Solution 8 – MATH-250 Advanced Numerical Analysis I

($\star$) **Problem 4.** (Do note the later submission deadline due to the Easter break.)

Let $k\colon [0,1]\times[0,1]\to\mathbb{R}$ and $u\colon [0,1]\to\mathbb{R}$ be continuous functions. We define the integral operator $F\colon [0,1]\to\mathbb{R}$ as the integral of u with the kernel k using

$$F(x) = \int_0^1 k(x,y)u(y)\,\mathrm{d}y. \tag{1}$$

For a partition of $[0,1]$ into $N>0$ subintervals denote $h=\frac{1}{N}$ and let Q_h be the composite trapezoidal rule on the N subintervals of length h. Further define the subinterval's boundary points $x_i = i\cdot h$ for $i=0,1,\dots,N$.

(a) We want to apply Q_h to approximate the operator (1) at each x_i

$$Q_h[k(x_i,\cdot)u(\cdot)] = \hat{F}(x_i) \approx F(x_i) = \int_0^1 k(x_i,y)u(y)\,\mathrm{d}y, \quad i=0,1,\dots,N.$$

To this end we define the function value vectors

$$\hat{\mathbf{f}} = [\hat{F}(x_0),\hat{F}(x_1),\dots,\hat{F}(x_N)]^\top \quad \text{and} \quad \mathbf{u} = [u(x_0),u(x_1),\dots,u(x_N)]^\top.$$

Show that there exists an $(N+1)\times(N+1)$ matrix A such that $A\mathbf{u}=\hat{\mathbf{f}}$. Provide explicit formulae for the entries a_{ij} of A.

(b) We now consider the opposite idea of (a). Given a vector of the integral operator's evaluations $\mathbf{f}=[F(x_0),F(x_1),\dots,F(x_N)]^\top$ we solve $A\hat{\mathbf{u}}=\mathbf{f}$, and use the result to approximate $F(z)$ for any arbitrary value of $z\in[0,1]$.

Assume that the matrix

$$K = \begin{pmatrix} k(x_0,x_0) & k(x_0,x_1) & \cdots & k(x_0,x_N) \\ k(x_1,x_0) & k(x_1,x_1) & \cdots & k(x_1,x_N) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_N,x_0) & k(x_N,x_1) & \cdots & k(x_N,x_N) \end{pmatrix} \tag{2}$$

is invertible and show that

$$\hat{F}(z) = [k(z,x_0),k(z,x_1),\dots,k(z,x_N)]K^{-1}\mathbf{f} \tag{3}$$

holds true.

(c) For $N=4$ suppose that the corresponding matrix K from (b) is invertible.

Show that $\hat{F}(x_i)=F(x_i)$ for $i=0,1,2,3,4$.

(d) We choose the *radial basis function kernel* $k(x,y)=\exp\left(-(x-y)^2/4\right)$.

Implement a Python function `approximate_operator(F, N, z)` that computes the vector $\hat{F}(\mathbf{z}) = [\hat{F}(z_1),\hat{F}(z_2),\dots,\hat{F}(z_m)]^\top$ given a vector $\mathbf{z}=[z_1,z_2,\dots,z_m]^\top, m>0$, using (3). Assure that your implementation requires $\mathcal{O}(N^3+mN^2)$ operations.

(e) Let $N \in \{2, 5, 10\}, m = 1000, \mathbf{z} = \texttt{np.linspace(0, 1, num=1000)}$, and define

$$F_1(x) = \sin(3\pi x) \quad \text{and} \quad F_2(x) = \exp\left(-|x - 0.5|^{2/3}\right).$$

For each N and each F_i plot the true function $F_i(\mathbf{z})$ and its approximation $\hat{F}_i(\mathbf{z})$. Compute the maximum absolute error $\max_{j=1,2,\ldots,m} |F_i(z_j) - \hat{F}_i(z_j)|$ and clearly display this error.

(f) Explain the behaviour for the approximation of F_2 with $N = 10$.

To mitigate this bad approximation we utilize regularisation. This means that instead of (3) we compute

$$\hat{F}^{(\gamma)}(z) = [k(z, x_0), k(z, x_1), \ldots, k(z, x_N)](K + \gamma \text{ id})^{-1}\mathbf{f}$$

for some small $\gamma > 0$.

Implement a Python function `approximate_operator_reg(F, N, z, gamma)` to compute $\hat{F}^{(\gamma)}$ similarly to (d); you may reuse your code from (d). Determine a value for γ such that the maximum absolute error of the approximation for F_2 and $N = 10$ is less than or equal to 10^{-1}.

(g) **Bonus (This part is not needed to get full marks.)**: Prove that the matrix K from (2) is always symmetric and positive semidefinite for the radial basis kernel $k(x, y) = \exp(-(x - y)^2/4)$. You can use the Schur product theorem or any other technique.

Schur Product Theorem. Let $A, B \in \mathbb{R}^{n \times n}$ be two symmetric and positive semidefinite matrices. Then their elementwise product $(A \odot B)_{ij} = a_{ij} \cdot b_{ij}$ is once again symmetric and positive semidefinite.

Solution.

(a) We apply the definition of the composite trapezoidal rule

$$\hat{F}(x_i) = \frac{1}{N}\left(\frac{1}{2}k(x_i, x_0)u(x_0) + \sum_{j=1}^{N-1} k(x_i, x_j)u(x_j) + \frac{1}{2}k(x_i, x_N)u(x_N)\right).$$

This computation can be seen as the inner product of the vectors $\mathbf{u}$ and

$$\ell_i = \frac{1}{N}\left[\frac{1}{2}k(x_i, x_0), k(x_i, x_1), \ldots, k(x_i, x_{N-1}), \frac{1}{2}k(x_i, x_N)\right]^\top,$$

whereupon we can evaluate these inner products for all $i = 0, 1, \ldots, N$ simultaneously, thus yielding the linear system

$$\hat{\mathbf{f}} = [\ell_0, \ell_1, \ldots, \ell_N]^\top \mathbf{u} = A\mathbf{u}.$$

(b) Given the matrix A from (a) we see that $A = KD$, where D is a diagonal matrix with the principal diagonal equal to $[\frac{1}{2N}, \frac{1}{N}, \ldots, \frac{1}{N}, \frac{1}{2N}]$. Therefore, $A^{-1} = (KD)^{-1} = D^{-1}K^{-1}$ and thus also $\mathbf{u} = A^{-1}\mathbf{f} = D^{-1}K^{-1}\mathbf{f}$. We further repeat the approach from (a) and see that

$$\begin{aligned}\hat{F}(z) &= \frac{1}{N}\left[\frac{1}{2}k(z, x_0), k(z, x_1), \ldots, k(z, x_{N-1}), \frac{1}{2}k(z, x_N)\right]\mathbf{u} \\ &= [k(z, x_0), k(z, x_1), \ldots, k(z, x_{N-1}), k(z, x_N)]DD^{-1}K^{-1}\mathbf{f}.\end{aligned}$$

(c) Using the formula from (b) we can see that

$$\hat{F}(x_i) = [k(x_i, x_0), k(x_i, x_1), \ldots, k(x_i, x_N)]K^{-1}\mathbf{f}.$$

Given that the row vector on the left of K^{-1} is equal to the i-th row of K it holds that $[k(x_i, x_0), k(x_i, x_1), \ldots, k(x_i, x_N)]K^{-1} = e_i$ and therefore $\hat{F}(x_i) = F(x_i)$.

(d, e) Available in the Jupyter notebook `homework08-sol.ipynb` on Moodle.

(f) The code is available in the Jupyter notebook `homework08-sol.ipynb` on Moodle.

The issue in the bad behaviour of the approximation of F_2 for $N = 10$ is the condition number of K. The matrix K possesses an eigenvalue close to 0 and thus $\mathrm{cond}(K)$ is very large. By adding the identity matrix we increase the magnitude the eigenvalue, whence the error of the final approximation may be better, depending on the value of the regularisation parameter γ. For viable values of γ please refer to the solution in the Jupyter notebook.

(d) We commence by writing $\exp(-\frac{1}{4}(x-y)^2) = \exp(-\frac{1}{4}(x^2+y^2))\exp(\frac{1}{2}xy)$. This means that we have to show for $K = A \odot B$ with $a_{ij} = \exp(-\frac{1}{4}(x_i^2 + x_j^2))$ and $b_{ij} = \exp(\frac{1}{2}x_i x_j)$ that A and B are both positive semidefinite.

For A this follows immediately by writing $A = \mathbf{w}\mathbf{w}^\top$ with

$$\mathbf{w} = \left[\exp(-\frac{1}{4}x_0^2), \exp(-\frac{1}{4}x_1^2), \ldots, \exp(-\frac{1}{4}x_N^2)\right]^\top.$$

For B we first consider the linear kernel $g(x_i, x_j) = x_i x_j$. The associated kernel matrix can be written as $\mathbf{x}\mathbf{x}^\top$ and is therefore positive semidefinite. This semidefiniteness remains when we scale by a positive factor, thus $\frac{1}{2}\mathbf{x}\mathbf{x}^\top$ is also positive semidefinite. Next, define $M^{\odot \ell}$ the ℓ times application of $\odot$ to M, i.e. $M^{\odot 3} = M \odot M \odot M$, and define $M^{\odot 0} = E$, where E is the matrix with all entries equal to 1. This notation allows us to compactly write the Taylor series expansion of the elementwise matrix exponential, that is

$$\exp^{\odot}(M) = \begin{pmatrix} \exp(m_{11}) & \exp(m_{12}) & \ldots & \exp(m_{1N}) \\ \exp(m_{21}) & \exp(m_{22}) & \ldots & \exp(m_{2N}) \\ \vdots & \vdots & \ddots & \vdots \\ \exp(m_{N1}) & \exp(m_{N2}) & \ldots & \exp(m_{NN}) \end{pmatrix} = \sum_{\ell=0}^{\infty} \frac{1}{\ell!} M^{\odot \ell}.$$

When computing $\exp^{\odot}(\frac{1}{2}\mathbf{x}\mathbf{x}^\top)$ we therefore observe that $(\frac{1}{2}\mathbf{x}\mathbf{x}^\top)^{\odot \ell}$ is always positive semidefinite by the Schur product theorem, wherefore the same must be true for B.

To conclude the proof we observe that the kernel matrix K must necessarily be symmetric because $k(x, y) = k(y, x)$.