1 Optimization in Model Fitting: Learning Rate Parameters of a Biochemical Reaction Model

In this problem, we will explore a use of optimization in the context of model fitting. We will specifically try to learn rate parameters of a biochemical reaction model. Let us suppose that we are studying a reaction system:

$$2A + B \rightleftharpoons C$$

with forward rate k_1 and reverse rate k_2. This can be described by a system of differential equations:

$$\frac{dA}{dt} = -2k_1A^2B + 2k_2C\frac{dB}{dt} = -k_1A^2B + k_2C\frac{dC}{dt} = k_1A^2B - k_2C$$

We will assume that we are given some measured concentrations of these molecules versus time and would like to learn the rate constants of the reaction.

a. Forward Euler Simulator

We will first need a way to simulate the reaction system. Write pseudocode for a forward Euler simulator that takes starting concentrations A_0, B_0, and C_0, a step size Δt, and a number of time steps n and produces a matrix of concentrations X of each of the reactants at each time point, where X_{1i} is the value of $A(i\Delta t)$, X_{2i} is the value of $B(i\Delta t)$, and X_{3i} is the value of $C(i\Delta t)$.

b. Goodness of Model Fit

We will next need a way to evaluate goodness of model fit. Suppose we have simulated matrix X computed by your routine above. We will assume our real data is a matrix Y with the same dimensions providing measured values of the reactants. To keep things simple, we will assume that our simulated values correspond to the same time points as our measured values. We will assess quality of fit by a least-squares measure $\Phi(k_1, k_2)$, i.e., sum of squares of differences between the corresponding elements of X and Y.

c. Numerical Gradient and Hessian

Since we depend on simulations to evaluate our objective function, we will need to compute gradients and Hessians numerically. Given current parameter estimates k_1 and k_2 and a perturbation Δk, provide numerical formulas to estimate the gradient and Hessian of $\Phi(k_1, k_2)$. First-order accuracy is sufficient and you can assume other parameters of the problem are as specified above.

d. Newton-Raphson Solver

Provide pseudocode to link your routines above into a Newton-Raphson solver to fit the parameters. You can assume you are given as input initial guesses as to the parameters k_1 and k_2 and a fixed number of rounds r of NR to run. Assume other parameters of the problem are defined as above.

e. Implementation

Write code implementing your full simulation-based Newton-Raphson model fitter. It should take as input a word file

f. Usage

Test your code on the provided example files PS3test1.txt and PS3test2.txt and provide the final best-fit k1 and k2 for each.