Problem Set 4

Due date: Feb 12, 2021

Instructions: Hand-in is in electronic form via email. Please create a single PDF-file containing all your answers and results. Show the names and student numbers of the group members on top. Maximum 4 students can work in a group. Make use of figures and tables. The code should be well documented and readable. Have a look at Latex-Example_with_Matlab (on Olat) for how to include Matlab code in a Latex file. Pay attention to the use of mcode package.

Exercise 1: Simple Functions

- Implement the Newton algorithm for any function $f : \mathbf{R} \to \mathbf{R}$
- Calculate the extrema of the two functions

$$f_1(x) = 2x^3 - x^2 - 3x + 2$$
$$f_2(x) = -x \exp(-x).$$

- Plot the functions and show the critical points on the graphs

Exercise 2: Solving The Augmented Solow Growth Model

The full code for this exercise is provided on OLAT. You do NOT need to program anything yourself. Just have a look at the code and answer the questions in which you are asked to derive or to interpret something.

Consider the human capital augmented Solow growth model (see the pdf on Augmented Solow Growth Model). Let the production function be Cobb-Douglas.You

Table 1: Parameters

s_K	s_H	n	g	δ_k	δ_h	z	α_K	α_H	k_0	h_0
0.200	0.200	0.010	0.015	0.10	0.06	1	0.33	0.33	3	3

Program the different algorithms enumerated below. The numerical root-finding algorithms should automatically adjust to the dimension of the problem, i.e., accept inputs that can be scalars or matrices.[1] Use each in turn to solve the augmented Solow model, i.e., find a root of the appropriate nonlinear equation system. The parameterization is given in table 1, with k_0 and h_0 denoting the starting values.[2] For each algorithm, report the number of iterations and the running time, and compare them.

1. Derive the analytic solution $x^* = (k^*, h^*)$, and compute its value for the given parameterization. (Refer to file `Augmented.Solow.Modell.pdf` for the model details.)
2. Now assume an analytic solution was not available, so that the nonlinear system of equations has to be solved numerically. Write a Matlab function that returns the function value and the Jacobian of the steady state condition.

[1] By contrast, the economic model, i.e., the augmented Solow model, has a fixed dimension.

[2] Other starting values may yield results that are not real, i.e., have a complex part. You may want to check with `isreal`.

3. Program and use Newton's method for root-finding. Letting $\hat{x}^* = \left(\hat{k}^*, \hat{h}^*\right)$ denote the approximate solution, report $\|\hat{x}^* - x^*\|_\infty$. Increase the accuracy of your algorithm by setting stricter convergence criteria, again report $\|\hat{x}^* - x^*\|_\infty$, and compare. (Alternatively, you may also decrease the accuracy and compare.)

4. Program and use the Broyden method for root-finding. Use the identity matrix as the initial guess for the Jacobian.

5. Program and use the Inverse Broyden method for root-finding . Use the identity matrix as the initial guess for the Jacobian.

6. Program and use a fixed-point iteration (hint: the supplied function has a slightly different output than in the previous methods). Save all results from the iteration steps. Display a plot showing the results for k over the iterations along with a horizontal line showing at the value of k^*. Do the same for h. Provide an economic interpretation of the plots.

Exercise 3: Cournot Oligopoly

Consider the following Cournot oligopoly model with n firms. The inverse demand function is

$$P(Q) = q^{-1/\lambda} = \Big(\sum_{i=1}^{n} q_i\Big)^{-1/\lambda}$$

Production costs are

$$c_i(q_i) = \frac{1}{2}\zeta_i q_i^2, \quad \forall i = 1, \dots, n$$

Firm i's profits are given by

$$\pi_i(q_1, \dots, q_n) = \Big(\sum_{i=1}^{n} q_i\Big)^{-1/\lambda} q_i - \frac{1}{2}\zeta_i q_i^2$$

For its optimization problem, firm i takes the output of all other firms as given. Thus, equilibrium output levels $(q_1, \dots, q_i)$ are the solution to

$$\frac{\partial \pi_i(q_1, \dots, q_n)}{\partial q_i} = \Big(\sum_{i=1}^{n} q_i\Big)^{-1/\lambda} - \frac{1}{\lambda}\Big(\sum_{i=1}^{n} q_i\Big)^{-1/\lambda - 1} q_i - \zeta_i q_i = 0, \quad \forall i = 1, \dots, n$$

Let $\lambda = 1.6$. The firm-specific costs, ζ_i, take their values on an equally spaced grid between 0.6 and 0.8, i.e., $\{\zeta_i = 0.6 + (i-1)\frac{0.8-0.6}{n-1},\ i = 1, \dots, n\}$.

1. Compute the equilibrium allocations for $n = 2$ firms.

2. Compute the equilibrium allocations for $n = 5$ firms.

3. Compute the equilibrium allocations for $n = 10$ firms.