Gaussian Process Regression – Lab 1

Mines Saint-Étienne, Data Science, 2018 - 2019

For this lab session, you will use the *R* language using *RStudio* editor. The use of *R* is strongly recommended since we will be using some specific *R* packages in next sessions. A reminder of *R* basic commands are available in this link.

A few good practice when coding:

- write your code in a script file
- make sure your script file can be executed in a row
- include comments in your code
- do not hesitate to create many script files
- read the error messages!

We recall some usual covariance functions on $\mathbb{R} \times \mathbb{R}$:

$$
\begin{aligned}
\text{squared exp.} \quad & k(x,y) = \sigma^2 \exp\left(-\frac{(x-y)^2}{2\theta^2}\right) \\
\text{Matern 5/2} \quad & k(x,y) = \sigma^2 \left(1 + \frac{\sqrt{5}|x-y|}{\theta} + \frac{5|x-y|^2}{3\theta^2}\right) \exp\left(-\frac{\sqrt{5}|x-y|}{\theta}\right) \\
\text{Matern 3/2} \quad & k(x,y) = \sigma^2 \left(1 + \frac{\sqrt{3}|x-y|}{\theta}\right) \exp\left(-\frac{\sqrt{3}|x-y|}{\theta}\right) \\
\text{exponential} \quad & k(x,y) = \sigma^2 \exp\left(-\frac{|x-y|}{\theta}\right) \\
\text{Brownian} \quad & k(x,y) = \sigma^2 \min(x,y) \\
\text{white noise} \quad & k(x,y) = \sigma^2 \delta_{x,y} \\
\text{constant} \quad & k(x,y) = \sigma^2 \\
\text{linear} \quad & k(x,y) = \sigma^2 xy \\
\text{cosine} \quad & k(x,y) = \sigma^2 \cos\left(\frac{x-y}{\theta}\right) \\
\text{sinc} \quad & k(x,y) = \sigma^2 \frac{\theta}{x-y} \sin\left(\frac{x-y}{\theta}\right)
\end{aligned}
$$

Sampling from a GP

1. The script `kernFun.R` contains the implementations of the following type of kernels: linear (`linKern`), cosine (`cosKern`), and exponential (`expKern`). Each function takes as input the vectors `x`, `y` and `param` and that returns the matrix with general term $k(x_i, y_j)$. Using a similar structure, implement the functions for the Matern 5/2 (`mat5_2Kern`) kernel.

2. Create a grid of 100 points on $x, y \in [0, 1]$ and compute the covariance matrix associated to one of the kernel you wrote previously. How can you simulate zero-mean Gaussian samples based on this matrix? The function `mvrnorm()` from package *MASS* can be useful here.

3. Change the kernel and the kernel parameters. What are the effects on the sample paths? Write down your observations.

Gaussian process regression

From now on, let us choose the Matérn 5/2 kernel. We want to approximate the test function

$$f:\ x \in [0,1] \mapsto x + \sin(4\pi x) \tag{1}$$

by a Gaussian process regression model:

$$\begin{aligned} m(x) &= k(x, X)k(X, X)^{-1}Y \\ c(x, y) &= k(x, y) - k(x, X)k(X, X)^{-1}k(X, y) \end{aligned}$$

4. Create a design of experiments X composed of 15 points in the input space (regularly spaced for instance) and compute the vector of observations $Y = f(X)$.

5. Write two functions `m` and `c` that return the conditional mean and covariance. These functions take as inputs the scalar/vector of prediction point(s) `x`, the DoE vector `X`, the vector of responses `Y`, a kernel function `kern`, and the vector `param`.

6. Draw on the same graph $f(x)$, $m(x)$ and 95% confidence intervals: $m(x) \pm 1.96\sqrt{c(x, x)}$.

7. Generate samples from the conditional process.

8. Change the kernel as well as the values in `param`. What is the effect of
   - σ^2 on $m(x)$? Can you prove this result?
   - σ^2 on the conditional variance $v(x) = c(x, x)$? Can you prove this result?
   - θ on $m(x)$ (try (very) small and large values)?
   - θ on $v(x)$ (try (very) small and large values)?

Making new from old (bonus)

Implement a kernel such that the sample paths are smooth and odd functions (i.e. such that $f(x) = -f(-x)$ for all $x \in \mathbb{R}$). How does it improve the approximation on the test function 1 on the interval $[-1, 1]$? (by using the same design points X as before)?