbol{x}_{n+l}|\boldsymbol{W}, \boldsymbol{\Sigma}_0\right) p(\boldsymbol{W}|\boldsymbol{\Psi}, \boldsymbol{\Gamma}) \mathrm{d}\boldsymbol{W} = \int p\left(\boldsymbol{x}_{n+l}|\operatorname{vec}\left(\boldsymbol{W}^{\mathrm{T}}\right), \boldsymbol{\Sigma}_0\right) p(\operatorname{vec}\left(\boldsymbol{W}^{\mathrm{T}}\right)|\boldsymbol{\Psi}, \boldsymbol{\Gamma}) \mathrm{d}\operatorname{vec}\left(\boldsymbol{W}^{\mathrm{T}}\right). \quad (11)$$

Parameter Estimation (cont'd)

- The negative log of the marginal likelihood is acquired

$$\mathcal{L}\left(\boldsymbol{x}_{n+l}|\boldsymbol{\Psi},\boldsymbol{\Gamma},\boldsymbol{\Sigma}_0\right) = \frac{1}{2}\ln|\boldsymbol{\Sigma}_0| + \frac{N+1}{2}\ln|\boldsymbol{\Psi}| + \frac{M}{2}\ln|\boldsymbol{\Gamma}| + \frac{1}{2}\ln|\boldsymbol{A}| + E(\boldsymbol{\mu}) + \frac{M}{2}\ln(2\pi), \tag{12}$$

- set $\frac{\partial \mathcal{L}}{\partial \Psi_i} = 0$, $\frac{\partial \mathcal{L}}{\partial \Gamma_j} = 0$, and $\frac{\partial \mathcal{L}}{\partial \sigma_i^2} = 0$, it is difficult to obtain explicit solutions of the hyperparameters Ψ_i, Γ_j and σ_i^2.
- The NAG (Nesterov's Accelerated Gradient) method is used to obtain numerical solutions of the hyperparameters

Parameter Estimation (cont'd)

- The negative log of the marginal likelihood is acquired

$$\mathcal{L}\left(\boldsymbol{x}_{n+l}|\boldsymbol{\Psi},\boldsymbol{\Gamma},\boldsymbol{\Sigma}_0\right) = \frac{1}{2}\ln|\boldsymbol{\Sigma}_0| + \frac{N+1}{2}\ln|\boldsymbol{\Psi}| + \frac{M}{2}\ln|\boldsymbol{\Gamma}| + \frac{1}{2}\ln|\boldsymbol{A}| + E(\boldsymbol{\mu}) + \frac{M}{2}\ln(2\pi), \tag{12}$$

- set $\frac{\partial\mathcal{L}}{\partial\Psi_i} = 0$, $\frac{\partial\mathcal{L}}{\partial\Gamma_j} = 0$, and $\frac{\partial\mathcal{L}}{\partial\sigma_i^2} = 0$, it is difficult to obtain explicit solutions of the hyperparameters Ψ_i, Γ_j and σ_i^2.
- The NAG (Nesterov's Accelerated Gradient) method is used to obtain numerical solutions of the hyperparameters

Parameter Estimation (cont'd)

- The negative log of the marginal likelihood is acquired

$$\mathcal{L}\left(\boldsymbol{x}_{n+l}|\boldsymbol{\Psi},\boldsymbol{\Gamma},\boldsymbol{\Sigma}_0\right) = \frac{1}{2}\ln|\boldsymbol{\Sigma}_0| + \frac{N+1}{2}\ln|\boldsymbol{\Psi}| + \frac{M}{2}\ln|\boldsymbol{\Gamma}| + \frac{1}{2}\ln|\boldsymbol{A}| + E(\boldsymbol{\mu}) + \frac{M}{2}\ln(2\pi), \tag{12}$$

- set $\frac{\partial\mathcal{L}}{\partial\Psi_i} = 0$, $\frac{\partial\mathcal{L}}{\partial\Gamma_j} = 0$, and $\frac{\partial\mathcal{L}}{\partial\sigma_i^2} = 0$, it is difficult to obtain explicit solutions of the hyperparameters Ψ_i, Γ_j and σ_i^2.
- The NAG (Nesterov's Accelerated Gradient) method is used to obtain numerical solutions of the hyperparameters

First hitting time

- The PDF of the degradation prediction takes the form

$$p\left(\boldsymbol{x}_{k+l}|\boldsymbol{x}_{n+l}\right) = \mathcal{N}\left(\boldsymbol{x}_{k+l}|\boldsymbol{\Sigma}_0(\boldsymbol{\Sigma}_0^{-1} \otimes \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_k))\boldsymbol{\Lambda}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}, \boldsymbol{\Sigma}_k\right), \tag{13}$$

- Given the observed measurements $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_k$, the RUL of each variable at time t_k is defined by

$$L_{ik} \triangleq L_i(t_k) = \inf\{t_l : x_i(t_k + t_l) \in \mathcal{B}_i|\boldsymbol{x}_{1:k}\}, \tag{14}$$

where $\inf\{\cdot\}$ denotes the infimum of a discrete; t_l represents the time length of the multi-step prediction; $x_i(t_k + t_l)$ is the degradation path at time $t_k + t_l$, $\boldsymbol{x}_{1:k}$ denotes the historical measurements from t_1 to t_k; and $\mathcal{B}_i$ refers to a boundary set, containing a boundary, barrier, or failure threshold.

First hitting time

- The PDF of the degradation prediction takes the form

$$p\left(\boldsymbol{x}_{k+l}|\boldsymbol{x}_{n+l}\right) = \mathcal{N}\left(\boldsymbol{x}_{k+l}|\boldsymbol{\Sigma}_0(\boldsymbol{\Sigma}_0^{-1} \otimes \boldsymbol{\phi}^{\mathrm{T}}(\boldsymbol{x}_k))\boldsymbol{\Lambda}\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}, \boldsymbol{\Sigma}_k\right), \tag{13}$$

- Given the observed measurements $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_k$, the RUL of each variable at time t_k is defined by

$$L_{ik} \triangleq L_i(t_k) = \inf\{t_l : x_i(t_k + t_l) \in \mathcal{B}_i | \boldsymbol{x}_{1:k}\}, \tag{14}$$

where $\inf\{\cdot\}$ denotes the infimum of a discrete; t_l represents the time length of the multi-step prediction; $x_i(t_k + t_l)$ is the degradation path at time $t_k + t_l$, $\boldsymbol{x}_{1:k}$ denotes the historical measurements from t_1 to t_k; and $\mathcal{B}_i$ refers to a boundary set, containing a boundary, barrier, or failure threshold.

RUL prediction

- the mean of the RUL is obtained by

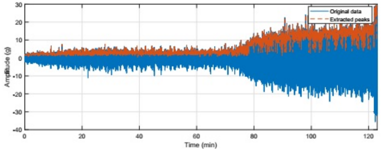
$$E_i(t_k) = \sum_{L_{ik}=0}^{+\infty} L_{ik} \cdot p_i(L_{ik}), \tag{15}$$

  where $p_i(L_{ik}) = \frac{\phi(g_{i,k+l})\Delta g_{i,k+l}}{1-\Phi(g_{i,k})}$, $\phi(\cdot)$ and $\Phi(\cdot)$ are the PDF and cumulative distribution function of a standard normal random variable, respectively;
- the confidence interval for the prediction can also be obtained.

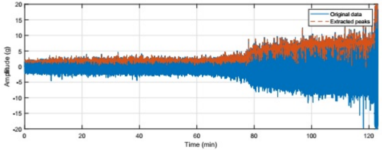
Table of Contents

Case study: Data

- A bearing dataset is used to demonstrate our proposed approach. Two accelerometers are placed on the bearings and positioned at 90° to each other, i.e., one is placed on the vertical axis and the other one is placed on the horizontal axis.
- Data from 78 minutes onwards are used for modelling.
- The operation is stopped when the amplitudes of the horizontal and vertical vibration signals are higher than 25g and 15g, respectively.



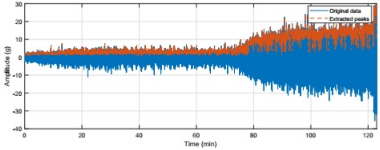
(a) The horizontal vibration signals.



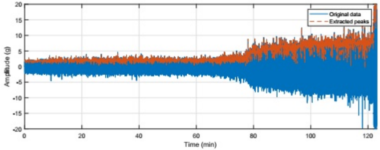
(b) The vertical vibration signals.

Case study: Data

- A bearing dataset is used to demonstrate our proposed approach. Two accelerometers are placed on the bearings and positioned at 90° to each other, i.e., one is placed on the vertical axis and the other one is placed on the horizontal axis.
- Data from 78 minutes onwards are used for modelling.
- The operation is stopped when the amplitudes of the horizontal and vertical vibration signals are higher than 25g and 15g, respectively.



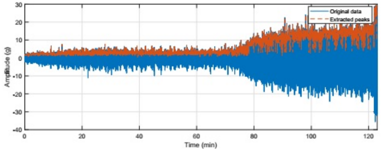
(a) The horizontal vibration signals.



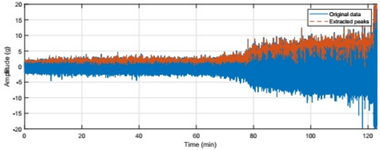
(b) The vertical vibration signals.

Case study: Data

- A bearing dataset is used to demonstrate our proposed approach. Two accelerometers are placed on the bearings and positioned at 90° to each other, i.e., one is placed on the vertical axis and the other one is placed on the horizontal axis.
- Data from 78 minutes onwards are used for modelling.
- The operation is stopped when the amplitudes of the horizontal and vertical vibration signals are higher than $25g$ and $15g$, respectively.



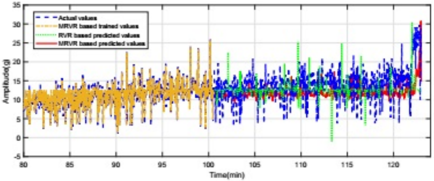
(a) The horizontal vibration signals.



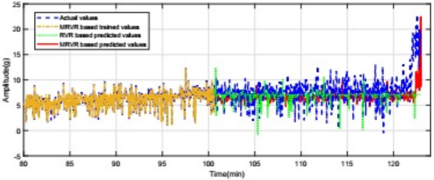
(b) The vertical vibration signals.

Case study: Degradation Path Prediction

- The peaks during $80 - 100min$ are used as inputs and those during $80.6 - 100.6min$ are used as outputs
- For the sake of comparison, both MRVR and RVR are built on the data
- the predicted degradation path based on the RVR cannot follow the actual vertical amplitude so well as that based on the MRVR.
- Observation: The MRVR outperforms the RVR.



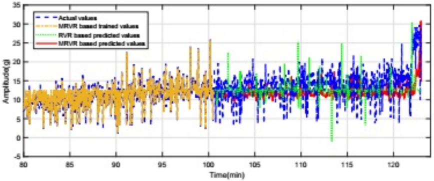
(a) The predicted amplitudes of the horizontal signal based on the RVR and MRVR.



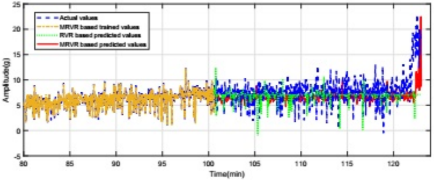
(b) The predicted amplitudes of the vertical signal based on the RVR and MRVR.

Case study: Degradation Path Prediction

- The peaks during $80 - 100min$ are used as inputs and those during $80.6 - 100.6min$ are used as outputs
- For the sake of comparison, both MRVR and RVR are built on the data
- the predicted degradation path based on the RVR cannot follow the actual vertical amplitude so well as that based on the MRVR.
- Observation: The MRVR outperforms the RVR.



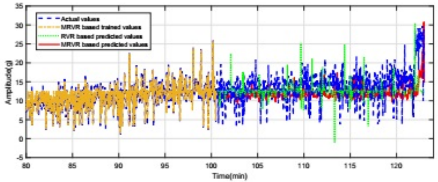
(a) The predicted amplitudes of the horizontal signal based on the RVR and MRVR.



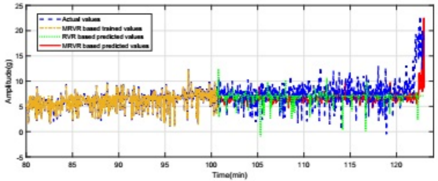
(b) The predicted amplitudes of the vertical signal based on the RVR and MRVR.

Case study: Degradation Path Prediction

- The peaks during $80 - 100min$ are used as inputs and those during $80.6 - 100.6min$ are used as outputs
- For the sake of comparison, both MRVR and RVR are built on the data
- the predicted degradation path based on the RVR cannot follow the actual vertical amplitude so well as that based on the MRVR.
- Observation: The MRVR outperforms the RVR.



(a) The predicted amplitudes of the horizontal signal based on the RVR and MRVR.



(b) The predicted amplitudes of the vertical signal based on the RVR and MRVR.

RUL prediction and Performance metrics

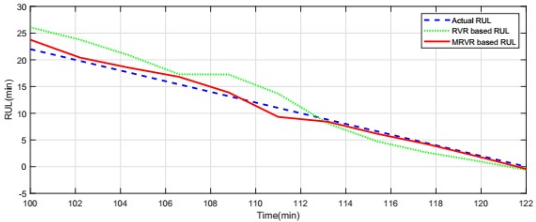


Figure: RUL Comparison

Performance metric	MRVR	RVR
MAE(min)	0.7960	2.3594
NRMSE(%)	8.7793	24.3193

Figure: Performance comparison

MAE=mean absolute error; NRMSE=Normalized Root Mean Relative Error

Conclusions

The MRVR outperforms!

Thank you all!

Questions?