UNIVERSITY OF CAMBRIDGE

MPhil in Machine Learning and, Machine Intelligence 2018-2019

Module Coursework Feedback

Module Title: Probabilistic Machine Learning

Module Code: 4F13

Candidate Number: F606F

Coursework Number: 1

I confirm that this piece of work is my own unaided effort and adheres to the Department of Engineering's guidelines on plagiarism. √

Date Marked: Click here to enter a date. Marker's Name(s): Click here to enter text.

Marker's Comments:

This piece of work has been completed to the following standard *(Please circle as appropriate)*:

	Distinction			**Pass**			**Fail (C+ - marginal fail)**		
Overall assessment (circle grade)	**Outstanding**	**A+**	**A**	**A-**	**B+**	**B**	**C+**	**C**	**Unsatisfactory**
Guideline mark (%)	90-100	80-89	75-79	70-74	65-69	60-64	**55-59**	**50-54**	**0-49**
Penalties	**10% of mark for each day, or part day, late (Sunday excluded).**								

The assignment grades are given **for information only**; results are provisional and are subject to confirmation at the Final Examiners Meeting and by the Department of Engineering Degree Committee.

4F13 Probabilistic Machine Learning: Coursework 1: Gaussian Process (996 words)

a.

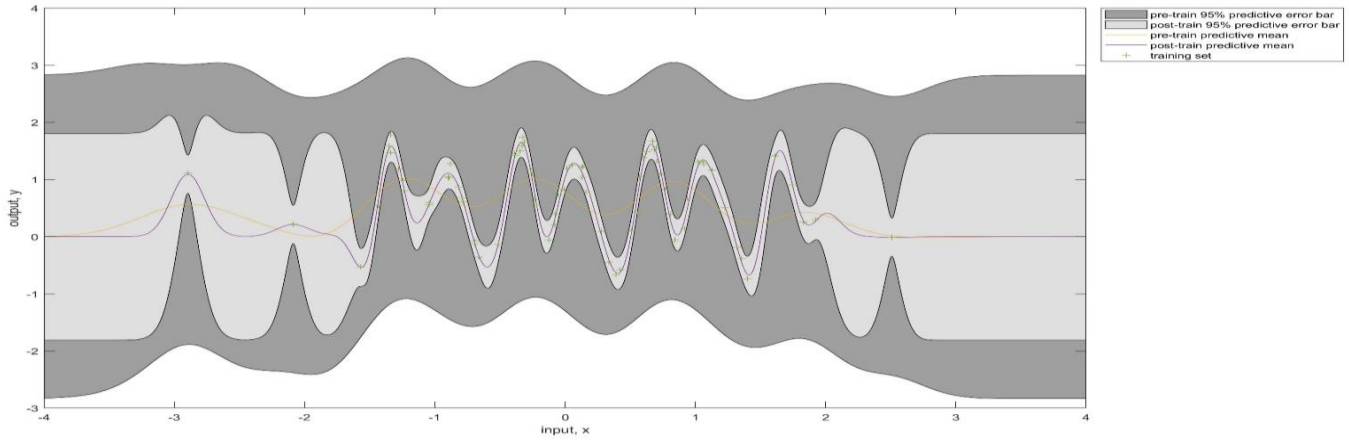

Figure 1 shows the predictive means and error bars before and after the minimization of nlml.

Table 1 shows the values nlml and hyperparameters before and after the minimization of nlml.

	Negative Log Marginal Likelihood ($nlml$)	Length-scale (l)	Signal Standard Deviation (v)	Noise Standard Deviation (σ_{noise})
Pre-train	92.91	0.37	1.00	1.00
Post-train	11.90	0.13	0.90	0.12

After the Negative Log Marginal Likelihood (nlml) is minimized from 92.91 to 11.90, the area of the shaded region for the predictive error bar is greatly reduced, especially for the region with more data points since the more data there is, more certain the prediction is.

The corresponding hyperparameters are then optimized as shown in the Table 1. Both signal and noise standard deviations are reduced so that the complexity penalty is reduced, and the data is more fit. The characteristic length-scale is also reduced, so that the value slightly exceeds the average period of the x-data in which the average separation between x-data points is 0.0721 < 0.13, which makes the signal variance less dominant to the covariance function.

---

b.

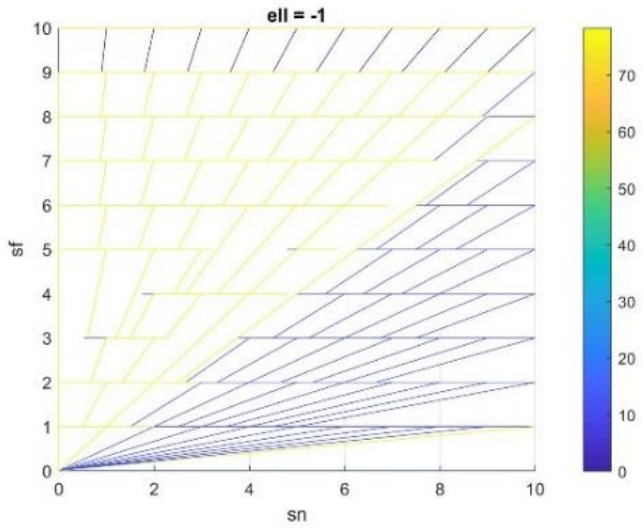

Figure 2 shows the nlml distribution with fixed initial setting of length-scale but different initial settings of signal and noise variance.

By initializing the hyperparameters differently, different local optimums are found. From Figure 2, different colours in the colour bar represent different values of nlml, but only two colours appear in the colormap. This means nlml is always minimized to a limited number of local minima, the minimization of nlml and optimization of hyperparameters depend on the initial settings of the hyperparameters. From the range of values have been tested, there are three nlml minima found, which are 11.90, 78.22 and 106.35 (Table 2).

Table 2 shows the initialization hyperparamters and their corresponding hyperparameters and nlml values after optimization.

Initialisation Hyperparameters		Negative Log Marginal Likelihood ($nlml$)	Length-scale (l)	Signal Standard Deviation (v)	Noise Standard Deviation (σ_{noise})
hyp.cov	hyp.lik				
[-1 , 0]	0	11.90	0.13	0.90	0.12
[-0.25, 0]	0	78.22	8.04	0.70	0.66
[-10 , 0]	0	106.35	0.00	0.71	0.71
[10 , 0]	0	106.35	0.00	0.71	0.71
[-0.25, -5]	0	78.22	8.04	0.70	0.66
[-10, 5]	0	106.35	0.00	0.71	0.71
[10, -5]	0	106.35	0.00	0.71	0.71
[-0.25, 0]	5	78.22	8.04	0.70	0.66
[-10, 0]	-5	106.35	0.00	0.71	0.71
[10, 0]	-5	106.35	0.00	0.71	0.71

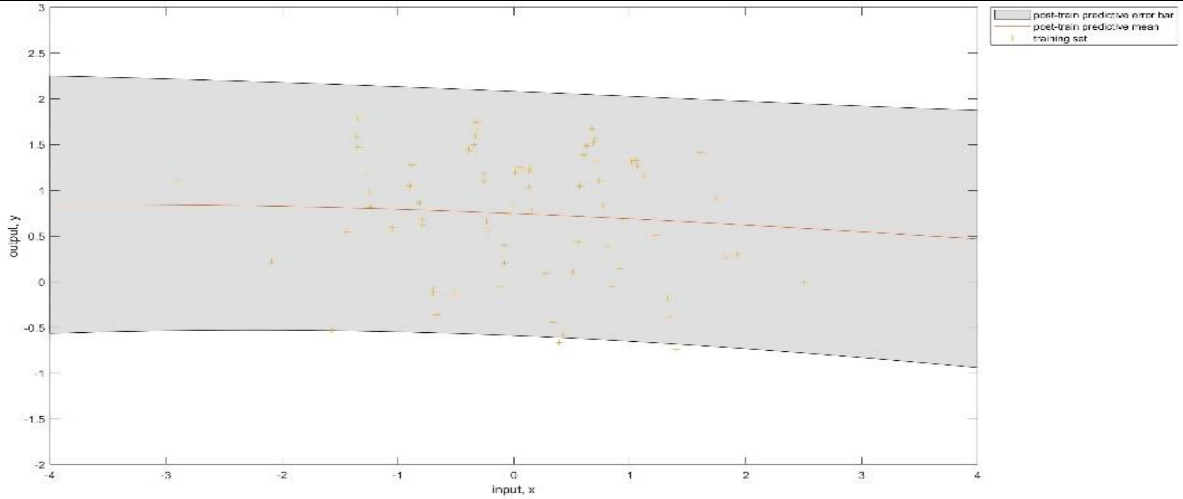

Figure 3 shows the predictive mean and error bar when the nlml is 78.22.

Table 3 shows the minimized results of nlml values from part a and b.

Part	Initialisation Hyperparameters		Negative Log Marginal Likelihood ($nlml$)	Length-scale (l)	Signal Standard Deviation (v)	Noise Standard Deviation (σ_{noise})
	hyp.cov	hyp.lik				
A	[-1, 0]	0	11.90	0.13	0.90	0.12
B	[0 , 0]	0	78.22	8.04	0.70	0.66

So to find next minimized value of nlml (78.22), the gaussian process is initialized with the hyperparameters (log(length-scale) = 0, log(variance of signal) = 0, log(variance of noise)= 0). After nlml reaches the local minimum, the hyperparameters are also optimized to the following values as shown in Table 3 (Part B) which produce the plot shown in Figure 3.

Compared with the nlml values (a: 11.90, b: 78.22), the optimized hyperparameters from part (a) is a better description to the data. The plot (Figure 1) in part (a) also shows smaller area of predictive error bars.

c.

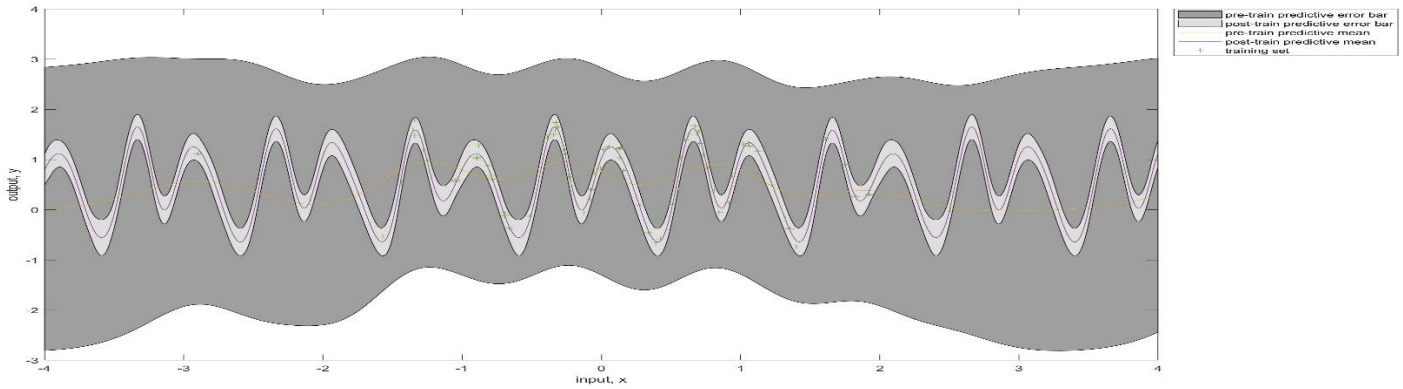

Figure 4 shows the predcitve means and error bars before and after the minimization.

Table 4 shows the values of nlml and hyperparameters before and after the minimization.

	Negative Log Marginal Likelihood ($nlml$)	Length-scale (l)	Signal Standard Deviation (v)	Noise Standard Deviation (σ_{noise})	Period (p)
Pre-train	92.73	0.37	1.00	1.00	7.39
Post-train	-1.60	0.27	0.85	0.12	3.00

In comparison to the fit from part (a), the error bar has a narrower width even for the region (outside interval [-1.5, 2]) with less training data. When comparing the marginal likelihood (a: nlml = 11.9, c: -1.6), this periodic model has a better fit to the data since it has a higher marginal likelihood and has an extra hyperparameter which increases the capacity to fit the data.

I do think the data generating mechanism was not strictly periodic even though both plots showed periodic structures for the interval [-1.5, 2]. It is possible that the data was generated with a covariance function that has several changepoints with periodic covariance function and other function. Also, there is not much data existing outside the interval [-1.5, 2] so it does not provide enough evidence to prove 'the mechanism was strictly periodic' outside the region mentioned.

---

d.

A small diagonal matrix is added to the covariance matrix by the following line. The magnitude is equal to 1e-6.

```
K = feval(covfuncd{:},hypd.cov,xd) + magnitude*eye(200);
```

Then I train the model by optimizing the hyperparameters and minimizing the nlml value and obtain the predictive mean and error bar. This process is repeated with different values of magnitude of diagonal matrix to generate different plots which are shown in Figure 5.

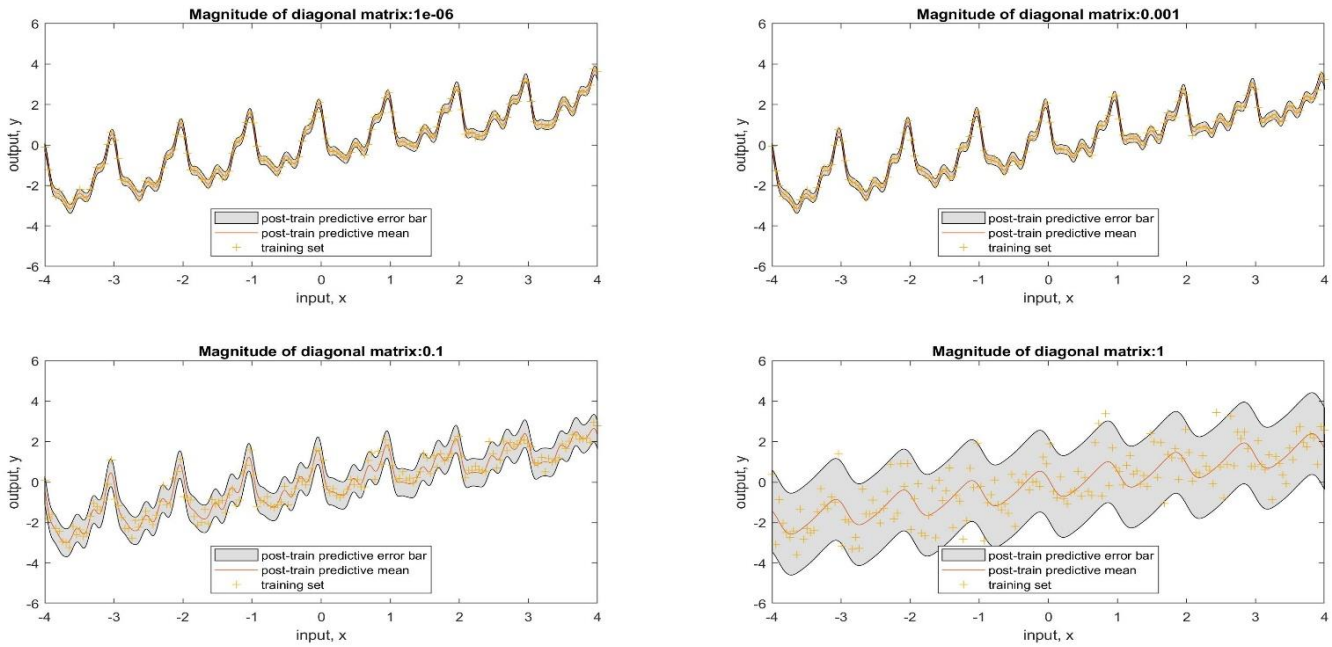

Figure 5 shows how the magnitude of the diagonal matrix affects the sample function.

To apply the Cholesky decomposition, the covariance matrix must be symmetric and positive definite. Adding the diagonal matrix helps to retain the symmetry. It also increases the values of the diagonal elements of the original covariance matrix so that the eigenvalues can all be positive, and the adjusted covariance matrix can be positive definite.

By changing the magnitude of the diagonal matrix, the noise variance is also changed. From the Figure 5, the area of the predictive error bars has increased when the magnitude of the diagonal matrix is increased. When the noise variance is increased, the nlml value is increased since it is penalized by the complexity as shown in Figure 6.

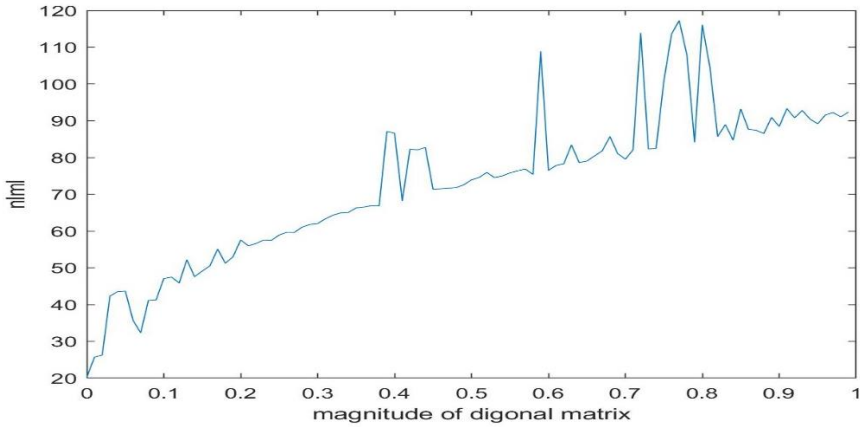

Figure 6 shows how the magnitude of the diagonal matrix affects the nlml value.

Since the covariance function is a product of periodic and squared-exponential kernels, both the predictive mean and bars oscillate in a periodic exponential pattern. Some sample functions are also plotted by changing each hyperparameter at a time based on the initial hyperparameters of log[l1, p1, stdf1, l2, stdf2] = [-0.5, 0, 0, 2, 0].

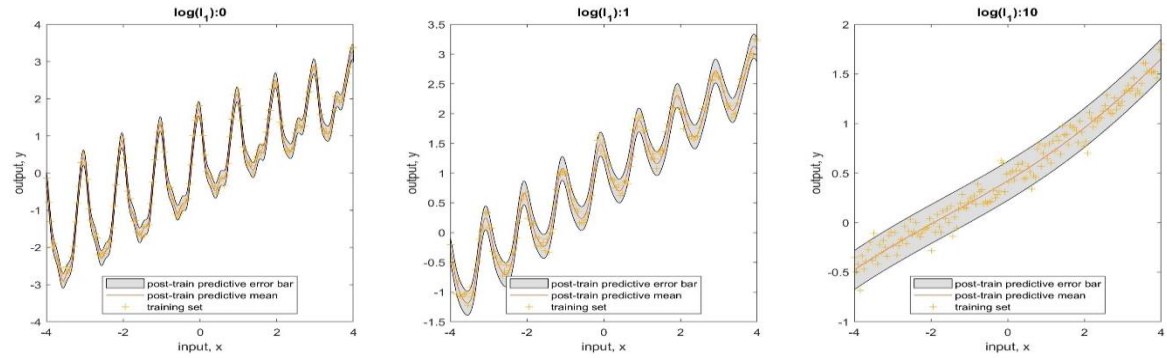

Figure 7 shows how the length-scale in the periodic covariance function affects the samples.

In terms of characteristic length-scales, the length-scale hyperparameter in the periodic function can alter the shape of the prediction (Figure 7). At a small length scale, the periodic function still retains some shape of periodic pattern. But when it grows, the periodic pattern is smoother. Later, the shape becomes almost flat with greater length-scale in periodic function.

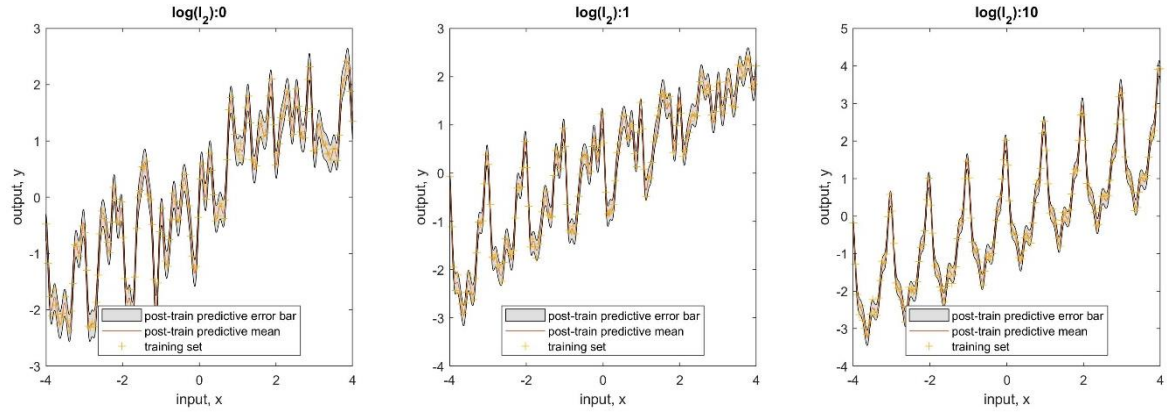

Figure 8 shows how the length-scale of the SE function affects the samples.

Refer to Figure 8, the increases in length-scale in the squared-exponential function tends to make the periodic function more stable. At a lower length-scale in SE function, the periodic pattern is irregular and has a lower period of oscillation. The periodic pattern with lower length-scale also tends to decay in amplitude.

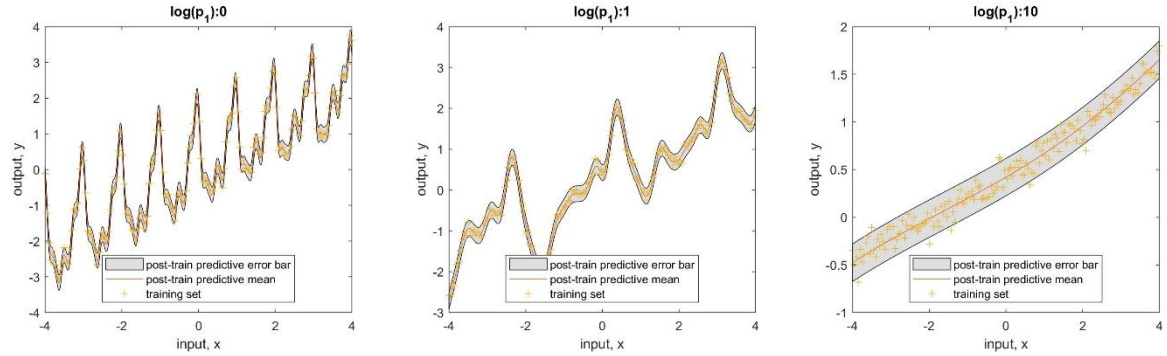

Figure 9 illustrates how the period of the periodic covariance function affects the samples..

Refer to Figure 9, when the period value of the periodic covariance function increases, it increases the width of the periodic pattern. So, the pattern oscillates less frequently.

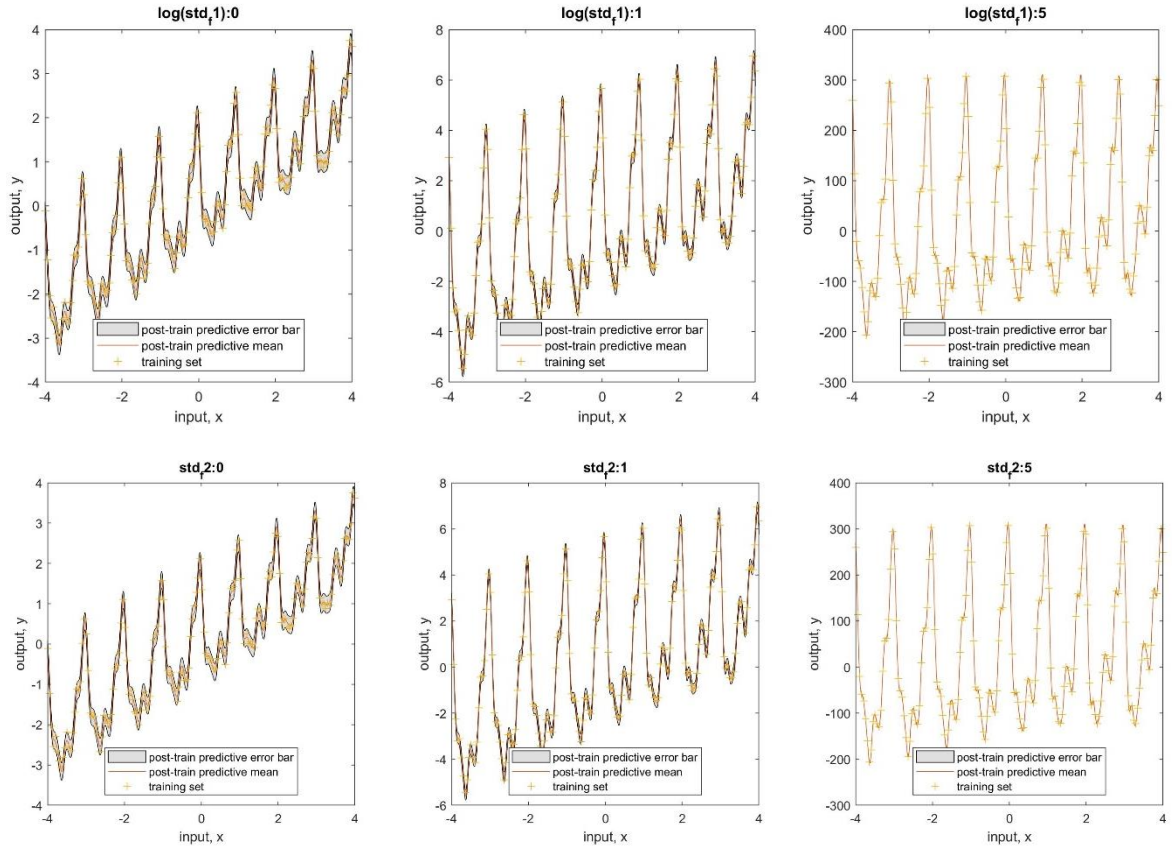

Figure 10 shows how the signal variances in both periodic (top row) and SEiso (bottom row) covariance functions affect the samples.

For the variance of the signal, when it increases, it increases the amplitude of the repeating patterns and reduces the size of the predictive error bars as shown in Figure 10. So the predictive error bar becomes thinner and thinner.

---

e.

The following lines of commands describe how I treat the hyperparameters structure.

Model 1:

```
hype1 = struct('mean',[],'cov',[0 0 0], 'lik', 0);
```

Model 2:

```
hype2 = struct('mean',[], 'cov',[0 0 0 0 0 0],'lik',0);
hype2.cov = 0.1*randn(6,1);
```

Then I train both models and obtain the predictive means and error bars for both models. Finally, to visualize the predictive mean, mesh function is used as follows to plot Figure 11(left). Note that nlz is the predictive mean symbol.

```
mesh(reshape(xs(:,1),100,100), reshape(xs(:,2),100,100),...
    reshape(nlz, 100,100))
```

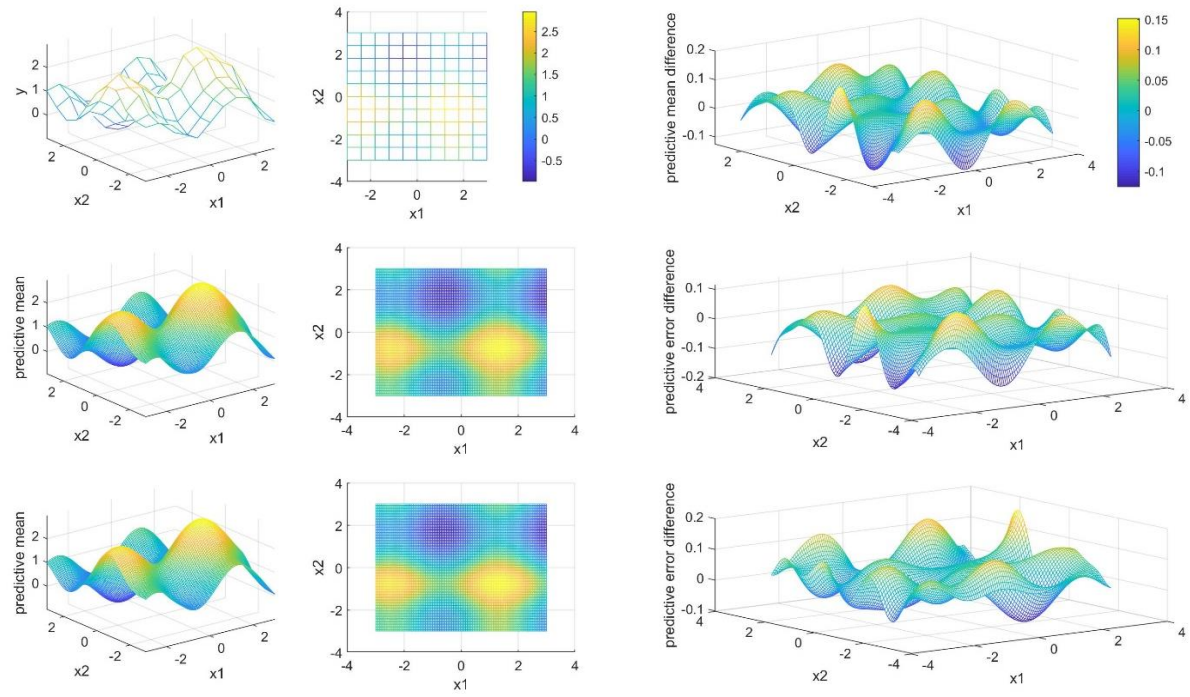

Figure 11 (left) visualizes the data at the top, the predictive mean of model 1 at the middle and the predictive mean of model 2 at the bottom.

Figure 11 (right) visualizes the difference between the predictive mean in model 1 and 2 at the top, the difference between error bar upper and lower bounds in model 1 and 2 respectively.

Table 5 shows the nlml values of model 1 and 2, and their marginal likelihood ratio.

Model	Negative Log Marginal Likelihood	Marginal Likelihood Ratio (1:2)
1	-19.22	3.27×10^{-10}
2	-66.39	

Predictions made by both model 1 and 2 show similar predictive mean shape to the data (as shown in Figure 11 (a)). By subtracting predictive means of both models, an insignificant difference is found in Figure 11 (b). Refer to Table 5, when comparing nlml values, model 2 has a better marginal likelihood so that it is a better fit to the data.

This is because model 2 is built using a sum of squared-exponential kernels which have 6 optimizable hyperparameters giving a higher degree of freedom to fit the data.