Report for project 2

Junshu Jiang

Problem 1

Let ϕ_i be the piecewise linear basis functions on $D = (a, b)$. Let T_n be subdivision on $[a, b]$ be $a = x_0 < x_1 < \cdots < x_{n-1} < x_n = b$. Then, the basis functions are defined as follows:

$$\phi_i(x) = \begin{cases} \frac{x-x_{i-1}}{x_i-x_{i-1}} & x \in [x_{i-1}, x_i] \\ \frac{x_{i+1}-x}{x_{i+1}-x_i} & x \in [x_i, x_{i+1}] \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

$$\phi_0(x) = \begin{cases} \frac{x_1-x}{x_1-x_0}, x \in [x_0, x_1] \\ \\ 0, \text{otherwise} \end{cases} \tag{2}$$

$$\phi_n(x) = \begin{cases} \frac{x-x_{n-1}}{x_n-x_{n-1}}, x \in [x_{n-1}, x_n] \\ \\ 0, \text{otherwise} \end{cases} \tag{3}$$

Let $\boldsymbol{C}$ denotes the mass matrix $\{C_{ij}\} = \langle \phi_i, \phi_j \rangle_{\mathcal{L}_2} = \int_D \phi_i \phi_j d\lambda$ where λ is Lebesgue measure. Then, consider the following cases:

1. $|i - j| = 1$, since symmetricity property of inner product: $\langle \phi_i, \phi_j \rangle_{\mathcal{L}_2} = \langle \phi_j, \phi_i \rangle_{\mathcal{L}_2}$, we only need to consider $i < j$. $\langle \phi_i, \phi_{i+1} \rangle_{\mathcal{L}_2} = \int_{x_i}^{x_{i+1}} \frac{(x_{i+1}-x)(x-x_i)}{(x_{i+1}-x_i)^2} dx = \frac{x_{i+1}-x_i}{6}.i = 0, \cdots, n-1$
2. $i = j$. For i on the boundary and not on the boundary the expression is different.
   - $i = 0:$ $\langle \phi_0, \phi_0 \rangle_{\mathcal{L}_2} = \int_{x_0}^{x_1} \left(\frac{x-x_0}{x_1-x_0}\right)^2 dx = \frac{x_1-x_0}{3}$
   - $i = n$: $\langle \phi_n, \phi_n \rangle_{\mathcal{L}_2} = \int_{x_{n-1}}^{x_n} \left(\frac{x_n-x}{x_n-x_{n-1}}\right)^2 dx = \frac{x_n-x_{n-1}}{3}$
   - $i = 1, \cdots, n-1$: $\langle \phi_n, \phi_n \rangle_{\mathcal{L}_2} = \int_{x_{i-1}}^{x_i} \left(\frac{x-x_{i-1}}{x_i-x_{i-1}}\right)^2 dx + \int_{x_i}^{x_{i+1}} \left(\frac{x_{i+1}-x}{x_{i+1}-x_i}\right)^2 dx = \frac{x_{n+1}-x_{n-1}}{3}$
3. $|i - j| \geq 2$. $\langle \phi_i, \phi_j \rangle_{\mathcal{L}_2} \equiv 0$ since ϕ_i and ϕ_j's support are disjointed.

Let $\boldsymbol{G}$ denotes the stiffness matrix $\{G_{ij}\} = \langle \phi_i', \phi_j' \rangle_{\mathcal{L}_2} = \int_D \phi_i' \phi_j' d\lambda$ where ϕ' is the weak derivative. In this question, we can directly use derivative to calculate it. Then, the derivative of basis functions are defined as follows:

$$\phi_i'(x) = \begin{cases} \frac{1}{x_i-x_{i-1}} & x \in [x_{i-1}, x_i] \\ -\frac{1}{x_{i+1}-x_i} & x \in [x_i, x_{i+1}] \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

$$\phi_0'(x) = \begin{cases} -\frac{1}{x_1-x_0}, x \in [x_0, x_1] \\ \\ 0, \text{otherwise} \end{cases} \tag{5}$$

$$\phi_n'(x) = \begin{cases} \frac{1}{x_n-x_{n-1}}, x \in [x_{n-1}, x_n] \\ \\ 0, \text{otherwise} \end{cases} \tag{6}$$

Then, consider the following cases as we did for mass matrix C:

1. $|i - j| = 1$, since symmetricity property of inner product: $\langle \phi_i', \phi_j' \rangle_{\mathcal{L}_2} = \langle \phi_j', \phi_i' \rangle_{\mathcal{L}_2}$, we only need to consider $i < j$. $\langle \phi_i', \phi_{i+1}' \rangle_{\mathcal{L}_2} = \int_{x_i}^{x_{i+1}} \frac{1}{(x_{i+1}-x_i)^2} dx = \frac{1}{x_{i+1}-x_i}.i = 0, \cdots, n-1$
2. $i = j$. For i on the boundary and not on the boundary the expression is different.
   - $i = 0:$ $\langle \phi_0, \phi_0 \rangle_{\mathcal{L}_2} = \int_{x_0}^{x_1} \frac{1}{(x_1-x_0)^2} dx = \frac{1}{x_1-x_0}$
   - $i = n$: $\langle \phi_n, \phi_n \rangle_{\mathcal{L}_2} = \int_{x_{n-1}}^{x_n} \frac{1}{(x_n-x_{n-1})^2} dx = \frac{1}{x_n-x_{n-1}}$
   - $i = 1, \cdots, n-1$: $\langle \phi_n, \phi_n \rangle_{\mathcal{L}_2} = \int_{x_{i-1}}^{x_i} \frac{1}{(x_i-x_{i-1})^2} dx + \int_{x_i}^{x_{i+1}} \frac{q}{(x_{i+1}-x_i)^2} dx = \frac{1}{x_{n+1}-x_n} + \frac{1}{x_{n+1}-x_n}$

3. $|i-j| \geq 2$. $\langle \phi_i', \phi_j' \rangle_{\mathcal{L}_2} \equiv 0$ since ϕ_i' and ϕ_j''s support are disjointed.

$$
\mathbf{C} = \begin{bmatrix} \frac{x_1-x_0}{3} & \frac{x_1-x_0}{6} & 0 & \cdots & 0 \\ \frac{x_1-x_0}{6} & \frac{x_2-x_0}{3} & \frac{x_2-x_1}{6} & \cdots & 0 \\ 0 & \frac{x_2-x_1}{6} & \frac{x_3-x_1}{3} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{x_n-x_{n-1}}{3} \end{bmatrix} ; \mathbf{G} = \begin{bmatrix} \frac{1}{x_1-x_0} & -\frac{1}{x_1-x_0} & 0 & \cdots & 0 \\ -\frac{1}{x_1-x_0} & \frac{1}{x_1-x_0}+\frac{1}{x_2-x_1} & -\frac{1}{x_2-x_1} & \cdots & 0 \\ 0 & -\frac{1}{x_2-x_1} & \frac{1}{x_2-x_1}+\frac{1}{x_3-x_2} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \frac{1}{x_n-x_{n-1}} \end{bmatrix} \tag{7}
$$

Remark. The implementation of $\boldsymbol{C}, \boldsymbol{G}$ matrix are in the attached Python code. R package *INLA* provides implementation for $\boldsymbol{C}, \boldsymbol{G}$ in function *inla.mesh.1d.fem* which are used to guarantee correct implementation.

Problem 2

The implementation for $\boldsymbol{A}_{N\times n}$ is straightforward. Iterating over N for number of locations and n for number of basis functions will give us observation matrix $\boldsymbol{A}$ of shape $N \times n$.

Goal.

1. we derive form of weak solution x_n of this type SPDE when finite element method applied. We end up with the general precision matrix Q^*_α for weak solution x_n.
2. we derive the likelihood $\log(y|\theta)$.

We only consider the case when $\alpha = 2$. For other positive integer case $\alpha = 1, 3, 4, \cdots$, we use the results in Theorem 2.

Definition 1 (Bilinear form for $L = k^2 - \Delta$ and weak solution for equation 11)**.**

1. The bilinear form associated with L is:

$$
\begin{aligned} B(Lu, v) =& (Lu, v)_{L_2} = (k^2 u, v)_{L_2} - (\Delta u, v)_{L_2} \\ =& (k^2 u, v)_{L_2} + (\nabla u, \nabla v)_{L_2} = \int_D k^2 u(s) v(s) ds - \int_D \nabla u(s)^T \nabla v(s) ds \end{aligned} \tag{8}
$$

2. for $u, v \in H^1_0(D)$, for $f \in H^{-1}(D)$, we say u is a weak solution to $Lu = f$ if

$$
B(u, v) = f(v), \forall v \in H^1_0(D)
$$

and here we consider the variational form:

$$
B(u, v) = W(v), \forall v \in H^1_0(D) \tag{9}
$$

Remark. Relax the condition in definition 1 to space V spanned by orthogonal basis $\{\phi_i\}_{i=1}^n$ of $H^1_0(D)$. By Hilbert projection theorem, we can have an approximation u_h of u in V. Now, we only consider the weak solution in V.

Theorem 1 (Galerkin's method)**.** Condtion 9 holds iff

$$
\begin{aligned} & B(u_h, \phi_i) = W(\phi_i) && i = 1, \cdots, n \\ \iff & B(\sum v_j \phi_j, \phi_i) = W(\phi_i) && i = 1, \cdots, n \\ \iff & \sum v_j B(\phi_j, \phi_i) = W(\phi_i) && i = 1, \cdots, n \\ \iff & \sum v_j (k^2 \phi_j, \phi_i) + v_j (\phi_j', \phi_i') = W(\phi_i) && i = 1, \cdots, n \end{aligned} \tag{10}
$$

Theorem 2 (Lindgren, Rue and Lindström)**.** For $\alpha \in \mathbb{N}$, a FEM approximation of following equation:

$$(k^2 - \Delta)^{\alpha/2} x = W \tag{11}$$

can be written as $x_h = \sum v_j \phi_j$ where $v = (v_1, v_n)^T \sim N(0, Q_\alpha^{-1})$ where:

$$Q_1 = K; Q_2 = K^T C^{-1} K, Q_\alpha = K^{-1} Q_{\alpha-2} C^{-1} K, \alpha = 3, 4, \cdots$$

The LHb of condition 10 is $G + k^2 C = \bar{W}$ where $\bar{W} = (W(\phi_1), \cdots, W(\phi_n))$ and $Cov[\bar{W}] = C$. Denote $K = G + k^2 C$. Since $\bar{W} \sim (0, C)$ so $\boldsymbol{v} = (v_1, \cdots, v_n) \sim N(0, K^{-1} C^{-1} K^{-T})$. Denote $Q_2 = K^T C^{-1} K$ which means the precision matrix when $\alpha = 2$. One can find it is the same in theorem 2. For other α, we directly use the result in this theorem. For the following derivation, we use a general $Q_\alpha, \alpha \in \mathbb{N}^+$.

Another issue we need to take care is τ. When take τ into consideration, the precision matrix for u would be:

$$Q_\alpha^* = diag(\tilde{\tau}) Q_\alpha diag(\tilde{\tau}) \tag{12}$$

In short $x_n(s) = \sum_{i=1}^n v_i \phi(s)$ with precision matrix Q_α^* is projection/approximation of weak solution x on subspace spanned by $\{\phi_j\}$.

Remark.

1. $x_n(s)$ is a Gaussian process.
2. Another way to understand Q_α^* is: let $(x_n(s_1), \cdots, x_n(s_n))$ be a random vector where s_i is just a point in subdivision we mention in Problem 1. Then $(x_n(s_1), x_n(s_n)) \sum \sim N(\mathbf{0}, {Q_\alpha^*}^{-1})$.
3. Explanation of observation matrix A: $\forall s_1, \cdots, s_N \in D$, $X_N = (x_n(x_1), \cdots, x_n(x_N))$ is multivaiate Gaussian and since $X_N = (\sum_{i=1}^n v_i \phi_1(s_1), \cdots, \sum_{i=1}^n v_i \phi_n(s_N)) = A\boldsymbol{v}$ where $\boldsymbol{v} = (v_1, \cdots, v_n)$ and A is the observation matrix $\mathbf{A}$ with elements $A_{ji} = \phi_i(s_j)$, $E[X_N] = \mathbf{0}_N$ and $Var[X_N] = A{Q_\alpha^*}^{-1} A^T$.

After we get $x_n(s)$, we can get $y(s)$ easily. Denote $y(s)$ as the observation process. Then we have $y(s) = x_n(s) + \epsilon(s)$ where $\epsilon_n(s)$is i.i.d at each location and has distribution $\sim N(0, \sigma^2)$.

Denote $Y_N = (y(s_1), \cdots, y(s_N))$, $X_N = (x(s_1), \cdots, x(s_N))$ Y is also Gaussian because X_N is Gaussian and independent with measure noise . Then we have $E[Y_N] = E[X_N] = \mathbf{0}_N$ and $Var[Y_N] = Var[X_N] + \sigma^2 I_N = A{Q_\alpha^*}^{-1} A^T + \sigma^2 I_N$. Therefore,

$$Y_N \sim N(\mathbf{0}_N, A{Q_\alpha^*}^{-1} A^T + \sigma^2 I_N), I_N = diag(1, \cdots, N) \tag{13}$$

And the log-likelihood function $\ell(Y_N|\theta) = log\pi(Y_N|\theta)$ is:

$$\begin{aligned} \ell(Y_N|\theta) &= -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log(\det(\boldsymbol{C}_\theta)) - \frac{1}{2}\boldsymbol{Y}_N^T \boldsymbol{C}_\theta^{-1} \mathbf{Y}_N \\ \boldsymbol{C}_\theta &= A{Q_\alpha^*}^{-1} A^T + \sigma^2 I_N \\ \theta &= (\theta^k, \theta^\tau, \sigma) \end{aligned} \tag{14}$$

Remark.

- In the implementation, constant term in $\ell(Y|\theta)$ can be ignored.
- For the implementation of $Q_\alpha^*, \alpha > 2$, it is implemented in the recursive manner

Problem 3

Since all the theoretical part is done in Problem 2, in this section we only illustrate the implementation and results. The code is in `code/Problem3&4.ipynb`. We use the *minimize* function to do the optimization. It is a method in *Package:scipy*. The optimization is done by minimizing the negative log-likelihood. The results are shown in table 1.

Converge params	σ^2	τ	κ	$-\ell(Y\|\theta)$
$\alpha = 1$	74	0.99	0.018	736
$\alpha = 2$	122	7.4	0.007	1165

Table 1: Parameter optimization when $\alpha = 1, 2$

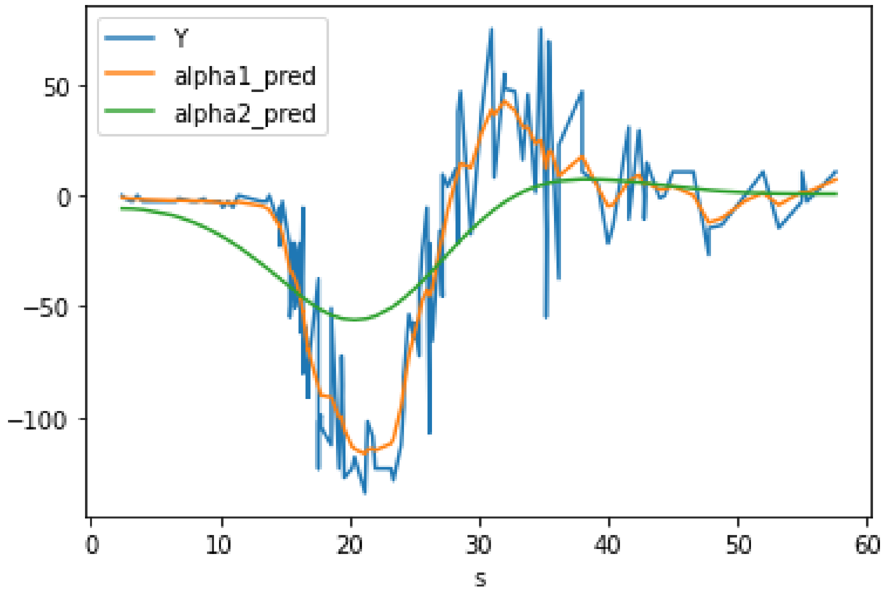


Figure 1: Conditional expectation of$E(X_N|Y_N = y_N, \theta^*)$

Problem 4

Theorem 3. conditional distribution of multivariate normal distribution

Let $X = [X_1, X_2] \sim N(\mu, \Sigma)$,then the conditional distribution of $X_1|X_2 = x_2$ is $N(\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2 - \mu_2), \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21})$

As shown in theorem 3, the point estimator for the latent process would be $E(X_N|Y_N = y_N, \theta^*)$ We use the optimized parameters in 1, to do the prediction.

$$\begin{aligned} E(X_N|Y_N = y_N, \theta^*) =& E[X_N] + \Sigma_{XY}\Sigma_{YY}^{-1}(y_n - E[Y_N]) \\ =& \Sigma_{XY}\Sigma_{YY}^{-1}y_n \\ =& Var[X_N]Var[X_N + \vec{\epsilon_N}]^{-1}y_n \\ =& [AQ_\alpha^{*\,-1}A^T][AQ_\alpha^{*\,-1}A^T + \sigma^2 I_N]^{-1}y_N \end{aligned} \tag{15}$$

Problem 5

1. As shown in Figure 1 in Problem 4, larger alpha will give more regularity to the solution. The prediction is smoother when $\alpha = 2$ than $\alpha = 1$.

2. To evaluate the performance of our modelling, wWe use $-\ell(Y_N|\theta^*, \alpha)$ negative log likelihood as assessment for the fitting. The results are shown in table 2. We iterate over hyperparemters $\alpha = 1, 2$. For each α, we iterate over number of mash location $n = 18, 47, 75, 94$ where are $20\%, 50\%, 80\%$ and 100% of the number of original data respectively. We can see that the likelihood is changing when number of basis changing.

3. Also, we don't necessarily to set the center of each ϕ_i to be the position of each location. We can arbitrarily set the center only need to make sure the observation and the region we are interested in are cover by the region of the whole domain. We set each ϕ_i with constant interval as instructed in the guide and iterate the same setting as above. The results are shown in table 3.

Remark.

- One interesting observation is that sometimes less number of basis yield better likelihood. This is an evidence of underfitting or other issues. Because theoretically, the fitting should be monotonically increasing with respect to the number of basis.
- However, it does not mean that it is always good to increase number of basis. The time complexity is increasing and the overfitting risk is also increasing when the number of basis increase without constraint.
- Overall, for the stationary model ,$\alpha = 1$ is better than $\alpha = 2$ in terms of likelihood. And from the Figure 1, it is also obvious $\alpha = 2$ introduce two much smoothness to the solution.
- Based on result in Table 2& 3, we would suggest choose the optimal setting as $n = 47, \alpha = 1$ out of consideration on time complexity and fitting.

	$n = 18$	$n = 47$	$n = 75$	$n = 94$
$\alpha = 1$	621	609	751	736
$\alpha = 2$	1015	672	883	1165

Table 2: $-\ell(Y_N|\theta^*, \alpha)$ of optimized parameters on irregular mesh

$-\ell(Y_N\|\theta^*, \alpha)$	$n = 18$	$n = 47$	$n = 75$	$n = 94$
$\alpha = 1$	623	609	606	603
$\alpha = 2$	1030	713	1039	1168

Table 3: $-\ell(Y_N|\theta^*, \alpha)$ of optimized parameters on regular mesh

Problem 6

[Leave blank]

Problem 7

In this question, we fit a non-stationary model for $k(s)$ and $\tau(s)$. The equation is shown below:

$$\log \kappa(s) = \sum_{i=0}^{k} k_i s^i \qquad (16) \qquad\qquad \log \tau(s) = \sum_{i=0}^{k} \tau_i s^i \qquad (17)$$

Since τ, k and σ are strictly positive, we use the log transformation to make sure the parameters are positive and avoid problem in optimization. The experiment results are listed in Table 4. We iterate over $\alpha = 1, \alpha = 2$ and $order = 1, 2, 3$. We can see that the order=2 has the best fit. The nonstationarity of $\kappa(s)$ and $\tau(s)$ are shown in Figure 2.

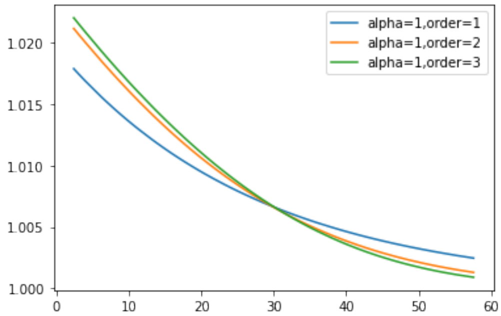
We do experiment iterating order=0, 1, 2, 3. The result is shown in table 4

order=	1	2	3	4
$\alpha = 1$	736	1102	1105	1107
$\alpha = 2$	1165	$621^{(*)}$	7016	72091

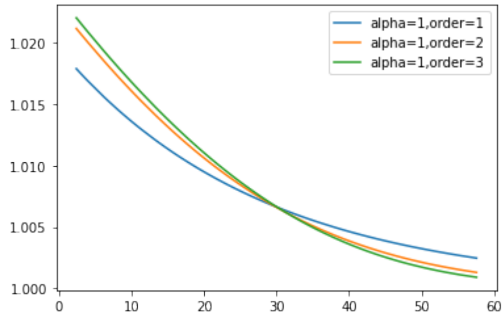
Table 4: $-\ell(Y_N|\theta^*, \alpha)$ of non-stationary model with different order

Remark.

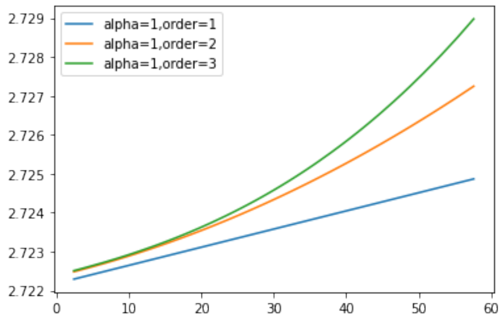
1. We include constant term in the model. The advantage is twofolds:1) Same implementation for stationary and non-stationary, when it is stationary we just set order=0; 2) The constant term helps the model to capture global effect.
2. We can see that the model with $\alpha = 2$ and order=2 has a big improvement compared with other model with $\alpha = 2$.



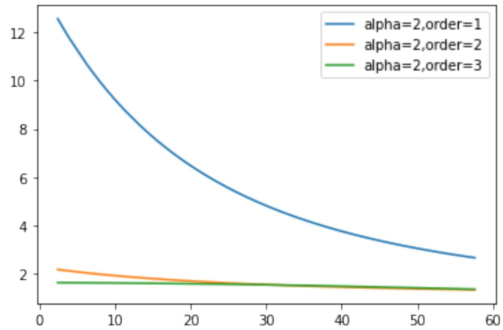
(a) $\alpha = 1, \tau^*(s)$



(b) $\alpha = 2, \tau^*(s)$



(c) $\alpha = 1, \kappa^*(s)$

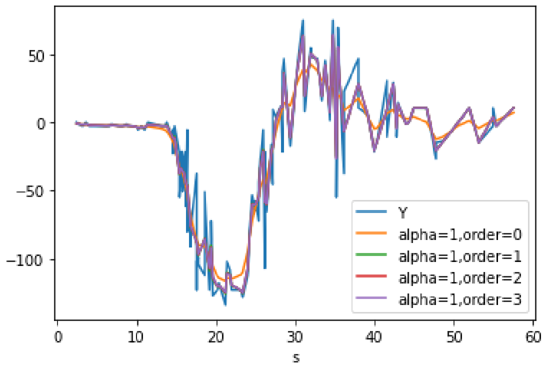


(d) $\alpha = 2, \kappa^*(s)$

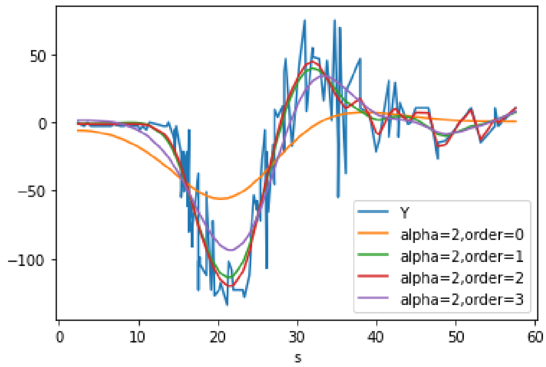
Figure 2: Nonstationary parameter estimation for k and τ

Problem 8

We estimate the latent process $X(s)$ for each observation location and for each order= $1, 2, 3, 4$. The results are shown in Figure 3. We can see that the setting $\alpha = 2, order = 2$ and setting $\alpha = 2, order = 0$ gives the best fits. For the formal model, we already study in the Problem 3,4&5. However, for the latter model, it has big improvement compared with its non-stationary counterpart.



(a) $\alpha = 1, order = 1, 2, 3$



(b) $\alpha = 2, order = 1, 2, 3$

Figure 3: Conditional expectation for non-stationary model

Problem 9

1. Yes. It looks necessary. Especially for $\alpha = 2$. When non-stationary modelling, I would only use model $\alpha = 1$. The setting $\alpha = 2, order = 2$ looks also like a promising model.

2. The variation is obvious for both $\alpha = 1$ and $\alpha = 2$. For $\alpha = 1$, the variation makes the

curve more rough. However, it is hard to tell whether it is good or not. From likelihood perspective, it is not good. For $\alpha = 2$, the variation makes the curve capable of catching the trend. Also, from the perspective of likelihood, it also improved a lot.