Data Analysis II - 2023

Exercise sheet no 2: Numerical optimisation — March 7, 2023

- `scipy.optimize.minimize(function, startval)`

Exercise 1: Minimising a function with numerical methods (20 Points)

The goal of this exercise sheet is to find the global minimum of the function,

$$f(x) = 300 + 50x - 4x^2 + 15x^3 - 20x^4 + 0.5x^5 + x^6, \tag{1.1}$$

using numerical methods. The first and second derivatives are the following:

$$f'(x) = 50 - 8x + 45x^2 - 80x^3 + 2.5x^4 + 6x^5, \tag{1.2}$$

$$f''(x) = -8 + 90x - 240x^2 + 10x^3 + 30x^4. \tag{1.3}$$

(a) Employ the Newton-Raphson method starting at $x_0 = 0$ to calculate the minimum value $f(x)_{min}$ and the x value at that minimum, x_{min}. Note down the CPU time and number of iterations needed to achieve a precision of 10^{-3} in x_{min}. (2 points)

(b) Repeat the exercise in a) but instead start at $x_0 = -10$. Did you obtain the same result? Which one corresponds to the global minimum? (1 points)

(c) Repeat the exercise in a) starting with $x_0 = 0$ but use gradient descent as shown in the lecture. Use a step size multiplier of $\gamma = 0.001$. Does it converge after 100 iterations? (2 points)

(d) Try the gradient descent again from $x_0 = 0$ but make the step size multiplier, γ, smaller. If it converges, note down the CPU time and number of iterations needed to achieve a precision of 10^{-3} in x_{min}. (2 points)

(e) Now imagine that the derivatives are unknown (e.g. not analytical). Use the central distance method to calculate the derivative needed for gradient descent and minimise again the function. Compare again the number of iterations and CPU time needed to point d). (2 points)

(f) Now use the `scipy.optimize.minimize` python method to optimise the function. Play around with the starting x_0-value until the program reports a successful convergence. Compare the CPU time with the Newton-Raphson method and gradient descent. (1 point)

Exercise 2: Likelihood fit with many parameters (10 Points)

The file `signal_and_background.txt` contains a list of random numbers that have been generated following a Gaussian signal distribution on top of a constant background. The pdf is then given by

$$P(x) = fG(x) + (1 - f)U(x), \tag{2.1}$$

where $G(x)$ is the Gaussian pdf with $\mu_s = 5$ and standard deviation $\sigma_s = 0.75$, $U(x)$ is the pdf of an uniform distribution in $[0, 10]$, and f is the fraction of signal in data.

(a) Write a function that evaluates the pdf for any of the random numbers for any given f, μ_s and σ_s. (2 points)

(b) Write a function which calculates the likelihood of the data, for given values of the three parameters μ_s, σ_s and f (4 points).

(c) Use the `scipy.optimise.minimize` method to find the best values of the parameters. Do the best fit values depend on the starting parameters? Hint: make sure the values of f are bounded to meaningful values in the minimization. (4 points)

Deadline for submission: Tuesday, 21 March 2023 18:00
Form: Please submit your solutions to OLAT. The solutions should be submitted as a single python script which creates the answers to the questions.