CS 3220: Homework 6

Instructor: Anil Damle
Due: December 14

Policies

You may discuss the homework problems freely with other students, but please refrain from looking at their code or writeups (or sharing your own). Ultimately, you must implement your own code and write up your own solution to be turned in. Your solution, including plots and requested output from your code should be submitted via the CMS as a pdf file. Additionally, please submit any code written for the assignment via the CMS as well.

Question 1

For this problem we are going to explore a mathematical model whose details span all three aspects of the course—a so-called Gaussian process. In particular, a Gaussian process is a model that can be used to capture time series or spatial data and model it via a multi-variate normal random variable. In other words, given a set of observation points $\{x_i\}$ for $i = 1, \ldots, n$ with $x_i \in \mathbb{R}^d$ a Gaussian process models observed data $\{z_i\}$ at the points $\{x_i\}$ as $\mathcal{N}(\mu, \Sigma)$ for some $\mu \in \mathbb{R}^n$ and $\Sigma \in \mathbb{R}^{n \times n}$. Here, μ_i is the mean of the observation z_i at location x_i and $\Sigma_{i,j}$ is the covariance between observations z_i and z_j at locations x_i and x_j. In other words,

$$z \sim \mathcal{N}(\mu, \Sigma)$$

and we call z_i an observation at location x_i. Importantly, the x_i are fixed, and here we will use them to define the covariance matrix. (If you would like a concrete example, you can think of the x_i as describing locations in the ocean and z_i as modeling the sea surface temperature at that point.)

So, how does this intersect with our class? We will consider a situation where we are given observed data at a set of points and would like to "fit" the underlying covariance matrix Σ. For the remainder of this problem we will consider $\mu = 0$ and $d = 2$ for simplicity. In general this problem may be tricky, but one way to make this feasible is to parametrize the entries of the covariance matrix in some way. To do this we define a kernel function

$$K(x, y) = \exp\left(-\|x - y\|_\theta^2\right) + 10^{-3}\delta(x, y)$$

for any two points $x, y \in \mathbb{R}^2$, where $\delta(x, y) = 1$ if $x = y$ and 0 otherwise. We then consider a model where for all pairs of points x_i and x_j the covariance matrix $K \in \mathbb{R}^{n \times n}$ takes the form

$$\Sigma_{i,j} = K(x_i, x_j; \theta)$$

where

$$\|x_i - x_j\|_\theta^2 = (x_i(1) - x_j(1))^2/\theta_1 + (x_i(2) - x_j(2))^2/\theta_2.$$

This model has two unknown parameters, θ_1 and θ_2 that dictate how quickly the covariance goes to zero in the two coordinate directions—they are often be described as length scales. We have also added a so-called "nugget effect" to simplify matters, this additive term models uncertainty in the measurements and could itself be unknown and estimated. Here we will use the reasonable value of 10^{-3}.

Concretely, we will assume we are given a set of observation locations $\{x_i\}_{i=1}^n$ and observed data $\{z_i\}_{i=1}^n$ that we assume is from an underlying Gaussian process that has a covariance matrix of this form. We would then like to estimate θ_1 and θ_2, we will call our estimators $\widehat{\theta}_1$ and $\widehat{\theta}_2$.

As we saw in class, one possible approach is maximum likelihood estimation and that is what we will use here. In particular, the log-likelihood function is (up to additive constants)

$$\ell(\theta_1, \theta_2) = -\frac{1}{2} z^T \Sigma^{-1} z - \frac{1}{2} \log \det(\Sigma).$$

Importantly, given a positive definite matrix Σ and its Cholesky factor L such that $LL^T = \Sigma$ we can compute its log determinant via

$$\log \det(\Sigma) = 2 \sum_{i=1}^{n} \log L_{i,i}.$$

(As an aside, in general the computation of determinants if often avoided; if absolutely necessary it is often done via triangular factorizations.) Lastly, we can compute the derivatives of the log-likelihood via

$$\frac{\delta \ell(\theta)}{\delta \theta_p} = \frac{1}{2} z^T \Sigma^{-1} \Sigma_p \Sigma^{-1} z - \frac{1}{2} \text{Trace}(\Sigma^{-1} \Sigma_p), \quad p = 1, 2,$$

where

$$[\Sigma_p]_{i,j} = \left(\frac{\delta K(x, y; \theta)}{\delta \theta_p} \right) (x_i, x_j).$$

We will simply let $\widehat{\theta}_1$ and $\widehat{\theta}_2$ come from the MLE.

(a) First, we will generate realizations of a Gaussian process to see how the parameter $\theta \in \mathbb{R}^2$ changes the behavior of the model. For various values of θ_1 and θ_2 generate realizations of a Gaussian process on a regular grid of points $\{x_i\}$ in the box $[-1, 1] \times [-1, 1]$. 50 points per direction is sufficient, this means you will have $n = 2,500$. Visualize your generated data for several choices of θ you feel exhibit clearly different behavior and discuss why you observe what you do.

(b) Now, set $\theta_1 = 0.1$ and $\theta_2 = 1$ and pick 400 random points in $[-1, 1] \times [-1, 1]$ to be the set $\{x_i\}$. You may do this by drawing independent uniform random variables for the two coordinates of the data points and scaling and shifting them appropriately. Draw a realization of a Gaussian process with these parameters and observation points and call the observations $\{z_i\}$. Implement gradient descent with a simple line search to take the points $\{x_i\}$ and data $\{z_i\}$ and minimize the negative log-likelihood to find the MLE. Two tips: you may limit the number of steps to about 50 or 100 and after every step you may set your current iterates for θ to be the maximum of their current value and 10^{-3}. (This, crudely, helps avoid accidentally leaving the feasible set of $\theta_1 > 0$ and $\theta_2 > 0$—you will probably not run into issues if you let the line search do its job.) Starting "near" the "true" parameters use your algorithm to find the MLE $\widehat{\theta}$. Plot the value of the negative log-likelihood as a function of iteration, does it behave as expected? What are your final parameter values? are they near what you expect?

(c) Given the observed data, observation points, and estimated parameters $\widehat{\theta}$ we can generate realizations of the process conditioned on the observed data. Consider a set of points $Y = \{y_i\}_{i=1}^m$ where we wish to observe realizations of the process and call this realization $f \in \mathbb{R}^m$. To describe the distribution of f conditioned on observing z we define the matrices

$$[\Sigma(X, Y; \theta)]_{i,j} = K(X_i, Y_i; \theta)$$

for arbitrary sets X and Y. (For example, if we let $X = \{x\}_{i=1}^n$ then $\Sigma(X, X; \theta)$ is simply the covariance matrix for the observed data points.)

We now have f is distributed as $\mathcal{N}(\mu_f, \Sigma_f)$ where

$$\mu_f = \Sigma(Y, X; \theta)\Sigma(X, X; \theta)^{-1}z$$

and

$$\Sigma_f = \Sigma(Y, Y; \theta) - \Sigma(Y, X; \theta)\Sigma(X, X; \theta)^{-1}\Sigma(X, Y; \theta).$$

Using your fit parameters $\widehat{\theta}$ generate several realizations of your Gaussian process on a 50×50 regular grid in $[-1, 1] \times [-1, 1]$. Provide plots that include the generated realizations and observed data, do they match what you expect? (Note that this model gives us some natural notion of uncertainties, we are just generating realizations of the underlying model and could use many of them to say something about what we expect.)