2020.10.8 SICHEN LI

FORECASTING REMAINING USEFUL LIFE: INTERPRETABLE DEEP LEARNING APPROACH VIA VARIATIONAL BAYESIAN INFERENCES

GOAL

- Predicting remaining useful life (RUL) for machinery -> preemptive maintenance, prevent failure
- Forecasting via machine learning with “interpretability”
  - the decision logic of the model itself is transparent
  - != post-hoc explainability
- Two types of models
  - Probabilistic lifetime models: easy to interpret, not machine-specific
  - Data-driven ML: black-box, but good predicting power
  - Combine them

PROPOSED STRUCTURED-EFFECT NETWORK

- Decompose the RUL prediction into
  - population-wide baseline: A non-parametric, general lifetime common across all machines
  - machine-specific heterogeneity, including
    - A linear combination of sensor measurement
    - A recurrent component that incorporates historic data

$$SENN_\Theta(t; X_t, \ldots, X_1) = \underbrace{\lambda(t)}_{\text{Non-parametric component with explicit lifetime model}} + \underbrace{\beta^T X_t}_{\text{Linear component with current condition}} + \underbrace{RNN_\Theta(X_t, \ldots, X_1, t)}_{\text{Recurrent component with deep neural network}}$$

explicit probabilistic lifetime model

Parameter: a, b

1. COMMON BASELINE: PROBABILISTIC LIFETIME MODELS

- Use past lifetimes of all machines to fit a pre-defined pdf form
- The Weibull distribution, and the log-normal distribution over total lifetime Z: $P_{\text{Weibull}}(Z; a, b) = \frac{a}{b}\left(\frac{Z}{a}\right)^{b-1} e - (\frac{Z}{a})^{b}$ and

$$P_{\text{log-normal}}(Z; a, b) = \begin{cases} \frac{1}{\sqrt{2\pi}bZ} e - \frac{(log(Z) - a)^2}{2b^2}, & Z > 0, \end{cases}$$

- Conditional expectation, given that the machine had already run for time t

$$\lambda(t) = \mathbb{E}_{Z \sim \text{Weibull}(a,b)}[Z|Z > t] - t \quad \text{and} \quad \lambda(t) = \mathbb{E}_{Z \sim \text{log-normal}(a,b)}[Z|Z > t] - t$$

- Parameter: a, b

2. LINEAR COMPONENT

- input current sensor data: $\beta^T X_t$, or aggregation function over past sensor data: $\beta^T \phi(X_1, \ldots, X_t)$
- Parameter: beta

Aggregation function	Formula	Interpretation
Max	$\max(X_1,\ldots,X_t)$	Extrema
Min	$\min(X_1,\ldots,X_t)$	Extrema
Mean	$\mu = \frac{1}{t}\sum_{i=1}^{t} X_i$	Average sensor measurement
Range	$\max - \min$	Variability
Sum	$\sum_{i=1}^{t} X_i$	Total signal
Energy	$\sum_{i=1}^{t} X_i^2$	Total signal with focus on peaks
Standard deviation	$\sigma = \sqrt{\frac{1}{t}\sum_{i=1}^{t}(X_i - \mu)^2}$	Variability
Skewness	$\frac{1}{t}\sum_{i=1}^{t}\left(\frac{X_i - \mu}{\sigma}\right)^3$	Symmetry of deviation
Kurtosis	$\frac{1}{t}\sum_{i=1}^{t}\left(\frac{X_i - \mu}{\sigma}\right)^4$	Infrequent extreme deviations
Peak-to-peak	$\frac{1}{n_1}\sum_{i=1}^{n_1} \text{loc max} + \frac{1}{n_2}\sum_{i=1}^{n_2} \text{loc min}$	Bandwith
Root mean square	$\sqrt{\frac{1}{t}\sum_{i=1}^{t} X_i^2}$	Total load focus on peaks
Entropy	$-\sum_{i=1}^{t} P(X_i)\log P(X_i)$	Information signal
Arithmetic mean of power spectral density	$20\log_{10}\frac{\frac{1}{t}\sum_{i=1}^{t}\text{fft}(X_i)}{10^{-5}}$	Frequency of oscillations
Line integral	$\sum_{i=1}^{t-1}\lvert X_{i+1} - X_i\rvert$	Path length
Kalman filter	$Y_t - b - \sum_{i=1}^{p} a_i X_{t-i}$	Unexpected deviation

3. RNN

Hidden state h1

$$RNN_\Theta = f_{\mathrm{NN}}([X_t, f_{\mathrm{NN}}([X_{t-1}, \ldots f_{\mathrm{NN}}([X_1, 0_n])])]).$$

- RNN iterates over the sequence while updating its hidden state ht, which summarizes the already-seen sequence
- The neural network can absorb the variance that cannot be explained by the other components
- Parameter: Theta

MODEL ESTIMATION THROUGH BAYESIAN INFERENCE

- Determine the optimised combined set of unknown parameters $\theta^* = \{a, b, \beta, \Theta\}$ by maximizing the overall likelihood:

$$\theta^* = \operatorname{argmax}_\theta \; P(\theta \mid X) = \frac{P(X \mid \theta)\, P(\theta)}{P(X)} = \frac{P(X \mid \theta)\, P(\theta)}{\int P(X \mid \theta)\, P(\theta)\, \mathrm{d}\theta}.$$

- all parameters with a pre-defined prior distribution
  - Non-parametric component $a \sim \mathcal{N}(a_{empirical,1}), b \sim \mathcal{N}(b_{empirical,1})$
  - Linear component: normal $\beta_i \sim \mathcal{N}(0,10)$; Laplace prior for feature selection
  - Recurrent component: 2-layer LSTM, 100 and 50 neurons; All weights ~ Gaussian prior with standard deviation 1

VARIATIONAL BAYES METHOD

- approximates the true posterior via a variational distribution $Q_\lambda(\theta) \approx P(\theta \mid X)$
- -> Find the optimal lambda*, along with the corresponding distribution Q* that is closest
- Derive a variational lower bound $ELBO(\lambda)$ and do gradient descent to optimise the three components simultaneously

$$ELBO(\lambda) =: \mathbb{E}_{Q_\lambda}[\log P(X, \theta)] - \mathbb{E}_{Q_\lambda}[\log Q_\lambda(\theta)]$$

- Choose λ by $\nabla ELBO(\lambda)$

DATASET

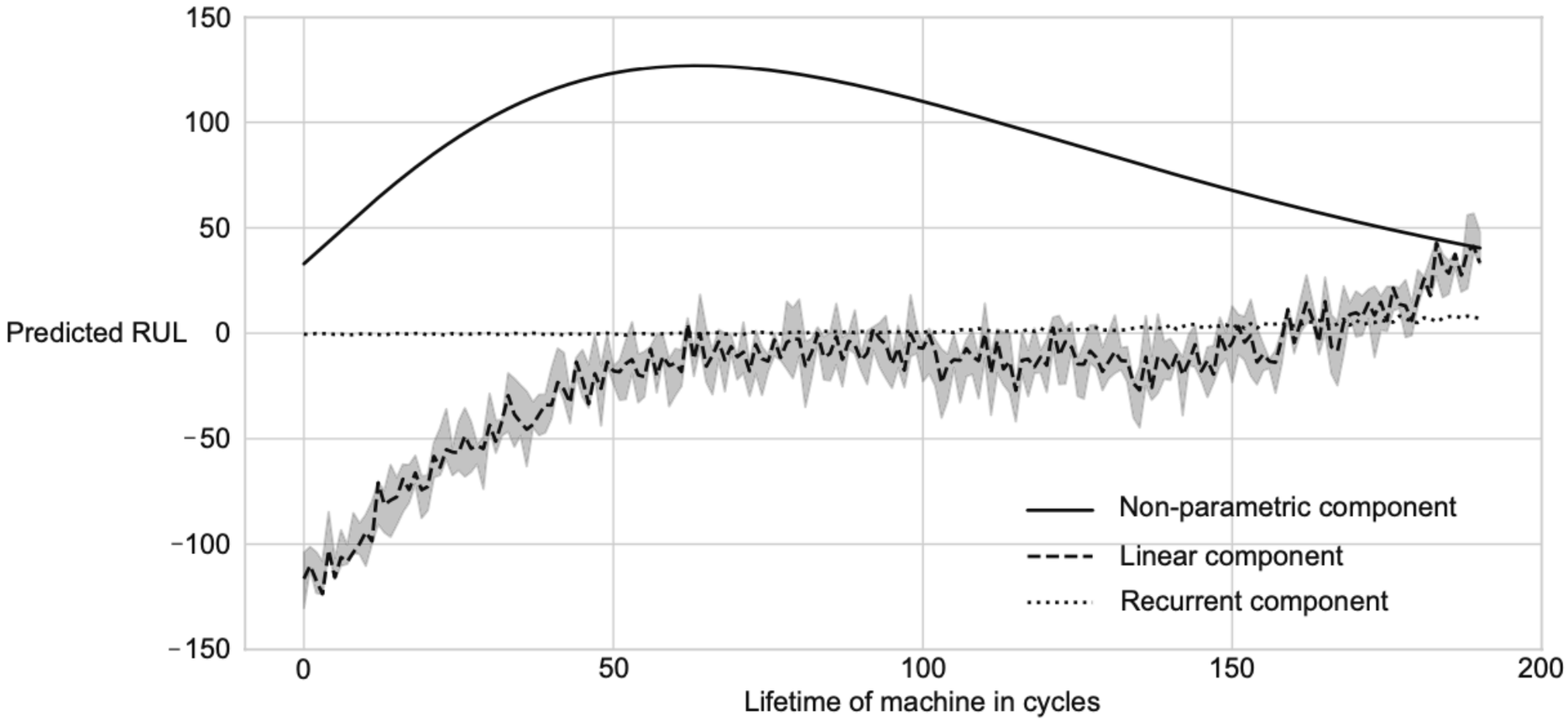
- Turbofan Engine Degradation Simulation dataset
- Predict the RUL (measured in cycles) based on sensor data from 200 aircraft engines -> 100 training, 100 testing
- 21 sensors

FORECAST RESULT

- Feature-engineering helps
- Traditional ML < Structured-effect network < RNN

Method		MAE
BASELINES WITHOUT SENSOR DATA		
Empirical RUL		45.060
Conditional expectation (Weibull)		27.794
Conditional expectation (log-normal)		**27.409**
TRADITIONAL MACHINE LEARNING		
Ridge regression		19.193
Ridge regression (with feature engineering)		18.382
Lasso		19.229
Lasso (with feature engineering)		18.853
Elastic net		19.229
Elastic net (with feature engineering)		18.245
Random forest		17.884
Random forest (with feature engineering)		**17.793**
SVR		18.109
SVR (with feature engineering)		21.932
RECURRENT NEURAL NETWORKS		
LSTM		**11.188**
STRUCTURED-EFFECT NEURAL NETWORKS		
Distribution	*Linear component*	
Weibull	None	15.862
Weibull	Regularized	17.433
Weibull	Feature engineering	13.392
Weibull	Regularized feature engineering	14.989
log-normal	None	15.061
log-normal	Regularized	16.319
log-normal	**Feature engineering**	**13.267**
log-normal	Regularized feature engineering	14.545

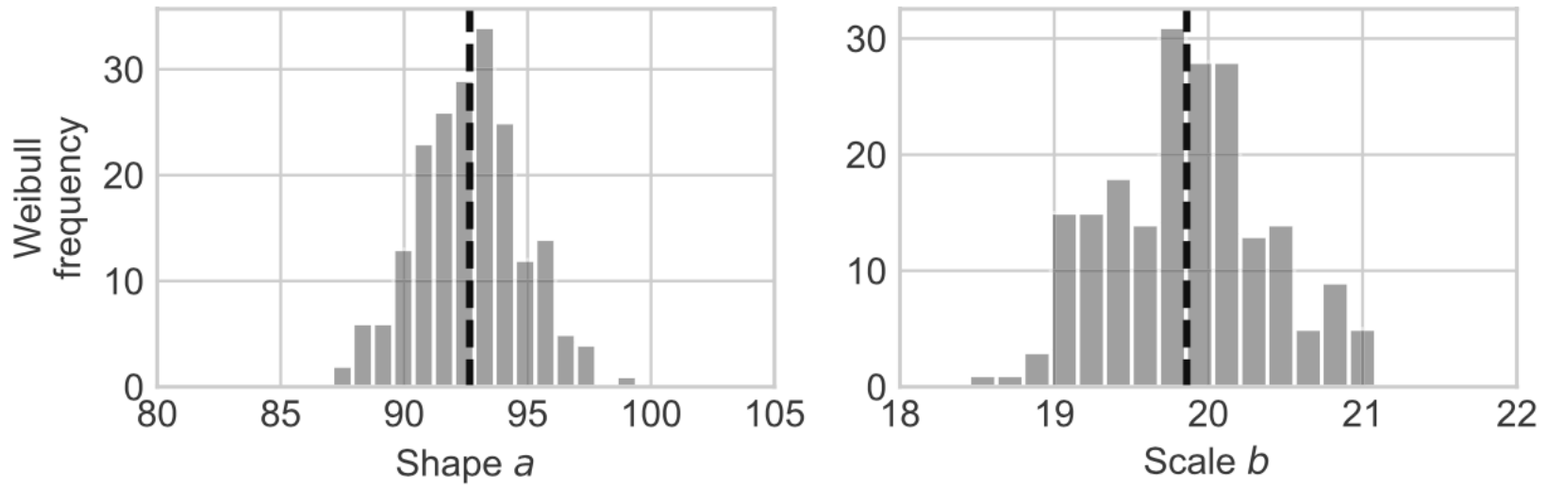
FORECAST DECOMPOSITION



- The distribution-based lifetime component contributes a considerable portion <- overall nature of the RUL (0.175 of variance)
- The sensor measurements introduce a within-engine and within-time variability (0.408 of variance)
- The recurrent neural network introduces a non-linear black-box component: very small, maybe due to enough predicting power of the current measurement X_t (0.064 of variance)

POSTERIOR DISTRIBUTION

Distribution parameters a and b



Linear components corresponding to 21 sensors

Sensor	Mean estimate	Standard deviation	Standardized coefficient
X_9	−33.169	0.498	−16.506
X_{12}	49.721	0.250	12.440
X_{21}	44.932	0.258	11.601
X_7	48.154	0.230	11.073
X_{11}	−24.622	0.357	−8.796
X_{20}	42.850	0.184	7.880
X_{14}	−22.540	0.317	−7.155
X_4	−16.183	0.275	−4.447
X_{15}	−13.934	0.297	−4.145
X_2	−12.446	0.316	−3.931
X_6	19.678	0.159	3.126
X_3	−7.851	0.318	−2.494
X_{17}	−9.951	0.208	−2.066
X_8	−4.121	0.367	−1.511
X_{16}	−0.273	2.105	−0.574
X_{13}	−1.424	0.348	−0.495
X_{19}	0.234	2.028	0.474
X_{10}	0.126	1.900	0.239
X_{18}	−0.095	1.936	−0.184
X_1	−0.070	1.937	−0.135
X_5	0.001	1.993	0.002

DERIVATION OF OPTIMISATION PROCEDURE

Appendix A. Derivation of ELBO for structured-effect neural network

Our suggested approach draws upon variational Bayesian methods and approximates the true posterior via a variational distribution $Q_\lambda(\theta) \approx P(\theta \mid X)$. Here $Q_\lambda(\theta)$ refers to a family of distributions that is indexed by λ and, hence, our optimization problem translates into finding the optimal λ^* along with the corresponding distribution Q_{λ^*}. The following theorems state the mathematical definition of λ^* and introduce a tractable approximation.

Theorem 1. *The optimal λ^* is given by*

$$\lambda^* = \text{argmin}_\lambda \; \mathbb{E}_{Q_\lambda}[\log Q_\lambda(\theta)] - \mathbb{E}_{Q_\lambda}[\log P(X, \theta)] + \log P(X). \tag{A1}$$

Proof. *The fit between the variational distribution $Q_\lambda(\theta)$ and the posterior distribution $P(\theta \mid X)$ can be measured by the Kullback-Leibler divergence. Hence, we yield*

$$\lambda^* = \text{argmin}_\lambda \; KL(Q_\lambda(\theta) \,||\, P(\theta \mid X)). \tag{A2}$$

Inserting the definition of the Kullback-Leibler divergence results into

$$\lambda^* = \text{argmin}_\lambda \, \mathbb{E}_{Q_\lambda}[\log Q_\lambda(\theta)] - \mathbb{E}_{Q_\lambda}[\log P(\theta \mid X)] = \text{argmin}_\lambda \, \mathbb{E}_{Q_\lambda}[\log Q_\lambda(\theta)] - \mathbb{E}_{Q_\lambda}[\log P(X, \theta)] + \log P(X). \tag{A4}$$

–

Unfortunately, Eq. (1) is intractable, as it depends on the marginal likelihood of the model, $\log P(X)$. Therefore, the following theorem derives an approximation for the marginal likelihood of the model.

Theorem 2. *The marginal likelihood of the model$\log P(X)$ can be approximated by the evidence lower bound, ELBO(λ), i.e.,*

$$\log P(X) \geq \mathbb{E}_{Q_\lambda}[\log P(X, \theta)] - \mathbb{E}_{Q_\lambda}[\log Q_\lambda(\theta)] = ELBO(\lambda). \tag{A5}$$

DERIVATION OF OPTIMISATION PROCEDURE (CONT.)

Proof. *Utilizing Jensen's inequality, it holds that*

$$\log P(X) = \log \int P(X, \theta)\, \mathrm{d}\theta = \log \int P(X, \theta) \frac{Q_\lambda(\theta)}{Q_\lambda(\theta)}\, \mathrm{d}\theta = \log \mathbb{E}_{Q_\lambda}\left[\frac{P(X, \theta)}{Q_\lambda(\theta)}\right] \geq \mathbb{E}_{Q_\lambda}\left[\log \frac{P(X, \theta)}{Q_\lambda(\theta)}\right] = \mathbb{E}_{Q_\lambda}[\log P(X, \theta)] - \mathbb{E}_{Q_\lambda}[\log q(\theta)]. \tag{A7}$$

_

Theorem 3. *The optimal λ^* can be approximated by*

$$\lambda^* = \mathrm{argmax}_\lambda\, ELBO(\lambda). \tag{A8}$$

Proof. *From Eqs. (1) and (2), it immediately follows that*

$$\lambda^* = \mathrm{argmin}_\lambda\ \log P(X) - ELBO(\lambda). \tag{A9}$$

As $\log P(X)$ is constant with respect to λ, the value λ^* can be approximated by maximizing ELBO(λ). _

In order to optimize *ELBO*(λ), we utilize gradient descent with the gradients defined by

$$\nabla_\lambda\, ELBO(\lambda) = \nabla_\lambda\, \mathbb{E}_{Q_\lambda}[\log P(X, \theta)] - \mathbb{E}_{Q_\lambda}[\log q(\theta)] = \mathbb{E}_{Q_\lambda}[\nabla_\lambda \log q(\theta)(\log P(X, \theta) - \log q(\theta))]. \tag{A11}$$

We further utilize Monte Carlo integration to obtain the estimates of the *ELBO*(λ) and the gradient.