Modular Emulation and Bayesian Calibration of Stellar Evolution Models with Gaussian Processes

Ravleen Bajaj (ravleen_bajaj@sfu.ca), Department of Statistics & Actuarial Sciences, Simon Fraser University

Introduction

Astrophysicists use computer-based stellar evolution models to infer a star's physical properties from its photometry, i.e., its brightness through different light filters. These models exhibit complex, non-stationary behaviour, particularly in late evolutionary stages. Conducting extensive analyses or quantifying uncertainties through repeated simulation runs is computationally expensive. This motivates the development of a statistical emulator based on (deep) Gaussian processes (GPs). We adopt a multi-stage, modular approach to both emulation and calibration, allowing for pragmatic inference while retaining uncertainty quantification.

Gaussian Process Emulation

Consider data y to be a noisy version of a computer model/simulator $\eta(\cdot)$ at the "correct" input θ:

$$y = \eta(\theta) + \epsilon \qquad \epsilon \sim \mathcal{N}(\vec{0}, \Sigma_y).$$

If $\eta(\cdot)$ is slow to evaluate, we require an **emulator**—a probabilistic model for $\eta(\cdot)$ at untried settings. With $\eta(x_i)$, $i = 1, \ldots, n$, denoting n runs of $\eta(\cdot)$ at inputs $x_1, \ldots, x_n$, a common model for $\eta(\cdot)$ is a **Gaussian Process (GP)** [1] such that:

$$\begin{pmatrix} \eta(x_1) \\ \vdots \\ \eta(x_n) \end{pmatrix} \sim (\vec{\mu}, k(x, x'))$$

where $k(x, x')$ is a **covariance function** that we specify. A common choice for $k(x, x')$ is **squared exponential** such that:

$$k(x, x') = \frac{1}{\lambda_\eta} \exp\left(-\vec{\beta}\|x - x\|^2\right).$$

A GP model prior to (Figure 1) and after conditioning on (simulated) observations (Figure 2) is illustrated below.

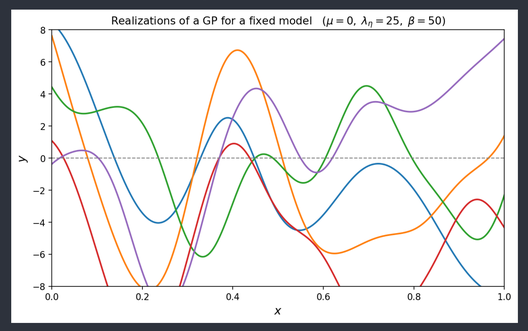


Figure 1. Realizations of a GP for a fixed model

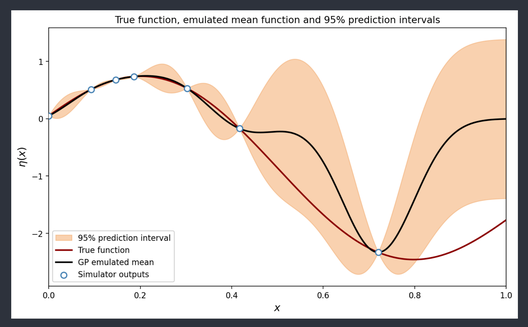


Figure 2. 1-D Realisation of the Fitted GP using Sq. Exponential Kernel

Modularized Bayesian Calibration

Rather than learn about θ and the GP hyperparameters $(\lambda_\eta, \vec{\beta})$ together, a **modular** approach first fits $\eta(\cdot)$ using only the simulator runs and then uses the fitted emulator, $\hat{\eta}(\cdot)$, for inference about θ. It is therefore a two-step process.

Step A: Obtain the fitted emulator, $\hat{\eta}(\cdot)$, i.e., obtain μ^* and Σ^*, which are functions of λ_η and $\vec{\beta}$, such that, $\hat{\eta} \sim \mathcal{N}(\mu^*, \Sigma^*)$.

Step B: Model $y \sim \mathcal{N}(\hat{\eta}(\theta), \Sigma_y)$, and use typical Bayesian methods for inference.

Illustrative Example

We now illustrate the modular calibration approach using a simple example. Consider a simulator given by:

$$\eta(\theta) = \theta^2 + e^{0.5\theta}$$

with $\theta = 0.5$ being the unknown model parameter. We generate $y \sim \mathcal{N}(\eta(0.5), 0.25^2)$ and take a Bayesian approach for model fitting, specifying the following priors:

$$\vec{\beta} \sim \text{Gamma}(2, 1), \quad \lambda_\eta \sim \text{Gamma}(5, 5), \quad \theta \sim \mathcal{N}(0.5, 0.3^2).$$

Three calibration scenarios are compared, summarized in Table 1, with each progressively relaxing fixed components to propagate additional sources of uncertainty.

	Case 1	Case 2	Case 3
GP hyperparameters $\boldsymbol{\beta}, \lambda$	Fixed at posterior medians $\hat{\boldsymbol{\beta}}, \hat{\lambda}$.	Fixed at posterior medians $\hat{\boldsymbol{\beta}}, \hat{\lambda}$.	Resampled at each step from Step-A chain.
Emulator mean $\boldsymbol{\mu}^*$	Plugged in deterministically.	Drawn: $\boldsymbol{\mu}^* \sim \mathcal{N}(\hat{\eta}_{\text{mean}}, \text{diag}(\hat{\eta}_{\text{var}}))$.	Drawn as Case 2, with current $(\boldsymbol{\beta}^{(j)}, \lambda^{(j)})$.
Emulator variance $\boldsymbol{\Sigma}^*$	N/A	Enters via the random draw of $\boldsymbol{\mu}^*$.	Enters via each random draw under current $(\boldsymbol{\beta}^{(j)}, \lambda^{(j)})$.
Uncertainty	Field noise.	Field noise + GP predictive.	Field noise + GP predictive + hyperparameter.

Table 1. Modularized calibration with varying uncertainty.

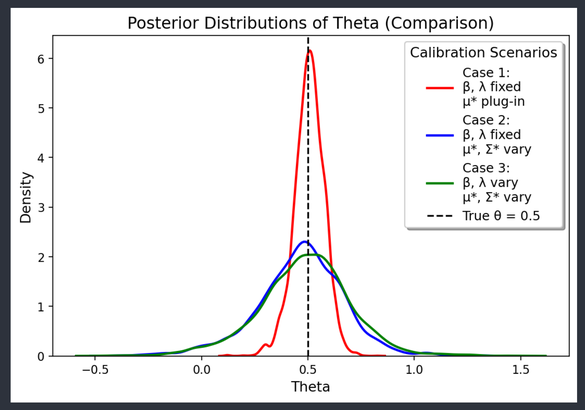


Figure 3. Posterior distribution of θ with varying conditions for calibration.

Stellar Evolution Tracks

For each of 3 wavelength bands, B, U, and V, we have 108 stellar evolution *runs*, with each run indexed by a pair of physical inputs $\boldsymbol{x}_j = ([\text{Fe/H}]_j, \log \text{Age}_j) \in \mathbb{R}^2$. For each run, we have a discretised functional output that is the observed brightness of the star in the given band along a fixed grid of Equivalent Evolutionary Points (EEPs); see Figure 4.

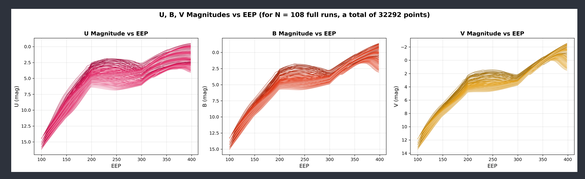


Figure 4. U, B, and V value as a function of EEP for the 108 stellar evolution runs. Each coloured curve is a ($[\text{Fe/H}]$, $\log \text{Age}$) realisation.

Functional PCA Reduction and LOOCV

We reduce the simulator output using the first $K = 3$ principal components obtained from the thin SVD of the mean-centred data. Following [1, 2], the simulator output is represented as

$$\hat{\eta}(x) = \sum_{k=1}^{K} \phi_k w_k(x) + \varepsilon, \quad x \in [0, 1]^2,$$

where ϕ_k are orthogonal basis vectors, $w_k(x)$ are input-dependent weights, and ε is an error term.

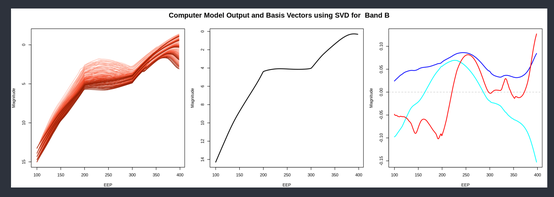


Figure 5. **Left**: B wavelength simulator runs. **Middle**: Mean output vector. **Right**: Set of 3 orthogonal basis vectors.

Each weight $w_k(x)$ is modeled independently using a zero-mean GP with a Matérn $5/2$ kernel, such that:

$$k(x, x') = \left(1 + \sqrt{5}\,\delta + \tfrac{5}{3}\delta^2\right) \exp(-\sqrt{5}\,\delta), \delta^2 = \sum_{d=1}^{2} \frac{(x_d - x'_d)^2}{\ell_{k,d}}.$$

We assess emulator performance using leave-one-out cross-validation (LOOCV) computing RMSE on the held out run.

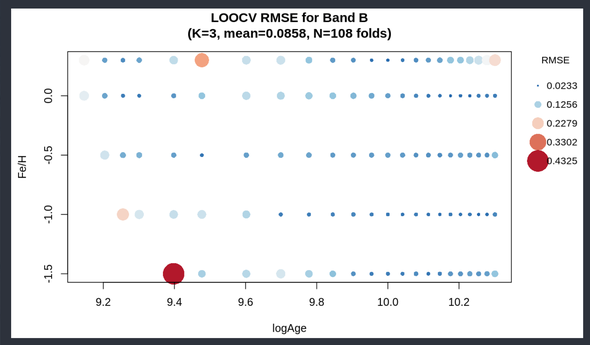


Figure 6. Per-fold LOOCV RMSE for the B-band emulator, plotted at each held-out run's location in the ($\log \text{Age}$, $[\text{Fe/H}]$) input plane.

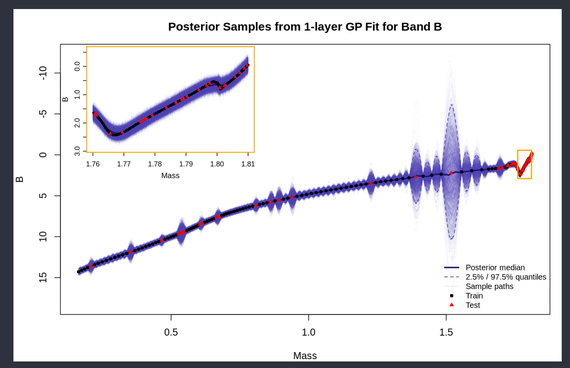
Deep Gaussian Processes (Deep GPs)

A standard GP assumes *stationarity*. Stellar evolution runs are smooth across most of the mass (i.e., EEP) range but exhibit more complex behavior at higher masses owing to the underlying physics. To model this non-stationarity in the response, we propose to use a Deep GP (DGP) which addresses this by stacking GP layers [3], i.e.,

$$Y_j \mid W \sim \mathcal{N}(0, \Sigma(W)), \qquad W_j \sim \mathcal{N}(0, \Sigma_j(\boldsymbol{x_j})),$$

so that the input x_j is first warped by a latent layer W before being passed to the outer GP.

In the proposed framework, posterior samples from the fitted DGP would serve as the simulated runs for obtaining $\hat{\eta}(x)$. We can then compare calibration scenarios under increasing levels of uncertainty propagation (similar to the illustrative example) to assess how non-stationary surrogate uncertainty affects the calibration posterior.



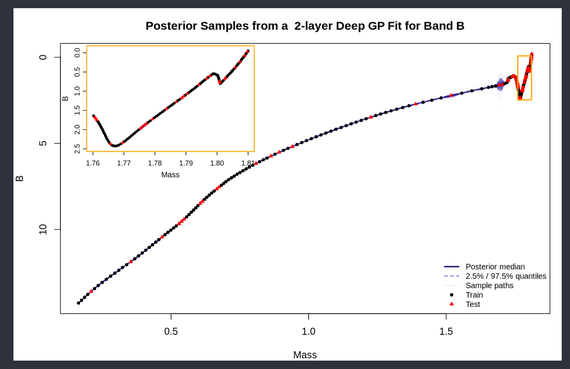


Figure 7. Posterior realizations with the B wavelength band as a function of stellar mass at fixed $[\text{Fe/H}] = -0.5$, $\log \text{Age} = 9$. **Top:** One-layer GP posterior with a stationary Matérn kernel as the inset. **Bottom:** Two-layer DGP posterior.

References

[1] Jim Gattiker, Dave Higdon, Sallie Keller-McNulty, Michael McKay, Leslie Moore, and Brian Williams. Combining experimental data and computer simulations, with an application to flyer plate experiments. *Bayesian Analysis*, 1, 12 2006.

[2] D. Higdon, E. Lawrence, K. Heitmann, and S. Habib. Simulation-aided inference in cosmology. In E. D. Feigelson and G. J. Babu, editors, *Statistical Challenges in Modern Astronomy V*, pages 41–57. Springer, New York, 2012.

[3] A. Damianou and N. D. Lawrence. Deep Gaussian processes. In *Proc. 16th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 207–215, 2013.

SFU SIMON FRASER UNIVERSITY