Models, Algorithms and Data (MAD): Introduction to Computing

Spring semester 2019

Prof. Dr. Jens Honore Walther
Dr. Julija Zavadlav
ETH Zentrum, CLT
CH-8092 Zürich

Set 6

Issued: 29.03.2019; Due: 07.04.2019

In this exercise, you will learn about interpolation using orthonormal basis functions and radial basis functions (RBF). Explore the pros and cons of the two types of basis functions while doing this exercise.

Question 1: Orthonormal polynomials

[Except for the visualization, this is a pen-and-paper question.]

Interpolation with the basic set of order-n polynomial basis functions $\{x^i | i = 0, ..., M\}$ has a major disadvantage: adding or deleting a basis function will require recalculating all coefficients of the new set of basis functions. This may happen when you realize that the chosen order M is not appropriate for interpolating a given data set. A set of orthonormal functions provides a solution to this problem. By construction, the coefficient of each basis function is calculated independently to the other basis functions. One way to construct a set of orthonormal functions is to use the Gram-Schmidt orthonormalization (see Section 6.1 in the lecture notes).

You are given a set of polynomial basis functions $G = \{g_1, g_2, g_3\}$, where $g_1(x) = 1, g_2(x) = x$, and $g_3(x) = x^2$ to perform function fitting on data points $\{(-3, 0)$, $(-1, 1/4)$, $(1, 1/2)$, $(3, 1)\}$.

a) Find the orthonormalized version $\Phi = \{\phi_1, \phi_2, \phi_3\}$ using Gram-Schmidt orthonormalization.

b) Use the basis function set $\{\phi_1, \phi_2\}$ to fit the data set, and visualize the result.

c) Include ϕ_3 into the set, i.e., use the complete set of Φ to fit the data set. Visualize the result and observe the improvement in the fitting. Note that the required extra effort for this task is very small.

Question 2: Surface reconstruction with Gaussian RBF

You are given a 3D data set sampled from an "unknown" 3D surface profile (see file 'q2-data.txt' and Figure 2). The data is given as $\{x_j, y_j, z_j\}$ with $j = 0, \dots, N-1$. You will perform a simplified version of 3D surface reconstruction to uncover the "unknown"/hidden surface using the 2D Gaussian radial basis functions (RBF):

$$g_i(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left(\frac{(x-\mu_{x,i})^2}{\sigma_x^2} + \frac{(y-\mu_{y,i})^2}{\sigma_y^2}\right)\right\} \tag{1}$$

where σ_x and σ_y control the influential distance of the basis function in x and y direction, respectively, and $(\mu_{x,i}, \mu_{y,i})$ are the coordinates of the centers of the radial basis function. Given a set of M radial basis functions, we wish to approximate the given data using

$$z(x, y) = \sum_{i=0}^{M-1} \alpha_i g_i(x, y), \tag{2}$$

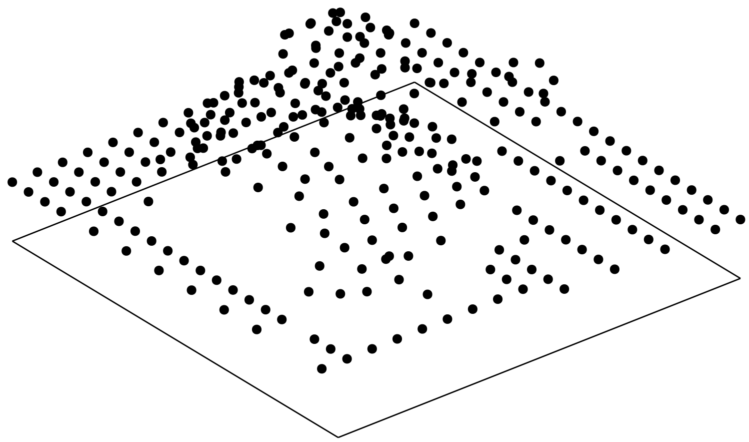

Figure 1: 3D data set for surface reconstruction with RBF.

where we wish to have $z(x_j, y_j) \approx z_j$ $(j = 0, \ldots, N-1)$. Here, we will choose $M = N$ such that we can find values of α_i to have $z(x_j, y_j) = z_j$. We will furthermore choose the Gaussian basis functions $(g_i(x, y), i = 0, \ldots, N-1)$ to be centered at each of the N data points (i.e. $\mu_{x,i} = x_i, \mu_{y,i} = y_i$). The parameters σ_x and σ_y are chosen as $\sigma_x = \sigma_y = 0.008$ (check for yourself the behavior of the RBF for different values of σ_x and σ_y).

In this task you will need to evaluate Eq. 1 with $(\mu_{x,i} = x_i, \mu_{y,i} = y_i)$. You will build a system of equations $A\,d = z$ as in

$$\underbrace{\begin{bmatrix} g_0(x_0, y_0) & g_1(x_0, y_0) & \ldots & g_{N-1}(x_0, y_0) \\ g_0(x_1, y_1) & g_1(x_1, y_1) & \ldots & g_{N-1}(x_1, y_1) \\ \vdots & \vdots & & \vdots \\ g_0(x_{N-1}, y_{N-1}) & g_1(x_{N-1}, y_{N-1}) & \ldots & g_{N-1}(x_{N-1}, y_{N-1}) \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{N-1} \end{bmatrix}}_{d} = \underbrace{\begin{bmatrix} z_0 \\ z_1 \\ \vdots \\ z_{N-1} \end{bmatrix}}_{z} \tag{3}$$

Finally, the system is solved and the resulting parameters α_i can be used in Eq. 2 to evaluate $z(x, y)$ at any point in space.

You are provided with a skeleton code in the file 'q2.py' (if you prefer you can write your own code instead). The provided code:

- reads data from the file 'q2-data.txt',
- (TO-DO) solves a linear system of equations,
- evaluates the sum of RBF on a fine grid of x and y values and
- plots the result.