In problem 1.b) you looked at how the error of RBF interpolants with different kernels varies with the shape parameter ε. In that problem you should have found that for the IMQ and Gaussian kernels the error for the smooth target function

$$f(\mathbf{x}) = f(x, y, z) = \cos(2(x + \tfrac{1}{2})^2 + 3(y + \tfrac{1}{2})^2 + 5(z - \tfrac{1}{\sqrt{2}})^2) \tag{1}$$

decreased as ε decreased toward zero, but before this value was reached the behavior became erratic and you probably received warning from MATLAB of the form "Warning: Matrix is close to singular or badly scaled. Results may be inaccurate." This problem comes from the fact that as ε decreases the IMQ and Gaussian kernels become increasingly flat, approaching the value 1. Thus, the columns of the RBF interpolation matrix $A_{i,j} = \phi(\|\mathbf{x}_i - \mathbf{x}_j\|)$ look more and more alike, with the matrix having rank 1 when $\varepsilon = 0$. So, the condition number of A grows rapidly as ε decreases.

As discussed in the lecture notes, despite this ill-conditioning with the A matrix, the RBF interpolant does not blow up. In the case of interpolation in 1D, Driscoll and Fornberg [1] were the first to show that RBF interpolants in the $\varepsilon \longrightarrow 0$ limit converges to the standard Lagrange interpolating polynomial of the data. This result has been extended to interpolation in higher dimensions with the result being that RBF interpolants (usually) converges to a multivariate polynomial interpolant (unlike 1D interpolation with polynomials, multivariate polynomial interpolation is not a unique procedure). In the case of interpolation on the sphere, Fornberg and Piret [5] showed that the RBF interpolant will converge to a spherical harmonic interpolant. Thus, there is a connection between RBF interpolation and classical spectral methods (which are based on polynomial or trigonometric interpolation).

For small shape parameters, the standard way of computing the RBF interpolants by solving a linear system involving the A matrix (the RBF-Direct approach) has to be avoided and one instead has to switch to a "stable" algorithm. These algorithms formulate the interpolation problem in a different manner to avoid the ill-conditioning. Presently the following algorithms have been developed [2–7], with the RBF-QR algorithm [5] of Fornberg and Piret being the one most applicable to RBF interpolation on the sphere. An implementation of this algorithm is provided in the `rbfsphere` package with the function `rbfqrinterp`.

> The computational cost of the RBF-QR procedure is quite a bit higher than the RBF-Direct approach. So, for moderate sized shape parameters, it is recommended that you use your code from Problem 1.

2.a) Use the `rbfqrinterp` function to redo problem 1.b) for the IMQ and GA kernels. Produce a plot of the errors associated with the interpolants for $\varepsilon = 0, 0.02, \ldots, 0.9$. How do these errors compare to the ones obtained with your function for RBF interpolation developed in 1.a).

2.b) Repeat part 2.a), but now use the non-smooth target function from 1.d):

$$f(\mathbf{x}) = f(x, y, z) = \begin{cases} z & \text{for } z \geq 0 \\ 0 & \text{for } z < 0 \end{cases} \tag{2}$$

Compare these results with those from part 2.a). Plot the difference between the interpolant and the target function f for both $\varepsilon = 0$ and $\varepsilon = 0.9$. How do these plots differ? Which value of ε produces a more oscillatory interpolant?

References

[1] T. A. DRISCOLL AND B. FORNBERG, *Interpolation in the limit of increasingly flat radial basis functions*, Comput. Math. Appl., 43 (2002), pp. 413–422.

[2] G. E. FASSHAUER AND M. J. MCCOURT, *Stable evaluation of Gaussian radial basis function interpolants*, SIAM J. Sci. Comput., 34 (2012), pp. A737–A762.

[3] B. FORNBERG, E. LARSSON, AND N. FLYER, *Stable computations with Gaussian radial basis functions*, SIAM J. Sci. Comput., 33(2) (2011), pp. 869–892.

[4] B. FORNBERG, E. LEHTO, AND C. POWELL, *Stable calculation of Gaussian-based RBF-FD stencils*, Comp. Math. Applic., 65 (2013), pp. 627–637.

[5] B. FORNBERG AND C. PIRET, *A stable algorithm for flat radial basis functions on a sphere*, SIAM J. Sci. Comput., 30 (2007), pp. 60–80.

[6] B. FORNBERG AND G. WRIGHT, *Stable computation of multiquadric interpolants for all values of the shape parameter*, Comput. Math. Appl., 48 (2004), pp. 853–867.

[7] E. LARSSON, E. LEHTO, A. HERYUDONO, AND B. FORNBERG, *Stable computation of differentiation matrices and scattered node stencils based on gaussian radial basis functions*, SIAM Journal on Scientific Computing, 35 (2013), pp. A2096–A2119.