Data Estimation and Inference Lab

Mrinank Sharma

1 Introduction

The *Gaussian Process (GP)* is a Bayesian non-parametric model which generalises the multivariate Gaussian distribution to functions, allowing for probablistic inference and reasoning over functions. This technique is particularly useful in prediction and forecasting problems. Here, we apply this model to predict for missing sensor measurements for a portion of data from `Sotonmet`, who publish measurements of quantities such as the tide height and air temperature for sailors and port authorities. The problem of missing data is especially pertinent in this context, as the exposure of the sensors to poor weather and other conditions can result in sensor failure. Concretely, we consider the tasks of retrospective prediction of missing readings as well as forecasting of future readings.

2 Introduction to GP Regression

Please note that this section primarily recapitulates parts of Chapter 2, Rasmussen and Williams (2005).

Definition 2.1 (Gaussian Process) *A Gaussian Process (GP) is a collection of random variables, any finite number of which have a joint Gaussian distribution.*

A GP is completely specified by its mean and covariance function, and defines a distribution over functions, as we can consider a function to be an infinite collection of numbers. The notation,

$$f(\boldsymbol{x}) \sim \mathcal{GP}(\mu(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')), \tag{1}$$

means that the function $f(\boldsymbol{x})$ is distributed according to a GP with mean function $\mu(\boldsymbol{x})$ and covariance function $k(\boldsymbol{x}, \boldsymbol{x}')$.

For Bayesian regression, we have training data $\mathcal{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$ where $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y \in \mathbb{R}$, and we seek to learn the posterior predictive distribution, i.e., $p(y^*|\boldsymbol{x}^*, \mathcal{D})$ where the $*$ denotes a point of interest. N is the number of training data points.

In order to perform inference, we must define a likelihood function and prior. We will assume a Gaussian likelihood, meaning that the measurements are obtained by corrupted the true function value with Gaussian noise. Often, the prior mean function is assumed to be 0, but not that this is not particularly restrictive as we may normalise the input data. Additionally, this does not constrain the mean of the predictive posterior to be 0.

$$\begin{aligned} p(y|\boldsymbol{x}, f) &= \mathcal{N}(y; f(\boldsymbol{x}), \sigma_n^2), & (2)\\ p(f) &= \mathcal{GP}(0, k(\boldsymbol{x}, \boldsymbol{x}')). & (3) \end{aligned}$$

We will refer to σ_n is known as the *model noise*. This prior and likelihood function yields a GP posterior on f:

$$\begin{aligned} p(f|\mathcal{D}) &= \mathcal{GP}(\mu_{\text{post}}(\boldsymbol{x}), k_{\text{post}}(\boldsymbol{x}, \boldsymbol{x}')), & (4)\\ \mu_{\text{post}}(\boldsymbol{x}) &= k(\boldsymbol{x}, \boldsymbol{X})[k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}]^{-1}\boldsymbol{y}, & (5)\\ k_{\text{post}}(\boldsymbol{x}, \boldsymbol{x}') &= k(\boldsymbol{x}, \boldsymbol{x}') - k(\boldsymbol{x}, \boldsymbol{X})[k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}]^{-1} k(\boldsymbol{x}, \boldsymbol{x}')^T. & (6) \end{aligned}$$

where $\boldsymbol{X} \in \mathbb{R}^{N \times d}$ is the matrix of training inputs and $\boldsymbol{y} \in \mathbb{R}^N$ is the vector of training targets. $k(\boldsymbol{X}, \boldsymbol{X}) \in \mathbb{R}^{N \times N}$ is the *kernel matrix*, evaluating the kernel function of each pair of input points, and $k(\boldsymbol{x}, \boldsymbol{X}) \in \mathbb{R}^{1 \times N}$ evaluates the kernel function between the input $\boldsymbol{x}$ and each training point.

The posterior predictive has the following form:

$$\begin{aligned} p(y^*|\boldsymbol{x}^*, \mathcal{D}) &= \mathcal{N}(y^*; \mu^*, \sigma^{2,*}), & (7)\\ \mu^* &= k(\boldsymbol{x}^*, \boldsymbol{X})[k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}]^{-1}\boldsymbol{y}, & (8)\\ \sigma^{2,*} &= k(\boldsymbol{x}^*, \boldsymbol{x}^*) - k(\boldsymbol{x}, \boldsymbol{X})[k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 \boldsymbol{I}]^{-1} k(\boldsymbol{x}, \boldsymbol{x}')^T + \sigma_n^2. & (9) \end{aligned}$$

The log marginal likelihood can be written as follows:

$$\begin{aligned} \log p(\boldsymbol{y}|\boldsymbol{X}) = &-\frac{1}{2}\boldsymbol{y}^T[k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 I]^{-1}\boldsymbol{y} & (10)\\ &-\frac{1}{2}\log|k(\boldsymbol{X}, \boldsymbol{X}) + \sigma_n^2 I| - \frac{N}{2}\log(2\pi). & (11) \end{aligned}$$

The choice of kernel has significant implications in GP regression. Often, a kernel is parameterised by some values and in this case, it is essential to choose appropriate values. We will refer to these values as *hyperparameters* One way of choosing these values is by *maximum likelihood* i.e. choose the hyperparameters which maximimise the likelihood of observing the data.

3 Experiments & Results

In this section, we perform regression tasks, including both lookahead prediction and retrospective predictive, using GPs with different covariance functions on the provided `Sotonmet` dataset. The input is taken to be the *reading time, t* and the target is the *tide height reading, $\tilde{y}$.*

Data Preprocessing. t is defined as the duration in minutes after the first reading. The target, y, is produced by normalising $\tilde{y}$ such that it has zero mean and unit variance.

Kernels Considered. We consider the following kernels. Note that (typically) σ determines the signal scale, and ℓ determines the length scale.

A. **Radial Basis Function (RBF) Kernel**

$$k_{\text{RBF}}(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\Big[-\frac{||\boldsymbol{x} - \boldsymbol{x}'||_2^2}{2\ell^2}\Big]. \tag{12}$$

B. **Periodic Kernel**

$$k_{\text{Per}}(\boldsymbol{x}, \boldsymbol{x}') = \sigma^2 \exp\Big[-\frac{2\sin^2(\pi||\boldsymbol{x} - \boldsymbol{x}'||_2/p)}{\ell^2}\Big]. \tag{13}$$

C. **Locally Periodic Kernel**

$$k_{\text{l-Per}}(\boldsymbol{x}, \boldsymbol{x}') = k_{\text{Per}}(\boldsymbol{x}, \boldsymbol{x}') \cdot k_{\text{RBF}}(\boldsymbol{x}, \boldsymbol{x}'), \tag{14}$$

where the individual kernel functions are **not** constrained to have the same length scale.

D. **Periodic + RBF Kernel**

$$k_{\text{summed}}(\boldsymbol{x}, \boldsymbol{x}') = k_{\text{Per}}(\boldsymbol{x}, \boldsymbol{x}') + k_{\text{RBF}}(\boldsymbol{x}, \boldsymbol{x}'), \tag{15}$$

where the individual kernel functions are **not** constrained to have the same length scale **or** scale.

Kernel hyperparameters were chosen through a combination of maximum likelihood and manual tuning, and optimisation was performed through using the *BFGS* algorithm (Wikipedia contributors, 2019a). In cases where this was not stable, the *Newton-Raphson* optimisation technique was used (Wikipedia contributors,

Kernel	Standardised RMSE	Standardised Held-Out RMSE	Mean Predictive Held-Out Per-Point Log Likelihood
RBF	0.172	0.328	0.929
Periodic	0.210	0.232	−35.31
Locally Periodic	0.170	0.321	0.954
RBF + Periodic	0.059	0.097	1.279

Table 1: Tabulated Results for optimal hyperparameters found for different kernel functions. Root Mean Square Error (RMSE) reported both for all points and the average predictive log-likelihood per-point and RMSE is reported for the missing points.

2019b). In all cases, finite differences were used to estimate required gradients.

Model Noise. $\sigma_n^2 = 0.25^2$ was chosen as well calibrated value. Please see Appendix B for a more detailed discussion.

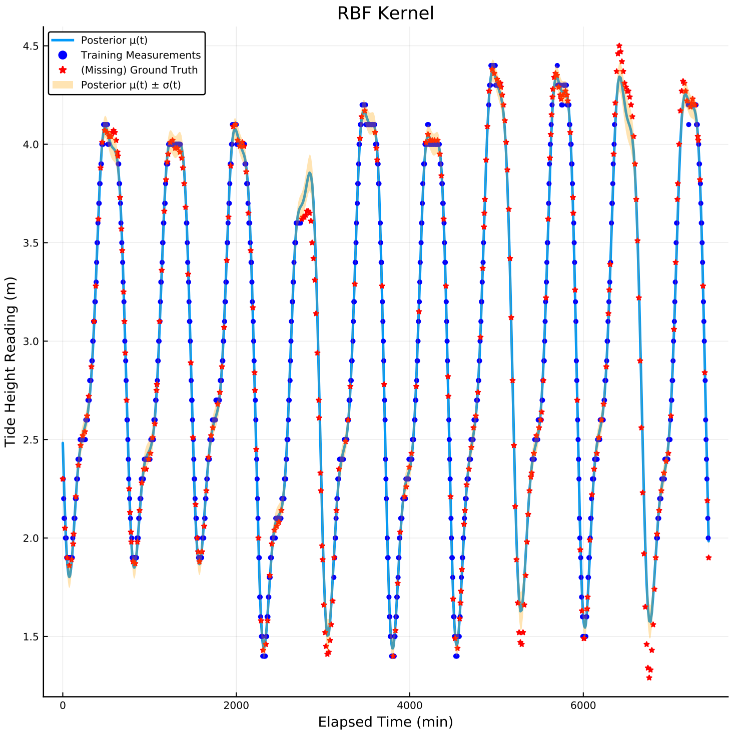

Figure 1: Retrospective predictions for the summed RBF and periodic kernel with the best hyperparameters found.

Lookahead Predictions. Please see the following links for lookahead prediction animations for the RBF + periodic kernel and the locally periodic kernel.

Tabulated Results. Please see Table 1 for tabulated results for different kernel values.

Please see Appendix A for the numerical hyperparameter values used.

4 Discussion

The choice of kernel has a significant affect on the performance of the GP across both the retrospective prediction task and the lookahead forecasting task. For the retrospective task, the RBF + Periodic kernel was found to be the best across all error metrics.

However, inspecting the lookahead animation plots, it is worth noting that forecasting with the RBF + periodic kernel has several underdesirable properties. Namely, the uncertainty estimates are poorly calibrated, with the mean function at a given point changing dramatically as nearby data arrives, *even though the model seems to be very certain*. Additionally, the uncertainty of points in the far future decreases significantly as more data arrives; this is due to the periodic component of the kernel effectively encoding a very *strong* prior. Thus for this task, the locally periodic kernel is far more appropriate as its uncertainty seems to be significantly better calibrated as seen in the animation.

Thus, a clear failure mode for GPs is when the kernel function chosen is inappropriate for the task to be performed or the underlying data. Please see Appendix C for examples of this.

There were a number of numerical issues when implementing the GP; namely, even though a valid kernel matrix must be positive definite, the generated kernel matrix had negative eigenvalues of very small magnitude. *Jitter* was added to the kernel matrix to allieviate this problem i.e., $10^{-10}\boldsymbol{I}$ was added to the kernel matrix. The typical issues of unstable matrix inversion of $k(\boldsymbol{X},\boldsymbol{X}) + \sigma_n^2\boldsymbol{I}$ were not faced due to the project implementation in `julia`, which automatically stabilises such calculations.

There were significant difficulties when attempting to optimise the hyperparameters. The method of finite differences led to instability in many of the standard optimisation algorithms and often these algorithms did not convergence. An appropriate optimisation initialisation can mitigate this issue, and such an initialisation was found by trial-and-error.

It is suprising that the locally periodic kernel only performs marginally better than the RBF kernel, as we would have expected the prior periodicity information to give a large advantage. The optimal hyperparameter learnt for the period far exceeds the true period, and thus the periodic component of this kernel plays a small role. It is suggested that this is due to a numerical optimisation issue.

5 Conclusions

Provided that the prior covariance function, the GP process can give excellence performance for regression problems, as observed here for predicting missing measurements in the `Sotonmet` dataset. The optimal covariance function does not only dependent on the data, but also depends on the type of regression problem being solved. There are numerical issues when applying these models, such as certain matrices being poorly conditioned, but there exist techniques to mitigate these issues. It is suggested that using an automatic differentiation technique to perform hyperparameter optimisation could also help alleivate this.

References

C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005. ISBN 026218253X.

Wikipedia contributors. Broyden–fletcher–goldfarb–shanno algorithm — Wikipedia, the free encyclopedia, 2019a. URL https://en.wikipedia.org/w/index.php?title=Broyden%E2%80%93Fletcher%E2%80%93Goldfarb%E2%80%93Shanno_algorithm&oldid=906946674. [Online; accessed 22-October-2019].

Wikipedia contributors. Newton's method — Wikipedia, the free encyclopedia, 2019b. URL https://en.wikipedia.org/w/index.php?title=Newton%27s_method&oldid=914039986. [Online; accessed 22-October-2019].

Wikipedia contributors. 68–95–99.7 rule — Wikipedia, the free encyclopedia, 2019c. URL https://en.wikipedia.org/w/index.php?title=68%E2%80%9395%E2%80%9399.7_rule&oldid=913993169. [Online; accessed 21-October-2019].

Appendices

A Optimal Hyperparameters

The optimal hyperparameters found were:

A. **RBF Kernel.** $\ell \simeq 94.63$ min, $\sigma = 1.35$ m.
B. **Periodic Kernel.** $\ell \simeq 1.13$ min, $\sigma \simeq 1.23$ m, $p \simeq 742$ min.
C. **Locally Periodic Kernel.** $\ell_{\text{periodic}} \simeq 250$ min, $\sigma \simeq 1.10$ m, $p \simeq 13000$ min, $\ell_{\text{RBF}} \simeq 107$ min.
D. **RBF + Periodic Kernel.** $\ell_{\text{RBF}} \simeq 92.76$ min, $\sigma_{\text{RBF}} \simeq 0.23$ m, $p \simeq 742$min, $\ell_{\text{periodic}} \simeq 1.00$ min, $\sigma_{\text{periodic}} \simeq 1.16$ m.

Note that the units of these parameters correspond to the normalised input data.

Fig. 2 shows graphical results obtained used these hyperparameter values.

B Model Noise Calibration

$\sigma_n = 0.25$ m (when corresponding to the standardised signal) was chosen. This was chosen to give good output uncertainty calibrating; Fig. 4 and Fig. **??** shows GP function and predictive posteriors for $\sigma_n = 0.1$ m and $\sigma_n = 0.25$ m.

These plots show that the latter value of σ_n is significantlly better calibrated as the measurements actually lie within the region of uncertainty which is not the case for $\sigma_n = 0.1$ m.

Since the uncertainty regions correspond to $\pm\sigma(t)$, and that 68% of values typically lie within a one standard deviation bound from the mean for the Gaussian distribution (Wikipedia contributors, 2019c), it is possible that this value of model noise is in fact too large, but a smaller value was not investigated due to time constraints.

C Failure Modes

A sigificant failure mode for the GP model is when the kernel function is chosen inappropriately. Fig. 5 shows an example with the RBF kernel where ℓ is chosen too small, resulting in uncertainty increasing between nearby training points and no useful predictions for the missing data (as the posterior effectively reverts to the prior).

Fig. 6 shows an additional example with the RBF kernel where ℓ is chosen too large, resulting in excessive smoothing and poor performance.

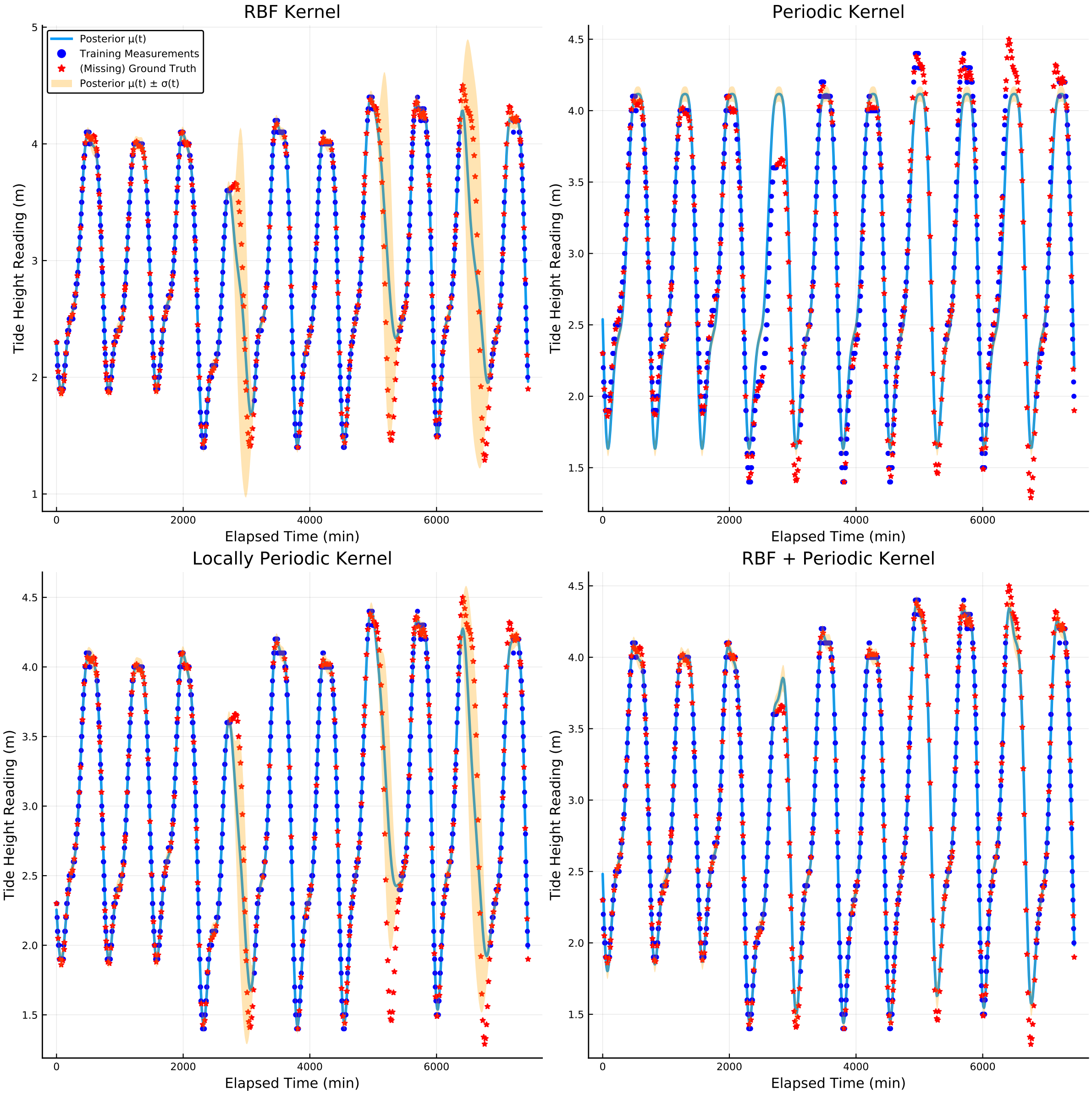

Figure 2: Retrospective predictions for the the four kernels considered, using the the best hyperparameters found.

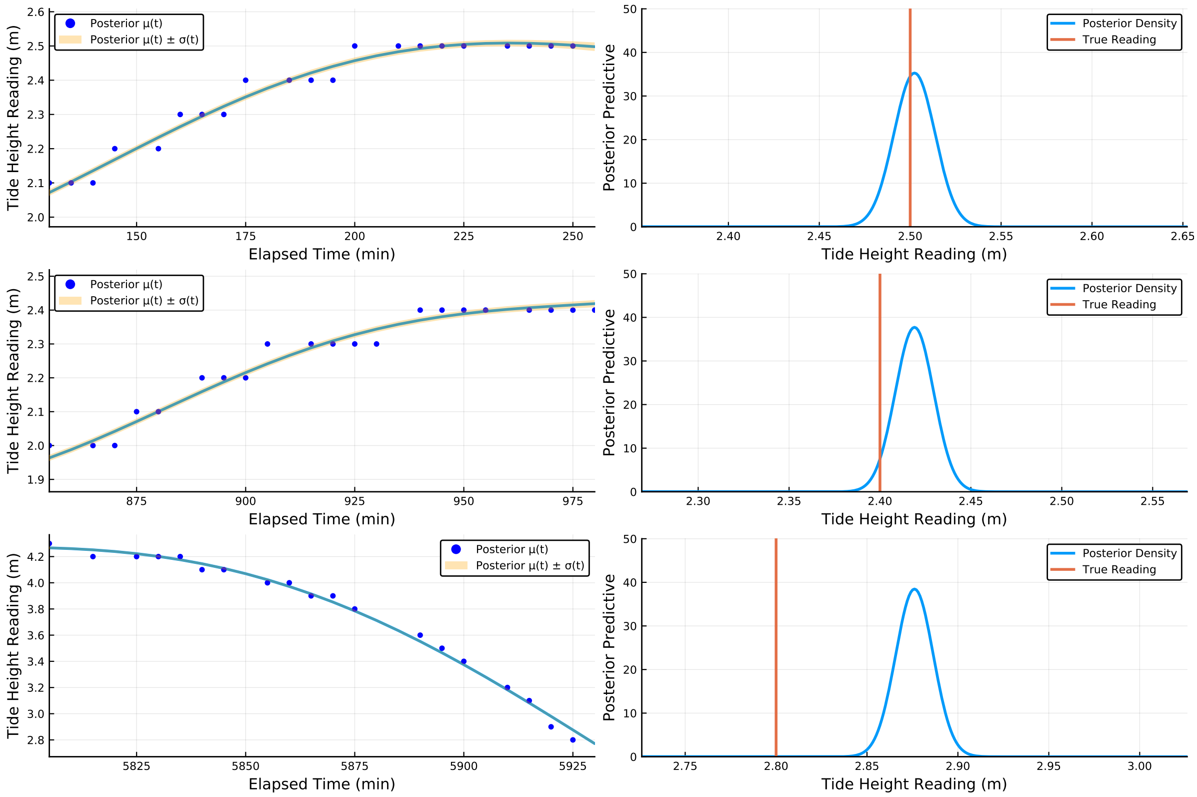

Figure 3: GP posterior given different training points (note that all training points until the latest point plotted were provided to the GP). Right plots show the predictive posterior density for the final measurement on the corresponding left hand side plot. $\sigma_n = 0.1$.

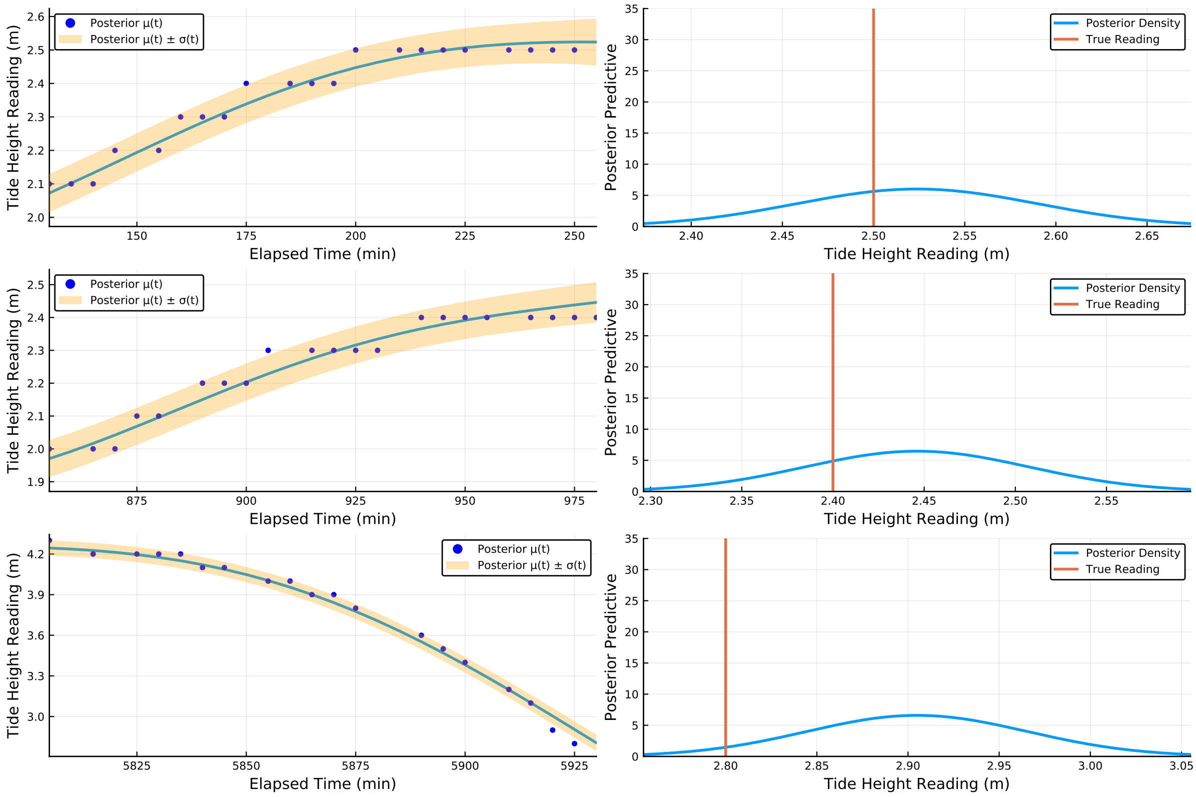

Figure 4: GP posterior given different training points (note that all training points until the latest point plotted were provided to the GP). Right plots show the predictive posterior density for the final measurement on the corresponding left hand side plot. $\sigma_n = 0.25$.

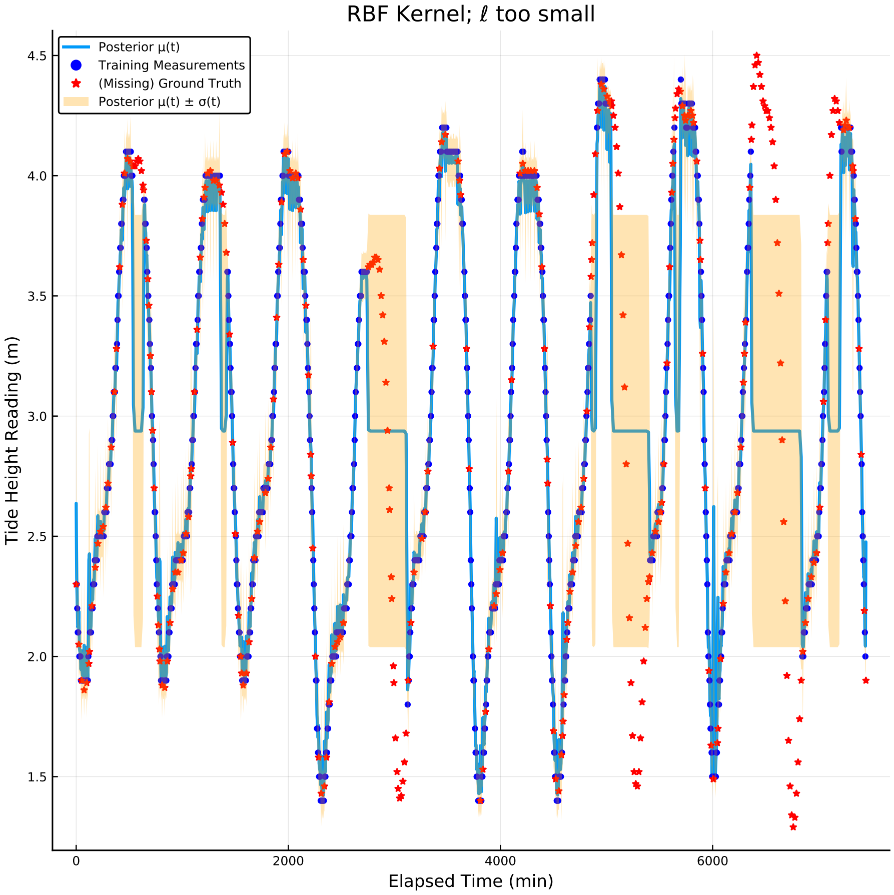

Figure 5: GP failure mode; ℓ is chosen too small using an RBF kernel, resulting in poor predictions for the missing data. $\ell = 5$ min

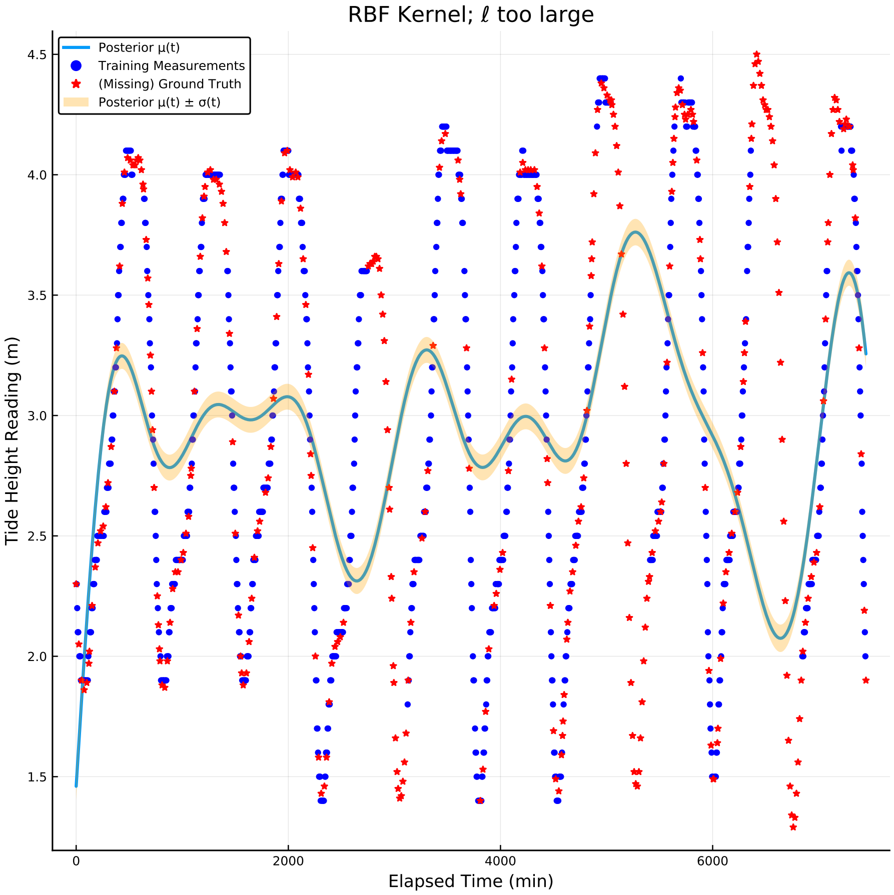

Figure 6: GP failure mode; ℓ is chosen too large using an RBF kernel, resulting in poor predictions for both the missing data and the training data, as too much smoothing is applied. $\ell = 600$ min.