CM32024 Assessment 1 Part B - Project Report

David Toma, Felix Blakemore, Grace Darcy, Jayden Evetts

December 2025

Contents

1 Executive Summary

This report presents a Bayesian analysis of the degradation behaviour of an aeronautical engine component, designated "CX", based on efficiency measurements collected over time for 75 individual components. The client's data comprises of irregular, noisy measurements over up to 5,000 operating hours, together with 5 material characteristics M_1-M_5 for each component.

The client's principal objectives are:

- Characterise degradation rates.
- Predict future efficiency (especially for sparsely observed components).
- Understand the influence of material characteristics.
- Assess the practicality of deploying a prediction model in-house.
- Gain an accessible understanding of Bayesian methodology.

Getting started, we model each component's efficiency as a noisy exponential decay function. For the baseline model, the efficiency $y_i(t)$ of component i at time t (in hours) is modelled as:

$y_i(t) = f_i(t) + \epsilon_i(t), \quad \epsilon_i(t) \sim \mathcal{N}(0, \sigma_{y_i}^2), \quad f_i(t) = u_i \exp\{-\frac{v_i t}{10000}\}$. where u_i represents the initial efficiency level and v_i is a component-specific dacy rate. Time is scaled by 10,000 to align with the engineers' practice of thinking in "ten-thousand-hour" units and to improve numerical stability.

For the enhanced model, we retain the exponential form but also allow the decay rate to depend linearly on the material characteristics M_1-M_5 by use of shared global weights w_j, so that:

$$f_i(t) = u_i \exp\Big\{-\Big(v_i + \textstyle\sum_{j=1}^{5} w_j m_{ji}\Big)\, \frac{t}{10000}\Big\},$$

Both models are implemented in a hierarchical Bayesian framwork using the NumPyro probabilistic programming library and Hamiltonian Monte Carlo with NUTS (No-U-Turn Sampler). Parameters u_i and v_i are draw from common prior distributions which reflect the engineering expectations (e.g. u_1 80-100, v_i 1-10). The noise standard deviation σ_y is given a weakly informative prior. For the enhanced model, the global weights w_j are given zero-centred priors.

For both baseline and enhanced models we use standard MCMC diagnostics. We check that the $\hat{R}$ convergence statistic is close to 1.0 for all parameters, that effective sample sizes are adequate, and that there are no divergences. ArviZ trace plots (Figure A1 and B1) confirm that the Markov chains are well-mixed and stationary. Model fit is illustrated for one well-observed component as well as three sparsely observed components (Figure A2-A5), showing the observed

data, the posterior mean degradation curve, and two levels of uncertainty. The levels of uncertainty are 95% intervals for the underlying degradation function and wider 95% intervals that also include measurement noise.

The enhanced model addressed client objectives by simultaneously modelling degradation and quantifying the influence of material characteristics. This is done by using posterior inference on the weights $w_1, ..., w_5$, which identifies which material diagnostics have the strongest and most reliable influence on the decay rate. Characteristics with weight posteriors far from zero are strongly associated with faster or slower efficiency loss. This then provides actionable guidance for design and maintenance. Components with high-risk material profiles can be flagged for closer monitoring or earlier intervention.

For the blind test, we use the enhanced model to estimate, for each of the last 25 components, the probability that efficiency will be at or below 35% at $t = 40$. We compute this using the predictive distribution of $y_i(40)$ from the model, combining posterior samples of (u_i, v_i, w_j) with the exponential degradation function and then evaluating the appropriate Normal cumulative distribution function. The resulting probabilities are provided in the file *predictions.csv* and summarised in Table C1.

We also provide a stand-alone deployable predictor in the notebook *deployable.ipynb*. This implementation loads pre-computed posterior samples and material characteristics from a single model file, and executes:

```
pred_mean, pred_std=predict(index, timestamp)
```

which returns the predicted mean efficiency and the standard deviation of the underlying degradation function at an arbitrary point in the future. This gives lenience to the client, as it allows their analytics team to integrate the model into in-house tools.

Overall, this Bayesian hierarchical approach provides an interpretable, engineer-aligned model for degradation, a principled quantification of uncertainty - particularly important when handling sparse data, and a deep insight into which material characteristics drive degradation. The main limitations relate to the assumes functional form of a single exponential decay, possible identifiability issues between v_i and w_j, and the computational cost of Bayesian sampling.

2 Introduction

The CX component's performance is quantified by an efficiency measurement on a 0-100 scale, where lower values indicated degraded performance and eventual need for overhaul or replacement. The client has supplied data for 75 components, each monitored over its life to date.

The data are stored in a csv file - *bml-component-data.csv*. Each measurement comprises of:

- an ID
- an index
- five real-valued material characteristics $M_1, ..., M_5$
- a timestamp t measured in hours, [0, 4900]
- an efficiency value, 0, 100.

Measurements are both irregular in time and noisy. The first 50 components generally have many observations spanning a substantial proportion of their lifetime, while the remaining 25 were commissioned later and have relatively few observations in comparison.

This raises a few challenged. Efficiency measurements are subject to random variation and instrument noise, and any model must distinguish systematic degradation from noise. In addition, the number and timing of observations vary substantially between components, making extrapolation inherently uncertain. Finally, components differ both in their initial efficiency and in their degradation rates, meaning a model must account for both shared patterns and individual deviations.

The client has requested a Bayesian hierarchical treatment of this model to better exploit the low-data pool, to provide full uncertainty quantification, and to enable probabilistic risk assessment for high-risk maintenance decisions.

3 Bayesian Modelling Framework

This section outline the hierarchical Bayesian framework applied to the data, covering the baseline and enhanced models, while talking about prior specification, noise model, computational approach and diagnostics.

3.1 Hierarchichal Bayesian Models

In hierarchical Bayesian models, parameters for individual units aree treated as random variables drawn from common prior distributions. This allows components to have a "shared strength", which means information from well-observed components informs plausible values for more poorly observed components. For each component i we have paramters $(u_i,\ v_i)$. These are not fitted independently

component-by-component, but are given common priors reflecting existing engineering knowledge, such as likely ranges for intial efficiency and decay rate. This is where the hierarchy comes into play, the component-level parameters drawn from shared priors with hyperparameters chosen based on domain experience.

The enhanced model adds global weights $w_1, ..., w_5$ that are common across all components. They are added in hopes of better capturing the influence of $M_1, ..., M_5$ on the decay rate.

3.2 Baseline Model

The baseline model follows the equation:

$$y_i(t) = f_i(t) + \epsilon_i(t), \;\; \epsilon_i(t) \sim \mathcal{N}(0, \sigma_{y_i}^2), \quad f_i(t) = u_i \exp\{-\tfrac{v_i t}{10000}\}.$$

Here:

- u_i is the intercept term, representing efficiency at $t = 0$ in the noise-free model.
- v_i is the decay rate parameter - larger term corresponds to faster degradation.
- σ_y is the standard deviation of the measurement noise, shared across components.

Time is rescaled as $t^{'} = t/10000$, and the model is written as $f_i(t^{'}) = u_i exp(-v_i t^{'})$. This is algebraically equivalent and improves numerical conditioning.

u_i is given a Normal prior with mean 90. v_i is given a Normal prior around a plausible mid-range decay rate (3-5), with variance chosen such that values 1-10 are covered. σ_y is given a Half-Normal prior with a scale of order 1, reflecting moderate but non-negligible measurement noise. These priors are weakly informative, and rule out implausible parameter values while allowing the data to dominate where sufficient information is available.

3.3 Enhanced model

The enhanced model incorporates M_1-M_5. The following equation reflects this:

$$f_i(t) = u_i \exp\Big\{-\Big(v_i + \textstyle\sum_{j=1}^{5} w_j m_{ji}\Big)\, \frac{t}{10000}\Big\},$$

where m_{ij} is the value of the characteristic M_j for component i and w_j is a global weight shared across components. Positive w_j values indicate that larger M_j increase the decay rate, leading to faster degradation, whereas negative w_j indicate that larger M_j slow degradation.

The prior of u_i in the enhanced model is the same, v_i has a tighter and lower mean prior. $w_j \sim N(0, \tau^2)$, and σ_y stays the same as well.

3.4 Computational Method and Diagnostics

Both models are implemented using NumPyro, which provides a high-level interface to probabilistic models in Python and uses JAX for computation. Hamiltonian Monte Carlo (HMC) with NUTS (No-U-Turn Sampler) to draw samples from the posterior.

The baseline model is run with at least 500 warm-up iterations and 1000 posterior samples, but can be extended. The enhanced model is run with at least 2000 warm-up iterations and 2000 posterior samples to ensure stable inference for the global weights.

MCMC diagnostics are then monitored. We require $\hat{R} \approx 1.0$ for all reported parameters. We check that the effective sample size is not too low, and we require that the sampler reports no divergences.

Trace plots produced with ArviZ show the sampled values over iterations for key parameters, together with their marginal posterior densities.

4 Baseline Model Results

This section presents the results for the baseline exponential-decay model, focusing on posterioir diagnostics and predictive performance for selected components

4.1 Posterior Diagnostics

[FIGURE A1]

Figure A1 shows the trace plots for representative subsets of u_i and v_i, together with the noise standard deviation σ_y and any hyperparameters if sampled. Each trace displays multiple chains overlaid, along with the corresponding marginal posterior density.

From this figure, we observe that $\hat{R}$ is very close to 1.0 for all monitored parameters, indicating good mixing and convergence. The effective sample sizes are large enough, and there are no divergences. This means that the Markov Chain Monte Carlo procedure has run correctly.

4.2 Model Fit for Individual Components

To illustrate how the baseline model fits individual components, we present four representative samples. One well-observed component from the dense data, and three sparsely observed components. For each component i we computer:

- The posterior predictive mean function:

  $$\mathbb{E}[\, f_i(t) \mid \text{data}\,]$$

  as the average of $f_i(t)$ evaluated across all posterior samples of $(u_i,\ v_i)$.

- The posterior standard deviation:

$\sqrt{\text{Var}(f_i(t) \mid \text{data})}$,

which is derived from the variability of $f_i(t)$ over the posterior samples.

- The total predictive standard deviation:

$\sqrt{\text{Var}(f_i(t) \mid \text{data}) + \sigma_{y_i}^2}$,

[FIGURE A2] [FIGURE A3] [FIGURE A4] [FIGURE A5]

In each figure, the black points show the observed efficiency measurements against time. The solid curve shows the posterior mean degradation function. The darker shaded band shows the 95% interval for the function-only uncertainty, and the lighter band shows the wider 95% interval including measurement noise.

For Figure A2, the posterior mean closely follows the data, and the function-only uncertainty band is relatively narrow over the observed time range. The wider band reflects realistic variablity in any future noisy measurements.

For the other components (which are sparse), the model still captures the general decay trend, but both uncertainty bands are substantially wider, particularly at times beyond the last observation. This reflects the limited information available to identify $(u_i,\ v_i)$ and truly emphasises the benefits of a Bayesian approach.

5 Enhanced Model Results

We now turn to the enhanced model that incorporates material characteristics M_1-M_5 via global weights w_j.

5.1 Posterior Diagnostics for Material Weights

[FIGURE B1]

Figure B1 shows the trace plots for the five material weights $w_1, ..., w_5$, together with any hyperparameters associated with their priors. Similarly to the baseline model, the traces convey stable, well-mixed chains with $\hat{R}$ values near 1.0 and good sample sizes.

5.2 Effects of M_1-M_5 and Importance Ranking

The posterior distributions of the weights w_j provide direct insight into the role of each material feature. If the posterior for w_j is concentrated away from zero, then M_j has a large effect on degradation. If the posterior for w_j is centred near zero or very wide, then no consistent effect of M_j.

We summarise these results by reporting, for each j:

- The posterior mean of w_j.
- 95% credible interval.

- An interpretation of the sign and magnitude of w_j

Based on the aforementioned summaries, we can rank the various characteristic by importance.

5.3 Comparison With Baseline Decay Rates and Predictive Accuracy

Comparing the posterior summaries for v_i across models confirms that the enhanced model treats v_i as a "residual" decay term after accounting for M_1-M_5.

The enhanced model typically yields similar or improved fit for well-observed components, with uncertainty bands that are comparable to the baseline where the data is dense. There is noticeably improved calibration for sparsely observed components, as information is borrowed through both the shared priors and the global weights.

This is especially valuable for the blind-test components, where extrapolation is performed. The enhanced model can fill in the blanks by recognising that components with similar material characteristics tend to degrade in similar ways.

6 Blind Test Prediction Explanation

The blind test requires us to compute, for each of the last 25 components (sparse data), the probability that efficiency is at or below 35% at $t = 40$ hours, using the enhanced model.

6.1 Predictive Distribution at $t = 40$

For a given component i in the blind-test set, and using the enhanced model, the latent degradation function at time t is:

$$f_i(t) = u_i \exp\Big\{ - \Big(v_i + \textstyle\sum_{j=1}^{5} w_j m_{ji} \Big) \, \frac{t}{10000} \Big\}.$$

The Bayesian posterior then gives us samples $\{u_i^{(s)}, v_i^{(s)}, w_1^{(s)}, ..., w_5^{(s)}\}_{s=1}^{S}$ from the join posterior distribution. For each posterior sample s, we can compute the corresponding degradation value at $t = 40$. We can either do this:

- Via the normal CDF: treat $f_i^{(s)}(40)$ as the mean of a Normal distribution for $y_i(40)$ with standard deviation equal to $\sigma_y^{(s)}$ and compute:

  $$p_i^{(s)} = \mathbb{P}\Big(y_i(40) \le 35 \mid u_i^{(s)}, v_i^{(s)}, \mathbf{w}^{(s)} \Big) = \Phi\left(\frac{35 - f_i^{(s)}(40)}{\sigma_{y_i}^{(s)}} \right),$$

  where Φ is the standard Normal CDF

- Via threshold counting: generate a direct sample of $y_i^{(s)}(40)$ from the Normal distribution centred at $f_i^{(s)}(40)$ with standard deviation $\sigma_y^{(s)}$, then count the proportion of sample with $y_i^{(s)}(40) \leq 35$.

For the blind test we use the predictive distribution of the underlying function $f_i(40)$ without explicitly introducing the additional noise variance into the probability calculation.

6.2 Resulting Probabilities and Risk Interpretation

For each component i in indices 50-74, we averagee the per-sample probabilities $p_i^{(s)}$ over $s = 1, ...S$ to obtain:

$\hat{p}_i = \frac{1}{S} \sum_{s=1}^{S} p_i^{(s)}$.

These $\hat{p}_i$ values then form the entries of *predictions.csv*.

In general, a small value of $\hat{p}_i$ indicates that the component is very unlikely to have dropped below 35% efficiency by $t = 40$. A larger $\hat{p}_i$ points out a higher risk that the component will be at or below the overhaul threshold at that time. These probabilities can be used to prioritise potential inspections (components with the highest $\hat{p}_i$ might be scheduled for earlier maintenance than normal).

7 Practical Deployment Explanation

To assess the practicality of running the prediction model in-house, we provide a stand-alone implementation in *deployable.ipynb* based on the enhanced model.

7.1 Prediction Interface

The deployed predictor includes the function:

```
pred_mean, pred_std=predict(index, timestamp)
```

where index is the integer component index in [0,74] and timestamp is the time t (scaled internally by 1/10000). This function returns pred_mean, which is the posterior mean of the underlying degradation function $f_i(t)$ at the specified time. It also returns pred_std, which is the posterior standard deviation of $f_i(t)$.

7.2 Use of Pre-Trained Parameters

The deployment notebook loads pre-trained samples of the model parameters and the material matrix M from a file, in this case *deployable_model.pkl*. This contains the posterior samples of $u_i, v_i, w_1, ..., w_5$ and σ_y, and the matrix M_{ij} for all components. The code then retrieves the appropriate column of M for the specified components, computes the effective decay rates for each posterior

sample via $\lambda_i = v_i + \sum_{j=1}^{5} w_j m_{ji}$, and evaluates. It then aggregates these into pred_mean and pred_std.

No further training or optimisation is required at deployment time, which is optimal for what the client requires.

7.3 Practical Usage By Analytics Team

In practice, the client's analytics team can load the deployment notebook, for any CX component, call predict(index, timestamp) at any time of interest, and use pred_mean as the best estimate of expected efficienct and pred_std to gauge how uncertain that estimate is. This provides a simple and transparent tool for in-house risk assessment.

8 Discussion and Limitations

While the proposed models meet the client's objectives, there are several important limitations and considerations.

8.1 Model Misspecification

Both baseline and enhanced models assume that degradation follows a single exponential decay in time. In reality, some components may exhibit more complex patterns. If this comes to fruition, the behaviour will remain unmodelled, and the fitted exponential curves may misrepresent both short-term and long-term behaviour, particularly when extrapolating.

8.2 Irregular Sampling and Data Sparsity

Irregular measurements and sparse data for later components are already taking a toll on the model. The hierarchical structure does its best to mitigate this by borrowing information across components, but there are inherent limits. For components with very few early-life measurements, the model's predictions at $t = 40$ rely heavily on patters inferred from other components and the material characteristics. Therefore, if the new component is atypical, its predictions may be biased. Moreover, without measurements over a wide time range, multiple (u_i, v_i) combinations may fit the data similarly well/poorly.

8.3 Computational Cost

Full Bayesian inference with HMC/NUTS is computationally more expensive than simple regression on maximum-likelihood approaches. For the current dataset, run times are manageable, but re-training the model frequently may require scheduling or batch processing, and scaling to substantially larger numbers of components or more complex models would increase computational demands.

8.4 Future Work

Several extensions could improve robustness:

- Introducing explicit hyperparameters for the distributions of u_i and v_i and sampling them from data would potentially improve the borrowing aspect across components.
- Using Bayesian model comparison tools, such as WAIC or LOO-CV to assess whether the enhanced model provides materially better predictive performance than the baseline.
- Potentially adding and using additional operational covariates, such as temperature cycles, load profiles, etc.

9 Conclusion

This project has applied hierarchical Bayesian modelling to address the client's objectives regarding the degradation of CX components:

- We have characterised the degradation rate of each component using an exponential decay model, with component-specific parameters reflecting initial efficiency and decay rate.
- We have produced probabilistic predictions of future efficiency, especially for sparsely observed components by propagating posterior uncertainty in degradation curves.
- We provided guidance for design optimisation and targeted monitoring by implementing an enhanced model using global weights and material characteristics.
- We demonstrated that the Bayesian model can be run in-house using a simple interface.
- We explained the rationale for using hierarchical Bayesian methods

In summary, our hierarchical approach offers a principled and practically usedful framwork for modelling CX component degradation, assessing influence of material characteristics while accommodating for their own personal needs. While there is scope for further improvements, the current models provide the client with a ready package for improved efficiency in their company.