Chapter 12: Gaussian Process Regression– Computations[a]

In this chapter we focus on the Gaussian process (GP) regression model (11.3) with known variance σ^2, which we recall is given by

$$Y_i^0 = f(x_i^0) + \epsilon_i, \quad , \epsilon_i \overset{\text{iid}}{\sim} \mathcal{N}_1(0, \sigma^2), \;\; f \sim \mathrm{GP}(0, k), \;\; \sigma^2 \sim \delta_\lambda \qquad (12.1)$$

with $\lambda > 0$ a hyperparameter.

By Proposition 11.1, $f|y_{1:n}^0 \sim \mathrm{GP}(f_n, k_n)$ where $f_n : \mathcal{X} \to \mathbb{R}$ and $k_n : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ are defined by

$$\begin{aligned} f_n(x) &= k_n(x)^\top \big(\boldsymbol{K}_n + \lambda \boldsymbol{I}_n\big)^{-1} y_{1:n}^0, && x \in \mathcal{X} \\ k_n(x, x') &= k(x, x') - k_n(x)^\top \big(\boldsymbol{K}_n + \lambda \boldsymbol{I}_n\big)^{-1} k_n(x'), && (x, x') \in \mathcal{X}^2 \end{aligned}$$

with $\boldsymbol{K}_n = \big(k(x_i^0, x_l^0)\big)_{i,l=1}^n$ and where

$$k_n(x) = \big(k(x_1^0, x), \ldots, k(x_n^0, x)\big), \quad \forall x \in \mathcal{X}.$$

Consequently, the GP regression (GPR) posterior has a closed form expression. However, computing f_n and/or k_n requires to compute the matrix $(\boldsymbol{K}_n + \lambda \boldsymbol{I}_n)^{-1}$, implying that, in general,

- The memory requirement of GPR is $\mathcal{O}(n^2)$ (to store $\boldsymbol{K}_n$).
- The computational complexity of GPR is $\mathcal{O}(n^3)$ (to invert the matrix $(\boldsymbol{K}_n + \lambda \boldsymbol{I}_n)$).

Therefore, the exact computation of the GPR posterior distribution is feasible only when n is small, and hence approximation methods are needed to make GPR widely applicable in practice.

[a] The main reference for this chapter is [12, Chapter 8].

Some particular cases

1. Let $\mathcal{X} = [0,1]$, $x_i^0 = i/n$ for $i = 1, \dots, n$ and k be such that $k(x,x') = h(|x-x'|/\gamma)$ for all $x, x' \in \mathcal{X}$ and for some function $h : [0,1] \to \mathbb{R}$. In this case, the Gram matrix $\boldsymbol{K}_n$ is Toeplitz[a] and computing $(\boldsymbol{K}_n + \lambda \boldsymbol{I}_n)^{-1}$ can be done in $\mathcal{O}(n^2)$ operations.

2. Let $m \in \mathbb{N}$, $(a_1, \dots, a_m) \in [0,\infty)^m$, $\phi_l : \mathcal{X} \to \mathbb{R}$ for $l = 1, \dots, m$, and let k be defined

$$k(x,x') = \sum_{l=1}^{m} a_l \phi_l(x) \phi_l(x'), \quad x, x' \in \mathcal{X}. \tag{12.2}$$

   Then, $\operatorname{rank}(\boldsymbol{K}_n) = \min(m,n)$ so that, for $n > m$, computing $(\boldsymbol{K}_n + \lambda \boldsymbol{I}_n)^{-1}$ requires $\mathcal{O}(m^3)$ operations.

3. Let $\mathcal{X} = \mathbb{R}$, $k(x,x') = h(|x-x'|/\gamma)$ for all $x, x' \in \mathcal{X}$ and for some function $h : [0,1] \to \mathbb{R}$, and let $f_{1:n} = (f(x_1^0), \dots, f(x_n^0))$. Then, under some conditions on h, it can be shown that the posterior distribution $\pi(f_{1:n}|y_{1:n}^0)$ corresponds to the smoothing distribution associated to a Gaussian linear state-space model [see **?** , Chapter 12]. In this case, using the Kalman filter, we can compute $\pi(f_{1:n}|y_{1:n}^0)$ exactly in $\mathcal{O}(n \log n)$ operations and with a memory requirement of size $\mathcal{O}(n)$[b]. This result for instance holds for the Matérn kernel $k_{\alpha,\gamma}$ when $\alpha = m/2$ for some $m \in \mathbb{N}$.

[a] Recall that the matrix $A = [a_{i,j}]_{i,j=1}^n$ is Toeplitz if $a_{i,j} = b_{i-j}$ for all i,j and for some real numbers $\{b_m\}_{m=-(n-1)}^{n-1}$.

[b] See Chapter 11, page 197, for the link between $\pi(f|y_{1:n}^0)$ and $\pi(f_{1:n}|y_{1:n}^0)$.

Approximation of the GPR posterior: the low rank approximation (or Nyström) approach

Following the discussion about the above second particular case, the idea of the low rank approximation approach is to:

1. Choose an integer $d < n$ according to the available computational resources (see below).

2. Find non-negative real numbers $\{a_l\}_{l=1}^d$ and real-valued functions $\{\phi_l\}_{l=1}^d$ such that the kernel $\tilde{k}^{(d)}$, defined by

$$\tilde{k}^{(d)}(x, x') = \sum_{l=1}^{d} a_l \phi_l(x) \phi_l(x'), \quad x, x' \in \mathcal{X}, \tag{12.3}$$

   is, in some sense, close to k.

3. Apply GP regression using the kernel $\tilde{k}^{(d)}$ instead of using the kernel k.

Assumption: We assume below that k is positive-definite, i.e. that $\sum_{i,j=1}^q a_i a_j k(x'_i, x'_j) > 0$ for all $a \in \mathbb{R}^q \setminus \{0\}^q$, all $x' \in \mathcal{X}^q$ and all $q \in \mathbb{N}$.

Definition of the kernel $\tilde{k}^{(d)}$

Let $\{\tilde{x}_i^0\}_{i=1}^d$ be such that $\{\tilde{x}_i^0\}_{i=1}^d = \{x_{i_l}^0\}_{l=1}^d$ for some $\{i_j\}_{j=1}^d$ verifying $1 \leq i_1 < \cdots < i_l \leq n$. To simplify the notation we relabel the elements in the set $\{x_i^0\}_{i=1}^n$ in such a way that $\tilde{x}_i = x_i^0$ for all $i \in \{1, \ldots, d\}$[a].

Next, let $\lambda_{d,1} \geq \cdots \geq \lambda_{d,d} > 0$ be the d eigenvalues of the matrix $\boldsymbol{K}_d$ and $\{u_{d,j}\}_{j=1}^d$ be an associated set of orthonormal eigenvectors.

Then, in the definition (12.3) of $\tilde{k}^{(d)}$, we let

$$a_j = \frac{\lambda_{d,j}}{d}, \quad \phi_j(x) = \frac{\sqrt{d}}{\lambda_{d,j}} k_d(x)^\top u_{d,j}, \quad \forall j \in \{1, \ldots, d\}. \tag{12.4}$$

Remark: The definition (12.4) of $\{a_j\}_{j=1}^d$ and of $\{\phi_j\}_{j=1}^d$ will be justified later.

In order to rewrite the kernel $\tilde{k}^{(d)}$ in a (much) more convenient way, let $\boldsymbol{\Lambda}_d = \mathrm{diag}(\lambda_{d,1}, \ldots, \lambda_{d,d})$ and $\boldsymbol{Q}_d \in O(d)$ be the $d \times d$ matrix whose jth column is the eigenvector $u_{d,j}$, so that

$$\boldsymbol{K}_d = \boldsymbol{Q}_d \boldsymbol{\Lambda}_d \boldsymbol{Q}_d^\top \Leftrightarrow \boldsymbol{K}_d^{-1} = \boldsymbol{Q}_d \boldsymbol{\Lambda}_d^{-1} \boldsymbol{Q}_d^\top.$$

Then, using (12.3)-(12.4), for all $x, x' \in \mathcal{X}$ we have

$$\begin{aligned}
\tilde{k}^{(d)}(x, x') &= \sum_{j=1}^d \frac{\lambda_{d,j}}{d} \frac{\sqrt{d}}{\lambda_{d,j}} k_d(x)^\top u_{d,j} \frac{\sqrt{d}}{\lambda_{d,j}} k_d(x')^\top u_{d,j} \\
&= k_d(x)^\top \Big(\sum_{j=1}^d u_{d,j} \frac{1}{\lambda_{d,j}} \big(u_{d,j}\big)^\top \Big) k_d(x') \qquad (12.5) \\
&= k_d(x)^\top \boldsymbol{Q}_d \boldsymbol{\Lambda}_d^{-1} \boldsymbol{Q}_d^\top k_d(x') \\
&= k_d(x)^\top \boldsymbol{K}_d^{-1} k_d(x').
\end{aligned}$$

[a] Below $\boldsymbol{K}_d$ and $k_d(x)$ are defined as $\boldsymbol{K}_n$ and k_n but with $n = d$.

Approximated posterior distribution

Let $\tilde{\boldsymbol{K}}_n^{(d)} = [\tilde{k}^{(d)}(x_i^0, x_l^0)]_{i,l=1}^n$ be the Gram matrix associated to the kernel $\tilde{k}^{(d)}$ and

$$\tilde{k}_n^{(d)}(x) = \big(\tilde{k}^{(d)}(x_1^0, x), \dots, \tilde{k}^{(d)}(x_n^0, x)\big), \quad \forall x \in \mathcal{X}.$$

Then, by Proposition 11.1, the posterior distribution of f given $y_{1:n}^0$ based on the kernel $\tilde{k}^{(d)}$ is such that $f|y_{1:n}^0 \sim \mathrm{GP}(f_n^{(d)}, k_n^{(d)})$, where $f_n^{(d)}$ and $k_n^{(d)}$ are defined by

$$\begin{aligned} f_n^{(d)}(x) &= \tilde{k}_n^{(d)}(x)^\top \big(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda \boldsymbol{I}_n\big)^{-1} y_{1:n}^0 \\ k_n^{(d)}(x, x') &= \tilde{k}^{(d)}(x, x') - \tilde{k}_n^{(d)}(x)^\top \big(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda \boldsymbol{I}_n\big)^{-1} \tilde{k}_n^{(d)}(x'). \end{aligned} \tag{12.6}$$

Let $\boldsymbol{K}_{nd} = [k_d(x_1^0) \dots k_d(x_n^0)]^\top$. Then, following proposition provides the key formulas for computing $f_n^{(d)}$ and $k_n^{(d)}$ in practice.

Proposition 12.1 *The functions $f_n^{(d)}$ is such that*

$$f_n^{(d)}(x) = k_d(x)^\top \big(\lambda \boldsymbol{K}_d + \boldsymbol{K}_{nd}^\top \boldsymbol{K}_{nd}\big)^{-1} \boldsymbol{K}_{nd}^\top y_{1:n}^0, \quad \forall x \in \mathcal{X}$$

while the kernel $k_n^{(d)}$ is such that

$$k_n^{(d)}(x, x') = \lambda k_d(x)^\top \big(\lambda \boldsymbol{K}_d + \boldsymbol{K}_{nd}^\top \boldsymbol{K}_{nd}\big)^{-1} k_d(x'), \quad \forall (x, x') \in \mathcal{X}^2.$$

Remark: By Proposition 12.1, it is not needed to compute the kernel $\tilde{k}^{(d)}$ in order to compute $f_n^{(d)}$ and $k_n^{(d)}$!

Remark: It is easy to check that when $d = n$ we have $f_n^{(d)} = f_n$ while $k_n^{(d)} \neq k_n$.

Proof of Proposition 12.1

The proof of Proposition 12.1 relies on the following result.

Lemma 12.1 (Matrix inversion lemma) *Let $\boldsymbol{Z}$ be an $n \times n$ matrix, $\boldsymbol{W}$ be an $m \times m$ matrix and $\boldsymbol{U}$, $\boldsymbol{V}$ be two $n \times m$ matrices. Assume that $\boldsymbol{Z}$ and $\boldsymbol{W}$ are invertible. Then,*

$$(\boldsymbol{Z} + \boldsymbol{U}\boldsymbol{W}\boldsymbol{V}^\top)^{-1} = \boldsymbol{Z}^{-1} - \boldsymbol{Z}^{-1}\boldsymbol{U}(\boldsymbol{W}^{-1} + \boldsymbol{V}^\top \boldsymbol{Z}^{-1}\boldsymbol{U})^{-1}\boldsymbol{V}^\top \boldsymbol{Z}^{-1}.$$

To prove Proposition 12.1 remark first that, using (12.5),

$$\tilde{k}^{(d)}(x) = \Big(k_d(x)^\top \boldsymbol{K}_d^{-1} k_d(x_1^0), \dots, k_d(x)^\top \boldsymbol{K}_d^{-1} k_d(x_n^0)\Big) = \boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1} k_d(x) \tag{12.7}$$

so that

$$\tilde{\boldsymbol{K}}_n^{(d)} = \boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1}\boldsymbol{K}_{nd}^\top. \tag{12.8}$$

Therefore,

$$\big(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda \boldsymbol{I}_n\big)^{-1} = \big(\boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1}\boldsymbol{K}_{nd}^\top + \lambda \boldsymbol{I}_n\big)^{-1} \tag{12.9}$$

and thus, using Lemma 12.1 with $\boldsymbol{Z} = \boldsymbol{I}_n\lambda$, $\boldsymbol{U} = \boldsymbol{V} = \boldsymbol{K}_{nd}$ and $\boldsymbol{W} = \boldsymbol{K}_d^{-1}$,

$$\begin{aligned}\big(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda \boldsymbol{I}_n\big)^{-1} &= \lambda^{-1}\boldsymbol{I}_n - \lambda^{-2}\boldsymbol{K}_{nd}\big(\boldsymbol{K}_d + \lambda^{-1}\boldsymbol{K}_{nd}^\top \boldsymbol{K}_{nd}\big)^{-1}\boldsymbol{K}_{nd}^\top \\ &= \lambda^{-1}\Big(\boldsymbol{I}_n - \boldsymbol{K}_{nd}\big(\lambda\boldsymbol{K}_d + \boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}\big)^{-1}\boldsymbol{K}_{nd}^\top\Big).\end{aligned} \tag{12.10}$$

Let $\boldsymbol{Q} = \lambda\boldsymbol{K}_d + \boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}$ and $x \in \mathcal{X}$. Then, using (12.7)-(12.10),

$$\begin{aligned}f_n^{(d)}(x) &= \tilde{k}_n^{(d)}(x)^\top\big(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda\boldsymbol{I}_n\big)^{-1} y_{1:n}^0 \\ &= k_d(x)^\top \boldsymbol{K}_d^{-1}\boldsymbol{K}_{nd}^\top\big(\boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1}\boldsymbol{K}_{nd}^\top + \lambda\boldsymbol{I}_n\big)^{-1} y_{1:n}^0 \\ &= \lambda^{-1}k_d(x)^\top\boldsymbol{K}_d^{-1}\boldsymbol{K}_{nd}^\top\big(\boldsymbol{I}_n - \boldsymbol{K}_{nd}\boldsymbol{Q}^{-1}\boldsymbol{K}_{nd}^\top\big)\, y_{1:n}^0 \qquad (12.11)\\ &= \lambda^{-1}k_d(x)^\top\boldsymbol{K}_d^{-1}\big(\boldsymbol{I}_d - \boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}\boldsymbol{Q}^{-1}\big)\boldsymbol{K}_{nd}^\top\, y_{1:n}^0 \\ &= \lambda^{-1}k_d(x)^\top\boldsymbol{K}_d^{-1}\big(\boldsymbol{Q}\boldsymbol{Q}^{-1} - \boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}\boldsymbol{Q}^{-1}\big)\boldsymbol{K}_{nd}^\top\, y_{1:n}^0 \\ &= \lambda^{-1}k_d(x)^\top\boldsymbol{K}_d^{-1}(\lambda\boldsymbol{K}_d)\boldsymbol{Q}^{-1}\boldsymbol{K}_{nd}^\top\, y_{1:n}^0 \\ &= k_d(x)^\top\boldsymbol{Q}^{-1}\boldsymbol{K}_{nd}^\top\, y_{1:n}^0 \qquad (12.12)\end{aligned}$$

showing the first part of the proposition.

Similarly, for every $x, x' \in \mathcal{X}$ we have, using (12.5) and (12.7)-(12.10),

$$\begin{aligned}k_n^{(d)}(x,x') &= \tilde{k}^{(d)}(x,x') - \tilde{k}_n^{(d)}(x)^\top\big(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda\boldsymbol{I}_n\big)^{-1}\tilde{k}_n^{(d)}(x') \\ &= k_d(x)^\top\Big(\boldsymbol{K}_d^{-1} - \boldsymbol{K}_d^{-1}\boldsymbol{K}_{nd}^\top\lambda^{-1}\big(\boldsymbol{I}_n - \boldsymbol{K}_{nd}\boldsymbol{Q}^{-1}\boldsymbol{K}_{nd}^\top\big)\boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1}\Big)k_d(x') \\ &= k_d(x)^\top\big(\boldsymbol{I}_d - \boldsymbol{Q}^{-1}\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}\big)\boldsymbol{K}_d^{-1}k_d(x') \\ &= k_d(x)^\top\big(\boldsymbol{Q}^{-1}\boldsymbol{Q} - \boldsymbol{Q}^{-1}\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}\big)\boldsymbol{K}_d^{-1}k_d(x') \\ &= k_d(x)^\top(\boldsymbol{Q}^{-1}\lambda\boldsymbol{K}_d)\boldsymbol{K}_d^{-1}k_n^{(d)}(x') \\ &= \lambda k_d(x)^\top\boldsymbol{Q}^{-1}k_d(x')\end{aligned}$$

where the third equality follows form the equality (12.11)-(12.12). The proof is complete. □

Low rank approximation: Computational complexity and uncertainty quantitation

1. The time and space complexity of GP regression with low rank approximation is as follows:

   Initialization:

   - Computing $\boldsymbol{K}_{nd}$ requires $\mathcal{O}(nd)$ operations, and the memory requirement to store this matrix is $\mathcal{O}(nd)$.

   - Computing $\boldsymbol{A}_d := \left(\lambda \boldsymbol{K}_d + \boldsymbol{K}_{nd}^\top \boldsymbol{K}_{nd}\right)^{-1}$ requires $\mathcal{O}(d^2 n)$ operations, and the memory requirement to store this matrix is $\mathcal{O}(d^2)$.

   - Computing $\boldsymbol{A}_d \boldsymbol{K}_{nd}^\top y_{1:n}^0$ requires $\mathcal{O}(d^2 n)$ operations, and the memory requirement to store this vector is $\mathcal{O}(d)$.

   Once the initialization step has been done,

   - Computing $f_n^{(d)}(x)$ for a given x requires $\mathcal{O}(d)$ operations.

   - Computing $k_n^{(d)}(x,x)$ for a given x requires $\mathcal{O}(d^2)$ operations.

2. If the kernel k is such that, for all x', $k(x,x') \to 0$ as $\|x\| \to \infty$, then $k_n^{(d)}(x,x) \to 0$ as $\|x\| \to \infty$.

   For such a kernel k, the approximate posterior variance $k_n^{(d)}(x,x)$ of $f(x)$ is close to zero when all the observations $\{\tilde{x}_j^0\}_{j=1}^d$ used to defined $\tilde{k}^{(d)}$ are far from x (!!!).

   $\Longrightarrow$ Underestimation of the uncertainty.

   Remark: By contrast, $k_n(x,x) \approx k(x,x)$ when x is far from all the observations $\{x_i^0\}_{i=1}^n$, where $k(x,x)$ is the prior variance of $f(x)$.

Understanding further the low rank approximation of the GPR posterior distribution

By (11.5), the mean function f_n can be written as follows:

$$f_n = \sum_{i=1}^{n} \alpha_{n,i} k(x_i^0, \cdot), \quad \alpha_n = \big(\boldsymbol{K}_n + \lambda \boldsymbol{I}_n\big)^{-1} y_{1:n}^0.$$

Similarly, using Proposition 12.1, $f_n^{(d)}$ can be written as follows:

$$f_n^{(d)} = \sum_{i=1}^{d} \alpha_{n,i}^{(d)} k(\tilde{x}_i^0, \cdot), \quad \alpha_n^{(d)} = \big(\lambda \boldsymbol{K}_d + \boldsymbol{K}_{nd}^\top \boldsymbol{K}_{nd}\big)^{-1} \boldsymbol{K}_{nd}^\top y_{1:n}^0.$$

Consequently, the approximation $f_n^{(d)}$ uses only d ‘basis’ functions of the set $\{k(x_i^0, \cdot)\}_{i=1}^n$ while f_n uses all of them.

Remark: Which d functions of the set $\{k(x_i^0, \cdot)\}_{i=1}^n$ are used by $f_n^{(d)}$ is determined by the choice of $\{\tilde{x}_i^0\}_{i=1}^m$.

Approximate Marginal likelihood function

Recall that the marginal likelihood plays a key role in GPR since the empirical Bayes approach is generally used to choose the hyperparameters of the model.

Proposition 12.2 *Let $\tilde{p}^{(d)}(y^0_{1:n}|\lambda,k)$ be the marginal likelihood of $y^0_{1:n}$ under the Bayesian model* (12.1) *with k replaced by $\tilde{k}^{(d)}$. Then,*

$$\log \tilde{p}^{(d)}(y^0_{1:n}|\lambda,k) = -\frac{1}{2}\Big(\log|\boldsymbol{\Sigma}_d| - \log|\boldsymbol{K}_d| + (n-d)\log\lambda\Big) - \frac{\|y^0_{1:n}\|^2 - \|\boldsymbol{\Sigma}_d^{-1/2}\boldsymbol{K}_{nd}^\top y^0_{1:n}\|^2}{2\lambda} - \frac{n}{2}\log 2\pi$$

where $\boldsymbol{\Sigma}_d = \lambda\boldsymbol{K}_d + \boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}$.

Remark: The memory requirement to compute $\log\tilde{p}^{(d)}(y^0_{1:n}|\lambda,k)$ is $\mathcal{O}(nd)$ and the computational cost is $\mathcal{O}(nd^2)$.

Proof: By Proposition 11.2,

$$\log\tilde{p}^{(d)}(y^0_{1:n}|\lambda,k) = -\frac{1}{2}\log|\tilde{\boldsymbol{K}}_n^{(d)} + \lambda\boldsymbol{I}_n| - \frac{1}{2}(y^0_{1:n})^\top(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda\boldsymbol{I}_n)^{-1}y^0_{1:n} - \frac{n}{2}\log 2\pi$$

and thus to prove the result we need to show that

$$(y^0_{1:n})^\top(\tilde{\boldsymbol{K}}_n^{(d)} + \lambda\boldsymbol{I}_n)^{-1}y^0_{1:n} = \frac{\|y^0_{1:n}\|^2 - \|\boldsymbol{\Sigma}_d^{-1/2}\boldsymbol{K}_{nd}^\top y^0_{1:n}\|^2}{\lambda} \tag{12.13}$$

$$|\tilde{\boldsymbol{K}}_n^{(d)} + \lambda\boldsymbol{I}_n| = \lambda^{n-d}|\boldsymbol{K}_d^{-1}|\,|\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd} + \lambda\boldsymbol{K}_d| \tag{12.14}$$

where (12.13) is a direct consequence of (12.10). To show (12.14) recall that, by Sylvester's determinant rule, if $\boldsymbol{A}$ is a $n\times d$ matrix and $\boldsymbol{B}$ an $d\times n$ matrix then

$$|\boldsymbol{I}_n + \boldsymbol{A}\boldsymbol{B}| = |\boldsymbol{I}_d + \boldsymbol{B}\boldsymbol{A}|. \tag{12.15}$$

Therefore,

$$\begin{aligned}
|\tilde{\boldsymbol{K}}_n^{(d)} + \lambda\boldsymbol{I}_n| = \lambda^n|\tilde{\boldsymbol{K}}_n^{(d)}/\lambda + \boldsymbol{I}_n| &= \lambda^n|(\boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1/2})(\boldsymbol{K}_d^{-1/2}\boldsymbol{K}_{nd}^\top)/\lambda + \boldsymbol{I}_n| \\
&= \lambda^n|\boldsymbol{K}_d^{-1/2}\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1/2}/\lambda + \boldsymbol{I}_d| \\
&= \lambda^n|\boldsymbol{K}_d^{-1/2}(\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd}/\lambda + \boldsymbol{K}_d)\boldsymbol{K}_d^{-1/2}| \\
&= \lambda^{n-d}|\boldsymbol{K}_d^{-1/2}(\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd} + \lambda\boldsymbol{K}_d)\boldsymbol{K}_d^{-1/2}| \\
&= \lambda^{n-d}|\boldsymbol{K}_d^{-1/2}|^2\,|\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd} + \lambda\boldsymbol{K}_d| \\
&= \lambda^{n-d}|\boldsymbol{K}_d^{-1}|\,|\boldsymbol{K}_{nd}^\top\boldsymbol{K}_{nd} + \lambda\boldsymbol{K}_d|
\end{aligned}$$

where the second equality uses (12.8) while the third equality uses (12.15) with $\boldsymbol{A} = (\boldsymbol{K}_{nd}\boldsymbol{K}_d^{-1/2})$ and $\boldsymbol{B} = \boldsymbol{A}^\top/\lambda$.□

Some comments

1. As in the previous chapter, let $\psi \in \mathbb{R}^q$ be the vector that contains all the parameters of the kernel k. Then, following the empirical Bayes approach, we set $(\lambda, \psi) = (\tilde{\lambda}_n^{(d)}, \tilde{\psi}_n^{(d)})$ where

$$(\tilde{\lambda}_n^{(d)}, \tilde{\psi}_n^{(d)}) \in \underset{\lambda \in \mathbb{R}_+, \psi \in \mathbb{R}^q}{\operatorname{argmax}} \log \tilde{p}^{(d)}(y_{1:n}^0 | \lambda, k_\psi). \qquad (12.16)$$

   In practice, an approximate solution to the optimization problem (12.16) is often obtained using a gradient based algorithm (such as a quasi-Newton algorithm). It can be shown [**?**] that the memory and computational requirement to compute the gradient of $\log \tilde{p}^{(d)}(y_{1:n}|\lambda, k)$ is $\mathcal{O}(nq)$ and $\mathcal{O}(\max(nq^2, nqd))$, respectively.

2. Although in what follows we will provide some justifications for the low rank approximation of the GPR posterior distribution, the loss of information induced by this approximation method (seems) not well understood.

3. Interestingly, several approximation methods lead to the same approximation $f_n^{(d)}$ of f_n (see [12], Chapter 8). However, these methods do not lead all to the same approximation of k_n.

Low rank approximation of the posterior: Justification

The definition (12.4) of the coefficients $\{a_j\}_{j=1}^d$ and of the functions $\{\phi_j\}_{j=1}^d$ are based on

- Mercer's theorem
- Nyström method to find a numerical solution to an integral equation (i.e. tom an equation where the unknown function appears under an integral sign).

We have already introduced Mercer's theorem in Chapter 4 (Theorem 4.1) but to justify the low rank approximation discussed above we need a slightly different version of this result.

Theorem 12.1 (Mercer's theorem, see [11], Theorem 11.15.) *Let $\mathcal{X}$ be a compact set, k be a continuous kernel on $\mathcal{X}$, μ be a probability distribution with support $\mathcal{X}$. Then, there exist real-valued continuous functions $(e_j)_{j\geq 1}$ and non negative real numbers $\lambda_1 \geq \lambda_2 \geq \dots$ such that*

$$\begin{aligned} \int_{\mathcal{X}} e_i(x)e_j(x)\mu(\mathrm{d}x) &= \delta_{i,j}, && \forall i,j \geq 1, \\ \int_{\mathcal{X}} k(x,x')e_j(x)\mu(\mathrm{d}x) &= \lambda_j e_j(x'), && \forall x' \in \mathcal{X}, \forall j \geq 1 \end{aligned} \tag{12.17}$$

and such that

$$k(x,x') = \sum_{j=1}^{\infty} \lambda_j e_j(x)e_j(x'), \quad \forall x,x' \in \mathcal{X}. \tag{12.18}$$

Remark: A continuous kernel on a compact set is called a Mercer kernel.

Low rank approximation: Basic idea

With a slight abuse of language, in what follow we refer to $(\lambda_j)_{j\geq 1}$ as the eigenvalues of k and to $(e_j)_{j\geq 1}$ as the corresponding eigenfunctions.

If $(\lambda_j)_{j\geq 1}$ and $(e_j)_{j\geq 1}$ are known then a natural idea is to take, in the definition (12.3) of $\tilde{k}^{(d)}$, $a_j=\lambda_j$ and $\phi_j=e_j$ for $j=1,\dots,d$ (i.e. we keep the d most significant eigenvalues/eigenfunctions of k). However, in general, $(\lambda_j)_{j\geq 1}$ and $(e_j)_{j\geq 1}$ are unknown and need to be estimated, which is done below using the Nyström method.

We assume in what follows that $\mathcal{X}=[0,1]^p$ and that k is continuous on $\mathcal{X}$ (so that the assumptions of Mercer's theorem are fulfilled). We also assume that $X_i^0 \overset{\text{iid}}{\sim} P_X$ for some distribution P_X with support $\mathcal{X}$ (this assumption will be used to apply Nyström method).

Then, applying Mercer's theorem with $\mu=P_X$, there exist real-valued continuous functions $(e_j)_{j\geq 1}$ and non negative real numbers $\lambda_1\geq\lambda_2,\dots$ such that (12.18) holds and such that

$$\mathbb{E}_{X\sim P_X}\big[k(X,x')e_j(X)\big]=\lambda_j e_j(x'),\ \forall x'\in\mathcal{X},\ j=1,\dots,d \tag{12.19}$$

We can therefore estimate $\{\lambda_j\}_{j=1}^d$ and $\{e_j\}_{j=1}^d$ by approximating the solutions of the integral equations (12.19).

Approximating the solution of (12.19) using Nyström method

Let $j \in \{1, \dots, d\}$ be fix and note that, since (i) $X_i^0 \overset{\text{iid}}{\sim} P_X$, (ii) $\mathcal{X}$ is compact, (iii) k is continuous on $\mathcal{X}$ and (iv) e_j is continuous on $\mathcal{X}$, it follows that (uniform weak law of large numbers)

$$\begin{aligned} &\sup_{x \in \mathcal{X}} \left| \frac{1}{n} \sum_{i=1}^{n} k(x, X_i^0) e_j(X_i^0) - \lambda_j e_j(x) \right| \\ &= \sup_{x \in \mathcal{X}} \left| \frac{1}{n} \sum_{i=1}^{n} k(x, X_i^0) e_j(X_i^0) - \int_{\mathcal{X}} k(x, x_1) e_j(x_1) \mathrm{d}x_1 \right| \to 0 \end{aligned} \tag{12.20}$$

in P_X-probability.

The result (12.20) implies that, in some sense,

$$\max_{l \in \{1,\dots,n\}} \left| \frac{1}{n} \sum_{i=1}^{n} k(x_l^0, x_i^0) e_j(x_i^0) - \lambda_j e_j(x_l^0) \right| \approx 0.$$

Therefore, if we define $u_j = n^{-1/2}(e_j(x_1^0), \dots, e_j(x_n^0))$ for $j = 1, \dots, d$, the above computations show that

$$\boldsymbol{K}_n u_j \approx n \lambda_j u_j, \quad \forall j \in \{1, \dots, d\}. \tag{12.21}$$

We now let $\lambda_{n,1} \geq \dots \geq \lambda_{n,n} > 0$ denote the n eigenvalues of the matrix $\boldsymbol{K}_n$ and $\{u_{n,j}\}_{j=1}^n$ be the corresponding (orthonormal) eigenvectors. Then,

$$\boldsymbol{K}_n u_{n,j} = \lambda_{n,j} u_{n,j}, \quad \forall j \in \{1, \dots d\}$$

which, together with (12.21), shows that

$$\frac{\lambda_{n,j}}{n} \approx \lambda_j, \quad u_{n,j} \approx u_j, \quad \forall j \in \{1, \dots d\}. \tag{12.22}$$

Remark: Under appropriate conditions it can be shown that $(\lambda_{n,j}/n) \to \lambda_j$ as $n \to \infty$.

Low rank approximating: Final step

Plugging the approximations given in (12.22) into (12.20) yields

$$\frac{1}{\sqrt{n}} k_n(x)^\top u_{n,j} \approx \frac{\lambda_{n,j}}{n} e_j(x), \quad \forall x \in \mathcal{X}, \quad \forall j \in \{1, \dots, d\}$$

implying that

$$e_{n,j}(x) := \frac{\sqrt{n}}{\lambda_{n,j}} k_n(x)^\top u_{n,j} \approx e_j(x), \quad \forall x \in \mathcal{X}, \quad \forall j \in \{1, \dots, d\}.$$

To sum-up, the above computations provide the following estimates of the first d eigenvalues and eigenfunctions of k (w.r.t P_X):

$$\lambda_j \approx \frac{\lambda_{n,j}}{n}, \quad e_j(x) \approx \frac{\sqrt{n}}{\lambda_{n,j}} k_n(x)^\top u_{n,j} \quad j = 1, \dots, d. \tag{12.23}$$

Therefore, letting $k^{(d,n)}$ the kernel defined by

$$k^{(d,n)}(x, x') = \sum_{j=1}^{d} \frac{\lambda_{n,j}}{n} e_{n,j}(x) e_{n,j}(x'), \quad x, x' \in \mathcal{X},$$

it follows that, by (12.18) and (12.23), we have $k^{(d,n)} \approx k$.

These computations suggest to take, in the definition (12.3) of $\tilde{k}^{(d)}$,

$$a_j = (\lambda_{n,j}/n), \quad \phi_j = e_{n,j}, \quad j = 1, \dots, d. \tag{12.24}$$

However, computing the eigenvalues and eigenvectors of $\boldsymbol{K}_n$ requires $\mathcal{O}(n^3)$ operations (which is precisely the cost that we want to reduce!). Therefore, instead of using the whole sample $\{x_i^0\}_{i=1}^n$ to estimate $\{\lambda_j\}_{j=1}^d$ and $\{e_j\}_{j=1}^d$, we use the subsample $\{\tilde{x}_i^0\}_{i=1}^d$.

Recalling that we are using the convention that $\tilde{x}_i^0 = x_i^0$ for all $i = 1, \dots, m$, repeating the above computations with n replaced by d (and $\boldsymbol{K}_n$ replaced by $\boldsymbol{K}_d$) gives the definition (12.4) of $\{a_j\}_{j=1}^d$ and of $\{\phi_j\}_{j=1}^d$.

Choosing the subset $\{\tilde{x}_i^0\}_{i=1}^d$: Empirical Bayes approach

Let $S_{n,d}$ be set that contains all the possible choices for the subset $\{\tilde{x}_i^0\}_{i=1}^d$ of $\{x_i^0\}_{i=1}^n$.

For a given $\{\tilde{x}_i^0\}_{i=1}^d \in S_{n,d}$ let

$$\log \tilde{p}^{(d)}(y_{1:n}^0|\{\tilde{x}_i^0\}_{i=1}^d) = \log \tilde{p}^{(d)}(y_{1:n}^0|\tilde{\lambda}_n^{(d)}, k_{\tilde{\psi}_n^{(d)}})$$

where $(\tilde{\lambda}_n^{(d)}, \tilde{\gamma}_n^{(d)})$ is as defined in (12.16) and where $\log \tilde{p}^{(d)}(y_{1:n}^0|\lambda, k)$ is as defined in Proposition 12.2.

Then, in the empirical Bayes approach, we choose the set $\{\tilde{x}_i^0\}_{i=1}^d \in S_{n,d}$ that maximizes the marginal likelihood of the observations $y_{1:n}^0$, i.e. we choose the set $\{\tilde{x}_i^{0,*}\}_{i=1}^d$ such that

$$\{\tilde{x}_i^{0,*}\}_{i=1}^d \in \underset{\{\tilde{x}_i^0\}_{i=1}^d \in S_{n,d}}{\operatorname{argmax}} \log \tilde{p}^{(d)}(y_{1:n}^0|\{\tilde{x}_i^0\}_{i=1}^d). \tag{12.25}$$

The set $S_{n,d}$ contains $\binom{n}{d}$ elements and therefore computing the marginal likelihood of $y_{1:n}^0$ for each element in this set is computationally prohibitive.

In practice, the empirical Bayes approach to select $\{\tilde{x}_i^0\}_{i=1}^d$ relies on greedy methods to approximate the solution to (12.25) (see for instance [**?**]).

Warning

If we used the Nyström method to justify the definition (12.24) of $\{a_j\}_{j=1}^d$ and $\{\phi_j\}_{j=1}^d$ then a necessary conditions for the above computations to also justify the definition (12.4) of $\{a_j\}_{j=1}^d$ and $\{\phi_j\}_{j=1}^d$ is that, as $d \to \infty$,

$$\frac{1}{d}\sum_{i=1}^{d}\varphi(\tilde{X}_i^0) \to \mathbb{E}[\varphi(X_1^0)] \quad \text{(in probability)} \tag{12.26}$$

for any $\varphi : \mathcal{X} \to \mathbb{R}$ such that $\mathbb{E}[\varphi(X_1^0)]$ exists. Indeed, a law of large number is needed for the key approximation (12.20) to hold when $\{\tilde{x}_i^0\}_{i=1}^d$ is used instead of $\{x_i^0\}_{i=1}^n$.

Consequently, using a too "exotic" mechanism for choosing the set $\{\tilde{x}_j^0\}_{j=1}^d$ may lead to a poor approximation of the eigenvalues and eigenfunctions of k, and thus to a poor approximation $\tilde{k}^{(d)}$ of k.

Letting $\mathcal{U}(S_{n,d})$ denote the uniform distribution on $S_{n,d}$, we remark that if $\{\tilde{X}_i^0\}_{i=1}^d \sim \mathcal{U}(S_{n,d})$ then (12.26) holds.

Based on this latter observation, a simple and "safe" (but probably sub-optimal) way to approximate (12.25) is to proceed as follows:

1. Choose an integer $L \geq 1$
2. Simulate L sets $\{\tilde{x}_{l,i}^0\}_{i=1}^d$ from $\mathcal{U}(S_{n,d})$
3. Choose $\{\tilde{x}_{L^*,i}^{(0)}\}_{i=1}^d$ such that

$$\{\tilde{x}_{L^*,i}^{(0)}\}_{i=1}^d \in \operatorname*{argmax}_{l\in\{1,\dots,L\}} \log \tilde{p}^{(d)}(y_{1:n}^0 | \{\tilde{x}_{l,i}^0\}_{i=1}^d).$$

Illustrative example: The fossil dataset

We recall that this dataset, already considered in Chapter 8, y_i^0 is ratio of strontium isotopes in the ith fossil and $x_i^0 \in \mathbb{R}$ is its age (measured in million of years), with $i \in \{1, \dots, n\}$ where $n = 106$. We let $k = ak_{\alpha,\gamma}$ (recalling that $k_{\alpha,\gamma}$ is the Matérn kernel) and we consider the model (12.1) with known noise variance. We let $\alpha = 5/2$ and choose the parameters (a, γ, λ) using the empirical Bayes approach[a].

Figure 12.1 below show the function f_n and its approximation $f_n^{(d)}$ obtained with $d = n/2$ and with $d = 10$.

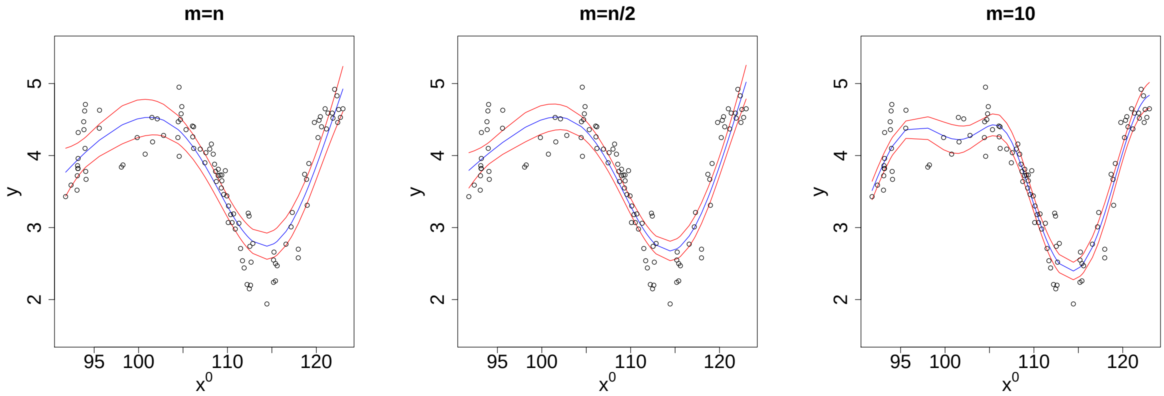

Figure 12.1: Function $f_n^{(d)}$ for $d \in \{10, n/2, n\}$ (blue curves) and (pointwise) 95% credible sets (red curves).

We observe that for $d = n/2$ the function $f_n^{(d)}$ is very similar to the function f_n, while for $d = 10$ significant differences between f_n and $f_n^{(d)}$ can be observed. We also remark that the approximate credible sets (i.e. those obtained with $d < n$) are narrower than the true credible sets. Hence, as mentioned above, the low rand approximation of the GPR posterior underestimates the uncertainty.

[a] We recall that maximizing the marginal log-likelihood function is not a convex optimization problem. It is therefore important to perform the optimization using different starting values, and to keep the solution that gives the largest value of marginal likelihood.

References

[1] Bishop, C. M. (2006). Pattern recognition. *Machine learning*, 128(9).

[2] Comon, P. (1994). Independent component analysis, a new concept? *Signal processing*, 36(3):287–314.

[3] Friedman, J., Hastie, T., Tibshirani, R., et al. (2001). *The elements of statistical learning*, volume 1. Springer series in statistics New York.

[4] Hastie, T., Tibshirani, R., and Wainwright, M. (2019). *Statistical learning with sparsity: the lasso and generalizations*. Chapman and Hall/CRC.

[5] Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430.

[6] Inaba, M., Katoh, N., and Imai, H. (1994). Applications of weighted voronoi diagrams and randomization to variance-based k-clustering. In *Proceedings of the tenth annual symposium on Computational geometry*, pages 332–339.

[7] Kanagawa, M., Hennig, P., Sejdinovic, D., and Sriperumbudur, B. K. (2018). Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*.

[8] Mairal, J. and Yu, B. (2012). Complexity analysis of the lasso regularization path. *arXiv preprint arXiv:1205.0079*.

[9] Mardia, K., Kent, J., and Bibby, J. (1979). *Multivariate analysis*. Probability and mathematical statistics. Academic Press Inc.

[10] Ng, A., Jordan, M., and Weiss, Y. (2001). On spectral clustering: Analysis and an algorithm. *Advances in neural information processing systems*, 14.

[11] Paulsen, V. I. and Raghupathi, M. (2016). *An Introduction to the Theory of Reproducing Kernel Hilbert Spaces*. Cambridge Studies in Advanced Mathematics. Cambridge University Press.

[12] Rasmussen, C. E. (2003). Gaussian processes in machine learning. In *Summer School on Machine Learning*, pages 63–71. Springer.

[13] Sande, E., Manni, C., and Speleers, H. (2020). Explicit error estimates for spline approximation of arbitrary smoothness in isogeometric analysis. *Numerische Mathematik*, 144(4):889–929.

[14] Sniekers, S., van der Vaart, A., et al. (2015). Adaptive bayesian credible sets in regression with a gaussian process prior. *Electronic Journal of Statistics*, 9(2):2475–2527.

[15] Szabó, B., Van Der Vaart, A. W., van Zanten, J., et al. (2015). Frequentist coverage of adaptive nonparametric bayesian credible sets. *The Annals of Statistics*, 43(4):1391–1428.

[16] Vaart, A. v. d. and Zanten, H. v. (2011). Information rates of nonparametric gaussian process methods. *Journal of Machine Learning Research*, 12(Jun):2095–2119.

[17] van der Vaart, A. W., van Zanten, J. H., et al. (2009). Adaptive bayesian estimation using a gaussian random field with inverse gamma bandwidth. *The Annals of Statistics*, 37(5B):2655–2675.

[18] van Wieringen, W. N. (2015). Lecture notes on ridge regression. *arXiv preprint arXiv:1509.09169*.

[19] Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416.

[20] Wood, S. N. (2017). *Generalized additive models: an introduction with R*. CRC press.